

М. Н. КИРСАНОВ

Maple и Maplet

Решения задач механики

УЧЕБНОЕ ПОСОБИЕ



САНКТ-ПЕТЕРБУРГ • МОСКВА • КРАСНОДАР
2012



ББК 22.21я73
К 43

Кирсанов М. Н.

К 43 Maple и Maple. Решения задач механики: Учебное пособие. — СПб.: Издательство «Лань», 2012. — 512 с.: ил. — (Учебники для вузов. Специальная литература).

ISBN 978-5-8114-1271-6

Изложены решения задач по теоретической механике и сопротивлению материалов: определение реакций опор составных конструкций, кинематический анализ многозвенных механизмов, составление уравнений движения в форме Лагранжа, расчет статически неопределимых рам и др. Даны тексты и пояснения к 55 вспомогательным и иллюстративным программам для решения задач в системе Maple, алфавитный предметный и именной указатель к командам и операторам этой системы, содержащий более 1600 имен и терминов. Описаны примеры программирования пользовательского интерфейса Maple и основные пакеты библиотеки Maple. Программы написаны для Maple 13, но могут быть использованы во всех версиях, начиная с Maple 8.

Книга может быть использована как при очной, так и при дистанционной формах обучения.

Для студентов и преподавателей университетов и технических вузов.

ББК 22.21я73

Рецензент:

А. В. МАТРОСОВ — кандидат технических наук, доцент кафедры информационных систем Санкт-Петербургского государственного университета.

Обложка

Н. А. ГОНЧАРОВА

Охраняется законом РФ об авторском праве. Воспроизведение всей книги или любой ее части запрещается без письменного разрешения издателя. Любые попытки нарушения закона будут преследоваться в судебном порядке.

© Издательство «Лань», 2012
© М. Н. Кирсанов, 2012
© Издательство «Лань»,
художественное
оформление, 2012



СОДЕРЖАНИЕ

Предисловие	7
Глава 1. Статика	9
1.1. Рамы	12
1.2. Трение	28
1.3. Ферма	32
1.4. Пространственная статика	45
Глава 2. Кинематика	50
2.1. Кинематика точки	51
2.2. Методы расчета кинематики плоского движения	55
2.3. Угловые скорости звеньев механизма	60
2.4. Угловые ускорения звеньев механизма	71
2.5. Механизм с муфтой	74
Глава 3. Динамика	86
3.1. Кинетическая энергия системы	86
3.2. Принцип возможных перемещений	89
3.3. Уравнение Лагранжа 2-го рода	91
3.3.1. Системы с одной степенью свободы	91
3.3.2. Системы с двумя степенями свободы	108
3.4. Функция Рауса	113
3.5. Уравнения Гамильтона	116
3.6. Колебание узла фермы	117
3.7. Уравнения движения с множителями связей	123
Глава 4. Соппротивление материалов	127
4.1. Моменты инерции сечения	127
4.2. Кручение вала	134
4.3. Секториальные характеристики сечения	140
4.4. Кручение тонкостенных стержней открытого профиля	154
4.5. Напряжения и деформации	158
4.6. Прогиб фермы	164
4.7. Статически неопределимые стержневые системы	166
4.7.1. Шарнирно закрепленный брус	166
4.7.2. Свободное твердое тело на упругих стержнях	176
4.8. Изгиб балки	179
4.9. Статически определимая рама	191

4.10. Статически неопределимая рама	196
4.11. Трехшарнирная арка	202
4.12. Двухшарнирная арка	206
4.13. Устойчивость	210
4.14. Стабильность	215
Глава 5. Введение в Maple	226
5.1. Ввод информации и простейшие вычисления	227
5.2. Программирование	229
5.2.1. Оператор цикла	230
5.2.2. Условный оператор	231
5.2.3. Процедуры	232
5.2.4. Функции	234
5.3. Преобразование и упрощение выражений	234
5.3.1. Оператор simplify	235
5.3.2. Оператор combine	236
5.3.3. Операторы factor, ifactor	237
5.3.4. Оператор collect	237
5.3.5. Оператор isolate	238
5.3.6. Оператор subs	238
5.3.7. Операторы ввода и вывода	239
5.4. Алгебраические операции	241
5.4.1. Простые операции	241
5.4.2. Пакет LinearAlgebra	243
5.5. Графики и рисунки	263
5.5.1. Простые возможности операторов plot и plot3d	263
5.5.2. Основные опции оператора plot	265
5.5.3. Пакет ImageTools	268
5.5.4. Пакет plots	271
5.5.5. Основные операторы пакета plottools	279
5.5.6. Основные опции оператора plot3d	287
5.5.7. Графические структуры	291
5.5.8. Пример 3-D анимации в Maple	292
5.6. Дифференциальные уравнения	294
5.6.1. Оператор dsolve	295
5.6.2. Пакет DEtools для решения обыкновенных дифференциальных уравнений	297
Глава 6. Программирование Maple	304
6.1. Первая программа	305
6.2. BoxColumn, BoxRow	307

6.3. Window	310
6.4. Label	313
6.5. Button	315
6.6. TextField	315
6.7. CheckBox	317
6.8. TextBox	318
6.9. ToggleButton	319
6.10. Table	320
6.11. Slider	321
6.12. RadioButton	322
6.13. Plotter	323
6.14. MathMLViewer	325
6.15. DropDownBox	325
6.16. ComboBox	326
6.17. ListBox	326
6.18. Диалоги	328
6.19. Maplet Builder	335
Глава 7. Maple-программы	338
7.1. Рама	338
7.2. Составная конструкция	339
7.3. Система с односторонними связями	341
7.4. Ферма. <i>Maplet</i>	342
7.5. Полка	349
7.6. Статические инварианты	352
7.7. Набор стандартных процедур для рисунков	354
7.8. Кинематика точки	357
7.9. Движение точки в полярных координатах	359
7.10. Многозвенный механизм	360
7.10.1. Расчет угловых скоростей	360
7.10.2. Анимация	362
7.11. Уравнения трех угловых ускорений	366
7.12. Механизм с муфтой	367
7.13. Кинетическая энергия	371
7.14. Уравнение Лагранжа 2-го рода. Одна степень свободы	372
7.15. Уравнение Лагранжа 2-го рода. Две степени свободы	376
7.16. Функция Рауса	381
7.17. Уравнения Гамильтона	386
7.18. Паллограф	387
7.19. Моменты инерции поперечного сечения. <i>Maplet</i>	390

7.20. Напряженное состояние в точке	398
7.21. Статически неопределимые системы	402
7.22. Кручение	407
7.22.1. Кручение неоднородного вала	407
7.22.2. Статически неопределимая задача о кручении	408
7.22.3. Кручение вала прямоугольного сечения	410
7.23. Кручение стержня открытого профиля.	413
7.23.1. Секториальные характеристики сечения. <i>Маллет</i>	413
7.23.2. Интегрирование уравнения кручения	419
7.24. Балка. <i>Маллет</i>	421
7.25. Статически определимая рама	428
7.25.1. Алгоритм	428
7.25.2. Файл данных о раме	430
7.25.3. Основная программа. Примеры	433
7.25.4. Подпрограммы	440
7.25.5. Определение реакций стандартным методом	445
7.26. Разрезная балка	446
7.26.1. Ввод данных	446
7.26.2. Программа	448
7.26.3. Балка на двух опорах	450
7.27. Линии влияния в разрезной балке. <i>Маллет</i>	450
7.28. Статически неопределимая рама	455
7.28.1. Ввод данных. Примеры	455
7.28.2. Работа программы	458
7.28.3. Кинематическая проверка	465
7.28.4. Расчет пять раз статически неопределимой рамы	466
7.29. Арка. <i>Маллет</i>	467
7.30. Устойчивость	472
7.30.1. Критическая нагрузка. <i>Маллет</i>	472
7.30.2. Подбор сечения	475
7.31. Стабильность	477
7.31.1. Обыкновенное дифференциальное уравнение	477
7.31.2. Условие стабильности процесса на плоскости	479
7.31.3. Нестабильность уравнения в частных производных	481
7.31.4. Нестабильность сжатого стержня в условии ползучести	482
Приложение 1. Геометрические характеристики сечений	485
Приложение 2. Допускаемые напряжения и коэффициент продольного изгиба φ	490
Список литературы	491
Предметный и именной указатель	494

Предисловие

Теоретическая механика и сопротивление материалов тесно связаны друг с другом. Продуманная аксиоматика, ясные и строгие уравнения и теоремы, положенные в основу теоретической механики, создают хорошую базу для прикладных задач сопротивления материалов. Решения задач механики не всегда стандартны. В любом случае для решения новых задач, которые ставит быстро развивающаяся техника, полезен опыт решения классических задач. В данной книге представлены как широко известные задачи механики, так и новые. В последнее время в сложившейся схеме решения задач механики появилось новое направление, основанное на применении систем компьютерной алгебры, работающих с аналитическими вычислениями. Обзор систем можно найти в [18]. Отметим только две основные системы: **Mathematica** и **Maple**.

Выбор в пользу **Maple** автор сделал не потому, что она в чем-то лучше или удобнее других систем. Просто это первая (после **REDUCE**) система аналитических вычислений, с которой ему посчастливилось познакомиться еще в 1990-х годах. Тогда это была студенческая версия **Maple**, записанная на нескольких гибких 5-дюймовых дисках без какой-либо документации. Возможно, система **Mathematica** (особенно ее последние версии) выигрывают в некоторых случаях у **Maple**, например по части решения дифференциальных уравнений. Однако, в общем, это приблизительно равные по возможностям системы, и выбор одной из них — это дело вкуса.

Сейчас уже трудно представить себе решение сколько-нибудь серьезной задачи механики без компьютерных вычислений. Применение системы компьютерной математики в одних случаях просто позволяет избежать рутинной части работы, сосредоточившись на содержании задачи и на поиске новых методов ее решения, в других — является единственным инструментом для решения задачи. Кроме того, системы компьютерной математики дают хороший иллюстративный материал к условиям и решениям в виде рисунков, графиков и анимированных изображений.

В книге приведены примеры решений задач по статике, кинематике, динамике и сопротивлению материалов.

Основы работы в **Maple**, описания специальных пакетов (линейная алгебра, дифференциальные уравнения, графика) даны в главе 5. Эту главу можно считать самостоятельным учебником **Maple**, рассчитанным не только на начинающих пользователей, но и на профессионалов, работающих с математическими пакетами. Все операторы и команды сведены в подробный алфавитный указатель, содержащий более 1600 наименований.

Большая часть задач предназначена для подготовки к экзаменам и зачетам. Некоторые могут быть частью научных разработок или технических расчетов. В главе 6 описано программирование графического интерфейса решений Maple, так называемых маплетов. В системе Maple изначально широко используется программирование алгоритмов и вычисление выражений. Программирование маплетов существенно отличается от привычных структур и правил языка в Maple, поэтому освоение маплетов потребует некоторых усилий.

В последней главе содержатся программы и маплеты для Maple 13. Дан разбор особенностей применения различных операторов и команд системы, рекомендации по программированию. Почти все программы работоспособны в более ранних версиях Maple, а используемые алгоритмы могут переноситься и на другие системы (Mathcad, Mathematica, MATLAB).

Архив всех программ из книги можно найти на сайте автора по адресу <http://vuz.exponenta.ru/progr.rar>.

Книга написана на основе опыта преподавания теоретической механики и математических моделей естествознания в МЭИ(ТУ).

Автор благодарен Выльевой С. В. за внимательное прочтение рукописи и Горшкову П. В. за создание мультимедийного курса «Теоретическая механика. Статика» на основе задач из этой книги (www.MediaShell.ru). Отдельная благодарность профессору Хроматову В. Е., сделавшему ряд замечаний по книге.

Все замечания о книге автор с благодарностью примет по адресу: mpri2004@yandex.ru.

Глава 1

СТАТИКА

Статика — один из трех основных разделов теоретической механики. В статике изучается равновесие тел под действием сил и свойства систем сил, необязательно находящихся в равновесии.

Для решения задач статики потребуются понятия проекции силы на ось и момента силы относительно точки и оси. Напомним, что проекция вектора силы \vec{F} на ось x определяется по формуле $F_x = F \cos \alpha$, где α — угол между положительным направлением оси и вектором силы, отсчитываемый против часовой стрелки. Если угол острый, то проекция положительная, если тупой — отрицательная. Если сила перпендикулярна оси, то ее проекция на эту ось равна нулю. Проекция силы, параллельной оси, равна F , если сила и ось направлены в одну сторону ($\alpha = 0$), и $-F$, если в разные стороны ($\alpha = 180^\circ$).

Общее определение момента \vec{M}_O силы \vec{F} относительно точки O дается векторным произведением¹

$$\vec{M}_O(\vec{F}) = \vec{r}_0 \times \vec{F} = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ x & y & z \\ F_x & F_y & F_z \end{vmatrix}, \quad (1.1)$$

где $\vec{r}_0 = x\vec{i} + y\vec{j} + z\vec{k}$ — радиус-вектор точки приложения вектора силы относительно точки O , \vec{i} , \vec{j} , \vec{k} — орты осей координат. Модуль момента вычисляем по формуле $M_O(\vec{F}) = r_0 F \sin \gamma$, где γ — угол между векторами \vec{r}_0 и \vec{F} . Направление вектора момента вычисляется по правилу векторного произведения. Плечо h силы относительно точки O — это кратчайшее расстояние от точки до линии действия силы; $h = r_0 \sin \gamma$.

Вектор момента перпендикулярен плоскости, в которой располагаются силы. Поэтому в задачах статики плоской системы сил момент можно рассматривать как скалярную величину — величину проекции вектора момента на нормаль к плоскости (ось z). Индекс z для сокращения записи часто опускают и отождествляют момент силы

¹Векторное произведение иногда обозначается квадратными скобками $[\vec{r}_0, \vec{F}]$. В системе **Maple** векторное произведение — оператор `CrossProduct` из пакета `LinearAlgebra` или `crossprod` из пакета `linalg`.



M_O относительно точки на плоскости со скалярной величиной — M_{Oz} . Отсюда вытекает практическое правило определения момента силы относительно точки в плоских задачах статики. Для вычисления момента силы относительно точки O (рис. 1) сначала находим проекции силы на оси, а затем, согласно (1.1), момент вычисляем по формуле $M_{Oz}(F) = -F_x \cdot y_0 + F_y \cdot x_0$. Другой способ вычисления момента: $M_{Oz}(F) = \pm Fh$, где h — плечо силы относительно точки O .

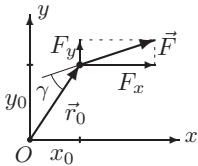


Рис. 1

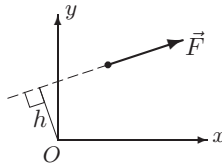


Рис. 2

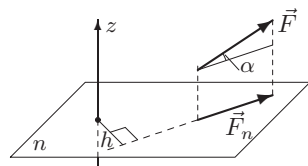










Рис. 3

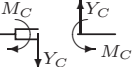
Знак определяется по правилу векторного произведения. Если сила поворачивает тело относительно центра по часовой стрелке — момент отрицательный, против часовой стрелки — положительный. На рис. 2 момент силы \vec{F} относительно точки O отрицательный: $M_{Oz} < 0$. Если сила или линия ее действия пересекает точку, то момент силы относительно этой точки равен нулю.

При решении задач пространственной статики требуется вычислять момент силы относительно оси, или, что то же, проекцию момента силы относительно точки (1.1) на ось, проходящую через нее. Иногда эту величину удобнее искать как момент проекции \vec{F}_n силы на плоскость, перпендикулярную оси, относительно точки пересечения оси с плоскостью (рис. 3). Знак определяем по направлению вращения вокруг оси с точки зрения наблюдателя, находящегося на конце оси. Если вращение происходит по часовой стрелке, то момент отрицательный, против часовой стрелки — положительный. Момент силы относительно оси равен нулю, если сила параллельна оси или пересекает ее, т. е. если сила и ось лежат в одной плоскости.

Кроме сил, в статике рассматриваются и пары сил. Пара — это совокупность двух равных параллельных противоположно направленных сил. Пара характеризуется моментом — суммой моментов ее сил относительно некоторой точки. Легко показать, что положение точки не существенно и на момент не влияет, поэтому момент пары является свободным вектором. Напомним, что вектор силы является скользящим вектором. В зависимости от знака момента пары на плоскости изображать пару будем изогнутой стрелкой \curvearrowright или \curvearrowleft . Вектор пары перпендикулярен ее плоскости.

Для решения задач о равновесии тел или системы тел необходимо выделить тело, равновесие которого изучается. Связи заменяем их

реакциями. Неподвижный шарнир  имеет две реакции  , подвижный  — одну  . Заделка  или  дает три реакции (две силы и момент):  . Если части рамы соединены скользящей заделкой вида  , то реакциями частей являются момент M_C

и сила Y_C :  , направленные в противоположные стороны.

Решение задач статики в системе Maple¹ приведено в последней главе. Большинство задач статики сводится к системе линейных уравнений. Рутинную часть работы по составлению и решению уравнений можно поручить Maple. Простейшая программа может выглядеть, например, так:

```
уравнение1:=X[a]*4.5+Y[a]*2=1:
уравнение2:=-X[a]*6+Y[a]*3-22:
solve({уравнение1, уравнение2}, {X[a], Y[a]});
```

Использован оператор `solve` для решения уравнений и систем уравнений². В первом аргументе в фигурных или квадратных скобках указаны уравнения системы, во втором — искомые неизвестные. В обозначениях переменных допускаются русские буквы. Если правая часть не обозначена (как во втором уравнении), то по умолчанию она принимается нулем. Более того, в операторе `solve` здесь можно и не указывать список неизвестных `{X[a], Y[a]}`, так как других переменных в системе нет. Как и в языке Pascal, присваивание обозначено двоеточием и равенством `:=`.

Решение задач в Maple имеет несколько преимуществ. Во-первых, компьютер выполняет математические действия, часто весьма громоздкие. Во-вторых, решение хорошо организовано, уравнение легко поправить и сразу же пересчитать, если найдена ошибка при составлении уравнения и ответ не сходится. В-третьих, работу удобно оформить, распечатав ее на принтере. Текст программы Maple конвертируется в простой текстовый файл и в форматы HTML, RTF, LaTeX. В меню **File** интерфейса для этого есть команда **Export As**.

Программу для Maple можно оформить в виде маплета с пользовательским интерфейсом, защищенным от случайного изменения кода. Об этом пойдет речь в гл. 6, с. 304.

¹ Демонстрационную бесплатную версию Maple VR4 можно найти по адресу <http://vuz.exponenta.ru/PDF/DNLD/MVR4DEMO.rar>

² Подробнее про оператор `solve` см. с. 241.

1.1. Рамы

Задача 1. Плоская рама закреплена в шарнире A и опирается на подвижный шарнир B (рис. 4). На раму действуют силы $F = 10$ кН, $P = 2$ кН и момент $M = 34$ кНм. Определить реакции опор. Дано: $\cos \alpha = 0,8$.

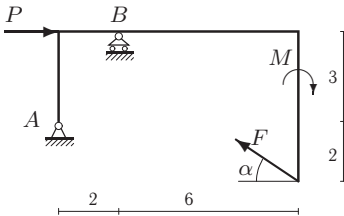


Рис. 4

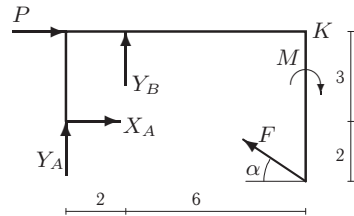


Рис. 5

Решение

Заменяем действие связей реакциями. В шарнире A возникают две реакции — горизонтальная сила X_A и вертикальная Y_A . Реакция подвижного шарнира B вертикальная (рис. 5).

Уравнения равновесия имеют вид

$$\begin{aligned} \sum X_i &= X_A - F \cos \alpha + P = 0, \\ \sum Y_i &= Y_A + Y_B + F \sin \alpha = 0, \\ \sum M_A &= 2Y_B - 2F \cos \alpha + 8F \sin \alpha - M - 3P = 0. \end{aligned} \quad (1.2)$$

Решаем систему уравнений и находим $Y_B = 4$ кН, $X_A = 6$ кН, $Y_A = -10$ кН. Вертикальная реакция Y_A получилась отрицательной. Это означает, что направление реакции противоположно выбранному. Для проверки решения составим сумму моментов всех сил, действующих на раму, включая найденные реакции, относительно произвольной точки, например, K :

$$\begin{aligned} \sum M_K &= -8Y_A - 6Y_B + 3X_A - 5F \cos \alpha - M = \\ &= 80 - 6 \cdot 4 + 3 \cdot 6 - 5 \cdot 10 \cdot 0,8 - 34 = 0. \end{aligned}$$

Сумма равна нулю. Решение найдено верно.

Maple-программа P3.mws для решения системы (1.2) дана на с. 339.

Задача 2. Плоская рама, состоящая из двух изогнутых под прямым углом стержней, имеет две опоры — скользящую заделку в точке A и неподвижный шарнир B . Части рамы соединены шарниром в точке C (рис. 6). Размеры даны в метрах. К раме приложена горизонтальная

сила $P = 1$ кН, вертикальная $F = 4$ кН и момент $M = 3$ кНм. Найти реакции опор.

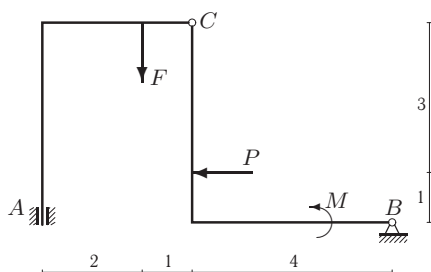


Рис. 6

Решение

Разобьем раму на две части по шарниру C . Отброшенные части заменим реакциями X_C , Y_C . На одну часть эти реакции действуют в одну сторону, на другую — в противоположную. Заделка A имеет две неизвестные реакции: момент M_A , препятствующий вращению, и горизонтальную силу X_A . Вертикальной реакции в точке A нет, так как стержень можно свободно вынуть из скользящей опоры. Действие неподвижного шарнира B заменяем горизонтальной реакцией X_B и вертикальной Y_B . Составим уравнения равновесия для левой части (рис. 7):

$$\begin{aligned} \sum X_i &= X_C + X_A = 0, \\ \sum Y_i &= Y_C - F = 0, \\ \sum M_A &= M_A - 4 \cdot X_C + 3 \cdot Y_C - 2 \cdot F = 0. \end{aligned} \quad (1.3)$$

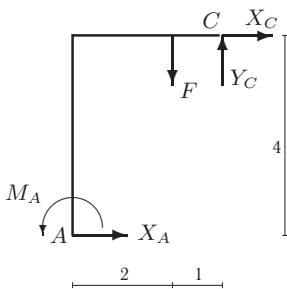


Рис. 7

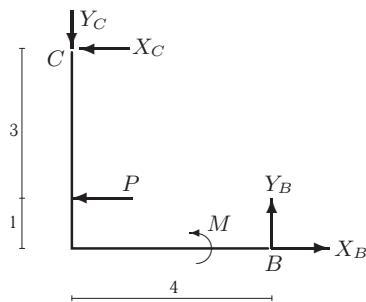


Рис. 8

Каждая из выделенных частей находится в равновесии. Составим уравнения равновесия для правой части рамы (рис. 8):

$$\begin{aligned}\sum X_i &= -X_C + X_B - P = 0, \\ \sum Y_i &= Y_B - Y_C = 0, \\ \sum M_B &= 4 \cdot Y_C + 4 \cdot X_C + 1 \cdot P + M = 0.\end{aligned}\tag{1.4}$$

Решаем систему уравнений (1.3)–(1.4), получаем $X_A = 5$ кН, $Y_B = 4$ кН, $M_A = -24$ кНм, $X_B = -4$ кН, $Y_C = 4$ кН, $X_C = -5$ кН.

Проверка. Составляем сумму моментов всех сил, действующих на раму, включая реакции опор, относительно какой-либо произвольной точки. Желательно, чтобы эта точка не лежала на линии действия реакций. Выберем точку K на правой части рамы¹ (рис. 9). Внутренние реакции сочленяющего шарнира C в уравнение не входят:

$$\begin{aligned}\sum M_K &= 4 \cdot Y_B - 1 \cdot P + M + M_A + 2 \cdot X_A + 2 \cdot X_B + 1 \cdot F = \\ &= 4 \cdot 4 - 1 \cdot 1 + 3 - 24 + 2 \cdot 5 - 2 \cdot 4 + 1 \cdot 4 = 0.\end{aligned}$$

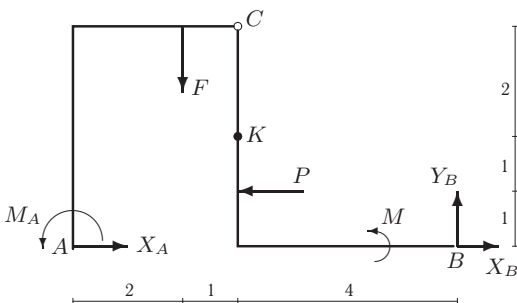


Рис. 9

Проверка выполнена. Реакции найдены верно.

Заметим, что если разбить раму не по шарниру, то в месте разбиения на каждую из образовавшихся частей, помимо двух взаимно перпендикулярных сил, надо будет приложить реактивный момент — еще одну неизвестную реакцию. В результате в задаче будет 7 неизвестных и прежние 6 уравнений. Так задачу решить нельзя.

Задача 3. Рама состоит из двух частей, соединенных скользящей заделкой в точке C . В точке A рама опирается на неподвижный шарнир, в точках B и D — на горизонтально подвижные шарниры. К раме

¹В общем случае точка может не принадлежать раме.

приложены силы $P = 15$ кН, $Q = 30$ кН и момент $M = 3$ кНм (рис. 10). Размеры даны в метрах, $\cos \alpha = 4/5$. Найти реакции опор.

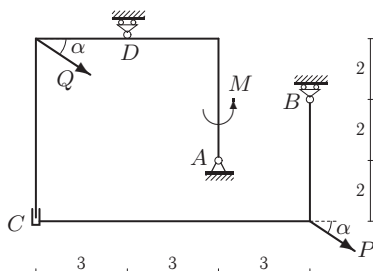


Рис. 10

Решение

Разобьем раму на две части по скользящей заделке C . Отброшенные части заменим реакциями — моментом M_C и горизонтальной силой X_C . Направления реакций заделки, приложенных к разным частям, взаимно противоположны¹. Отбросим внешние связи. Подвижные шарниры B и D заменим соответственно вертикальными реакциями Y_B и Y_D , неподвижный шарнир A — вертикальной реакцией Y_A и горизонтальной реакцией X_A .

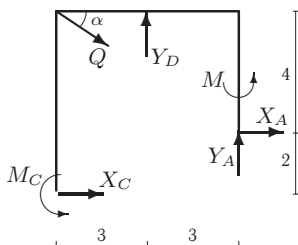


Рис. 11

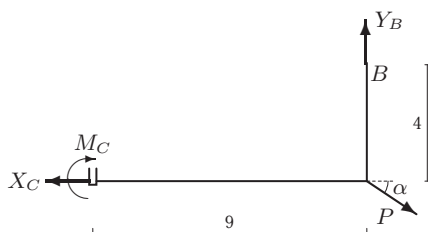


Рис. 12

¹ Не рекомендуем в подписях внутренних реакций на рисунке использовать знак вектора, например \vec{X}_C . Во-первых, это вынуждает ответную реакцию как-нибудь обозначать, например штрихом: \vec{X}'_C . При этом $\vec{X}'_C = -\vec{X}_C$. Появившийся знак минус обычно сбивает с толку, так как по ошибке часто пишут и $X'_C = -X_C$, что неправильно, так как модули этих сил равны, а взаимно противоположное направление уже учтено в рисунке. Во-вторых, если обозначать вектором реакции, то возникает вопрос, как обозначать реактивный момент? В действительности реактивный момент — это вектор, перпендикулярный плоскости, а на рисунках обозначено лишь направление поворота.

Обе части находятся в равновесии. Составим уравнения равновесия для части CA (рис. 11):

$$\sum X_i = X_C + Q \cos \alpha + X_A = 0, \quad (1.5)$$

$$\sum Y_i = Y_D + Y_A - Q \sin \alpha = 0, \quad (1.6)$$

$$\sum M_C = 6Y_A + 3Y_D - 2X_A - 6Q \cos \alpha + M_C + M = 0. \quad (1.7)$$

Составим уравнения равновесия для части CB (рис. 12):

$$\sum X_i = -X_C + P \cos \alpha = 0, \quad (1.8)$$

$$\sum Y_i = Y_B - P \sin \alpha = 0, \quad (1.9)$$

$$\sum M_C = 9Y_B - 9P \sin \alpha - M_C = 0. \quad (1.10)$$

Решаем систему уравнений (1.5)–(1.10), получаем $X_A = -36$ кН, $Y_A = 5$ кН, $Y_B = 9$ кН, $Y_D = 13$ кН, $X_C = 12$ кН, $M_C = 0$.

Заметим, что если уравнение моментов (1.10) составить относительно точки приложения силы P , то ответ $M_C = 0$ получится значительно проще.

Проверка. Рассмотрим равновесие рамы в целом (рис. 13). Приложим к раме найденные реакции опор и внешние силы. Составим уравнение равновесия — сумму моментов относительно точки K :

$$\sum M_K = 4X_A + 9Y_B + 6Y_A + 3Y_D + M + 6P \cos \alpha - 9P \sin \alpha = 0.$$

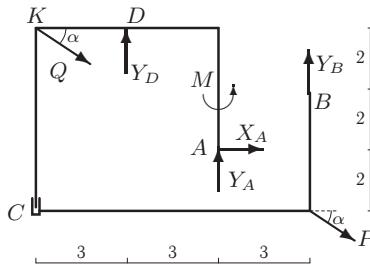


Рис. 13

Проверка выполнена, реакции найдены верно.

Задача 4. Рама, состоящая из двух частей, имеет две неподвижные шарнирные опоры. Части рамы соединены шарниром в точке C (рис. 14). На раму действует вертикальная сила $P = 3$ кН, горизон-

гальная $F = 4$ кН и момент $M = 8$ кНм. Размеры даны в метрах. Найти реакции опор.

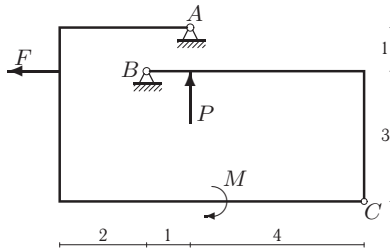


Рис. 14

Решение

Разобьем раму на две части по шарниру C . Отброшенные части заменим реакциями X_C , Y_C . При этом реакции, приложенные к одной части, направлены в одну сторону, к другой — в противоположную. Рассмотрим равновесие каждой части. Запишем три уравнения равновесия части AC (рис. 15):

$$\sum X_i = X_C + X_A - F = 0, \quad (1.11)$$

$$\sum Y_i = Y_C + Y_A = 0, \quad (1.12)$$

$$\sum M_C = -4Y_A - 4X_A + 3F - M = 0. \quad (1.13)$$

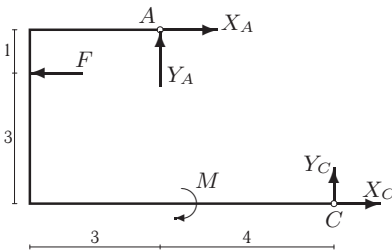


Рис. 15

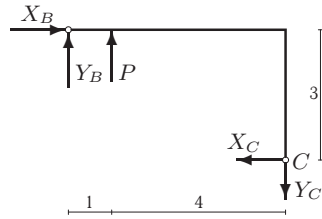


Рис. 16

Уравнения равновесия части CB (рис. 16) имеют вид

$$\sum X_i = -X_C + X_B = 0, \quad (1.14)$$

$$\sum Y_i = Y_B - Y_C + P = 0, \quad (1.15)$$

$$\sum M_C = -5Y_B - 3X_B - 4P = 0. \quad (1.16)$$

Решаем систему шести уравнений с шестью неизвестными. Получаем реакции: $X_A = -2$ кН, $Y_A = 3$ кН, $X_C = 6$ кН, $Y_C = -3$ кН, $Y_B = -6$ кН, $X_B = 6$ кН.

Проверка. Составляем сумму моментов всех сил, действующих на раму, включая реакции опор, относительно какой-либо произвольной точки, например точки K (рис. 17):

$$\begin{aligned}\sum M_K &= 3Y_A - 4X_A + 2Y_B - 3X_B + 3F + 3P - M = \\ &= 3 \cdot 3 + 4 \cdot 2 - 2 \cdot 6 - 3 \cdot 6 + 3 \cdot 4 + 3 \cdot 3 - 8 = 0.\end{aligned}$$

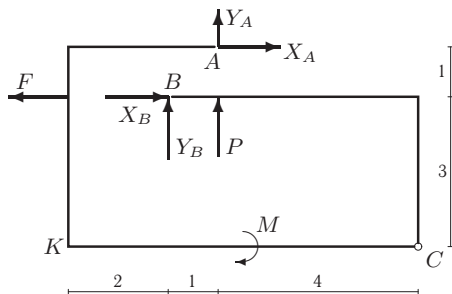


Рис. 17

Проверка выполнена. Реакции найдены верно.

Если по условию задачи требуется найти только реакции опор, а реакции внутреннего сочленяющего шарнира искать не надо, то нет необходимости составлять все шесть уравнений равновесия. Трехшарнирная схема рамы¹ позволяет составить две отдельные системы по два уравнения в каждой. Для этого надо дополнительно рассмотреть равновесие всей рамы в целом, не расчленяя ее на части. Записываем два уравнения моментов (рис. 17):

$$\sum M_A = -1 \cdot Y_B + 1 \cdot X_B - 1 \cdot F - M = 0, \quad (1.17)$$

$$\sum M_B = 1 \cdot Y_A - 1 \cdot X_A + 1 \cdot P - M = 0. \quad (1.18)$$

Уравнения (1.16) и (1.17) образуют замкнутую систему относительно неизвестных X_B и Y_B , а уравнения (1.13) и (1.18) — систему для X_A и Y_A . При таком подходе решение существенно упрощается, хотя все расчетные уравнения — уравнения моментов². Так как при решении уравнения проекций не использовались, в качестве проверки можно составить уравнения проекций для всей рамы в целом.

¹ Два опорных шарнира и один сочленяющий.

² Заметим, что обе полученные системы имеют один и тот же определитель, равный удвоенной площади треугольника ABC .

Задача 5. Конструкция состоит из двух изогнутых под прямым углом стержней и шарнирного треугольника. На конструкцию действуют силы $F_1 = 7$ кН, $F_2 = 2$ кН, $P = 4$ кН и момент $M = 3$ кНм. Конструкция опирается на неподвижные шарниры в точках B и C и вертикальный стержень в A (рис. 18). Найти реакции опор.

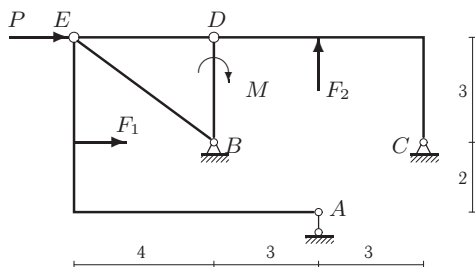


Рис. 18

Решение

Разделим конструкцию по шарнирам на три части и рассмотрим равновесие каждой из них. Действие опор заменим их реакциями. Реакции внутренних шарниров, соединяющих части, приложим к каждой из частей во взаимно противоположных направлениях¹. Силу P , действующую одновременно и на часть AE , и на часть BDE , отнесем произвольно к любому из тел, например AE (рис. 19). Внешние реакции не зависят от выбора части, к которой приложена P , более того, можно даже разбить силу на две — одну половину приложить к одной части, другую — к другой. Рассмотрим равновесие части AE . Действие вертикального опорного стержня в A заменим вертикальной реакцией Y_A . Внутренний шарнир E соответствует двум реакциям X_E и Y_E . Запишем уравнения равновесия части AE :

$$\begin{aligned} \sum X_k &= X_E + F_1 + P = 0, \\ \sum Y_k &= Y_A + Y_E = 0, \\ \sum M_E &= 7Y_A + 3F_1 = 0. \end{aligned} \quad (1.19)$$

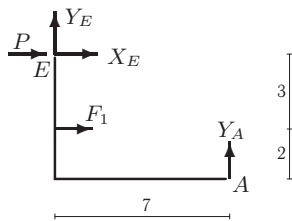


Рис. 19

¹Если в шарнире соединены два тела, то выбор взаимно противоположных направлений очевиден. А что делать, если шарниром соединены три тела? Ответ на этот вопрос дан в решении задачи на с. 21.

Рассмотрим равновесие других частей конструкции. Выделим треугольник BDE (рис. 20). Приложим внешний момент и реакции опор. Реакции X_E и Y_E направим в стороны, противоположные этим же реакциям на рис. 19.

Запишем уравнения равновесия части BDE (рис. 20):

$$\begin{aligned}\sum X_k &= X_B - X_E + X_D = 0, \\ \sum Y_k &= Y_B + Y_D - Y_E = 0, \\ \sum M_E &= 3X_B + 4Y_B + 4Y_D - M = 0.\end{aligned}\tag{1.20}$$

Рассмотрим равновесие части DC (рис. 21). Реакции X_D , Y_D направляем в стороны, противоположные изображенным на рис. 20.

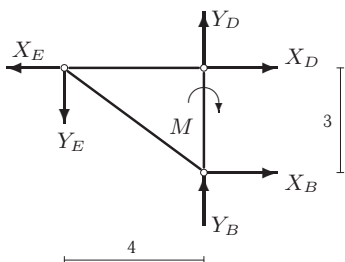


Рис. 20

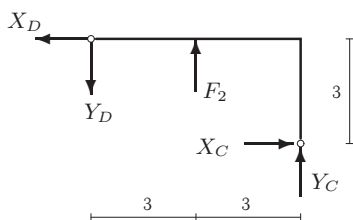


Рис. 21

Запишем уравнения равновесия части DC :

$$\begin{aligned}\sum X_k &= -X_D + X_C = 0, \\ \sum Y_k &= Y_C - Y_D + F_2 = 0, \\ \sum M_D &= 6Y_C + 3F_2 + 3X_C = 0.\end{aligned}\tag{1.21}$$

Решаем систему девяти уравнений (1.19)–(1.21). Получаем значения искомых реакций: $Y_A = -3$ кН, $X_B = -3$ кН, $Y_B = -2$ кН, $X_C = -8$ кН, $Y_C = 3$ кН, $X_D = -8$ кН, $Y_D = 5$ кН, $Y_E = 3$ кН, $X_E = -11$ кН.

Проверка. Запишем сумму моментов относительно точки E всех сил, приложенных к раме, включая реакции опор (рис. 22):

$$\begin{aligned}\sum M_E &= 7Y_A + 3F_1 + 3X_B + 4Y_B - M + 7F_2 + 3X_C + 10Y_C = \\ &= -21 + 21 - 9 - 8 - 3 + 14 - 24 + 30 = 0.\end{aligned}$$

Реакции внутренних шарниров приложены к отдельным частям рамы; в условие равновесия всей рамы в целом они не входят, и

проверить таким образом их нельзя. При необходимости можно выполнить проверку, выделив из рамы не три, а две части (две возможные комбинации: $AE + BDE$ и CD, AE и $BDE + CD$).

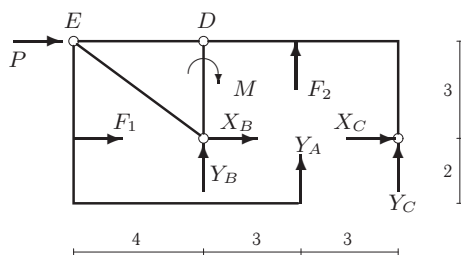


Рис. 22

Проверка выполнена. Реакции найдены правильно. Заметим, что проверка равенства нулю суммы проекций всех сил, приложенных к раме в целом, на ось x или y не является эффективной. Фактически эти суммы будут состоять из сумм уравнений для проекций из систем (1.19), (1.20), (1.21), и вероятные ошибки в уравнениях моментов никак не повлияют на равенство нулю в такой проверке.

Решение задачи в системе **Maple** приведено на с. 439.

Задача 6. Конструкция состоит из трех тел, соединенных шарниром D и стержнем 1 (рис. 23). Конструкция нагружена моментом $M = 102$ кНм и двумя силами: $F_1 = F_2 = 17$ кН. Размеры даны в метрах. Найти реакции опор и усилие в стержне 1.

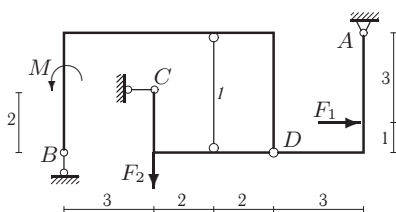


Рис. 23

Решение

Разбиваем конструкцию на три части, отбрасывая стержень 1 и шарнир D . Освобождаем конструкцию от внешних связей, заменяя их реакциями. Условно считаем, что в точке D часть DA взаимодействует сразу с двумя частями — BD и CD . Реакции от действия части BD на DA обозначим X'_D, Y'_D , а силы взаимодействия DA и CD — X''_D, Y''_D . Рассматриваем равновесие каждой из частей. Всего получается 9

неизвестных реакций. Записываем по три уравнения равновесия (два уравнения проекций и уравнение моментов относительно произвольной точки) для каждой из частей.

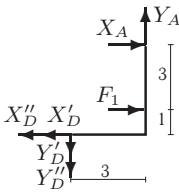


Рис. 24

К точке D приложены две горизонтальные и две вертикальные реакции. Уравнения равновесия части AD :

$$\begin{aligned}\sum X &= -X'_D - X''_D + F_1 + X_A = 0, \\ \sum Y &= -Y'_D - Y''_D + Y_A = 0, \\ \sum M_D &= 3Y_A - 4X_A - 1 \cdot F_1 = 0.\end{aligned}\quad (1.22)$$

Неизвестную реакцию стержня направляем в предположении, что стержень растянут. Уравнения равновесия части CD :

$$\begin{aligned}\sum X &= R_C + X''_D = 0, \\ \sum Y &= Y'_D + S - F_2 = 0, \\ \sum M_D &= -2S - 2R_C + 4F_2 = 0.\end{aligned}\quad (1.23)$$

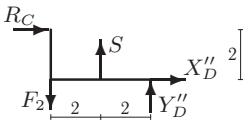


Рис. 25

Уравнения равновесия части BD :

$$\begin{aligned}\sum X &= X'_D = 0, \\ \sum Y &= R_B + Y'_D - S = 0, \\ \sum M_B &= -5S + 7Y'_D + M = 0.\end{aligned}\quad (1.24)$$

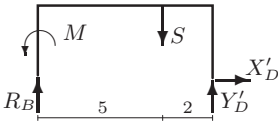


Рис. 26

Последовательно исключая неизвестные, решаем систему уравнений (1.22), (1.23), (1.24). Получаем $X'_D = 0$, $Y'_D = 14$ кН, $Y''_D = -23$ кН, $X_A = -11$ кН, $X''_D = 6$ кН, $Y_A = -9$ кН, $S = 40$ кН, $R_B = 26$ кН, $R_C = -6$ кН.

Проверка. Составляем сумму моментов всех сил, действующих на всю конструкцию в целом, включая найденные реакции, относительно

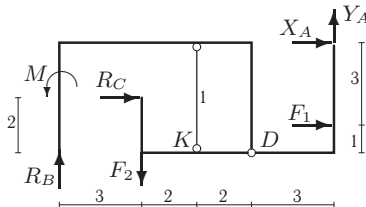


Рис. 27

произвольной точки. Для того чтобы одновременно проверить все ответы, желательно выбрать точку, не лежащую на линиях действия

реакций. Внутренние реакции в шарнире D и усилии S в стержне 1 в уравнение не входят. Сумма моментов относительно точки K имеет следующий вид:

$$\begin{aligned}\sum M_K &= M - 5R_B - 2R_C + 2F_2 - 1F_1 - 4X_A + 5Y_A = \\ &= 102 - 5 \cdot 26 + 2 \cdot 6 + 2 \cdot 17 - 1 \cdot 17 + 4 \cdot 11 - 5 \cdot 9 = 0.\end{aligned}$$

Проверка выполнена.

Задача 7. Конструкция состоит из трех тел, соединенных шарниром D и стержнем 1 (рис. 28) и нагружена моментом $M = 3$ кНм, и тремя силами $F_1 = 2$ кН, $F_2 = 4$ кН, $F_3 = 2$ кН. Размеры даны в метрах. Найти реакции опор и усилии в стержне 1.

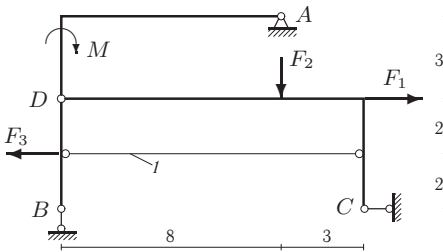


Рис. 28

Решение

Решение этой задачи отличается от предыдущей способом представления сил, действующих на сочленяющий шарнир D . При этом увеличивается число неизвестных, но структура связей становится более ясной.

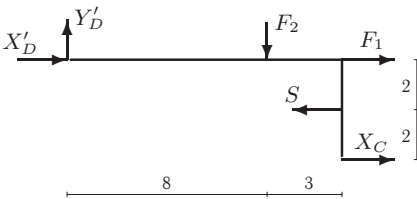


Рис. 29

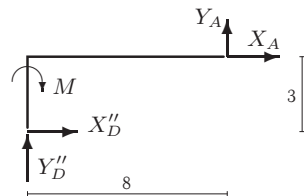


Рис. 30

Разбиваем конструкцию на части, отбрасывая стержень 1 и шарнир D . Освобождаем конструкцию от внешних связей, заменяя их реакциями. Реакции опорных стержней в точках B и C направлены вдоль стержней. Шарнир A имеет две реакции. В число тел, составляющих конструкцию, вводим еще одно тело — ось D . Считаем, что в точке D

все три части конструкции независимо друг от друга взаимодействуют с осью, их соединяющей. Реакции обозначим $X'_D, Y'_D, X''_D, Y''_D, X'''_D, Y'''_D$. Рассматриваем равновесие каждой из частей и равновесие оси. Всего получается 11 неизвестных реакций. Записываем по три уравнения равновесия (два уравнения проекций и уравнение моментов относительно произвольной точки) для каждой из частей и два уравнения равновесия оси в проекциях.

Уравнения равновесия части DC (рис. 29):

$$\begin{aligned}\sum X &= X'_D + F_1 - S + X_C = 0, \\ \sum Y &= Y'_D - F_2 = 0, \\ \sum M_D &= 4X_C - 2S - 8F_2 = 0.\end{aligned}\quad (1.25)$$

Уравнения равновесия части AD (рис. 30):

$$\begin{aligned}\sum X &= X''_D + X_A = 0, \\ \sum Y &= Y''_D + Y_A = 0, \\ \sum M_D &= 8Y_A - 3X_A - M = 0.\end{aligned}\quad (1.26)$$

Усилия S от действия внутренней связи, приложенные к частям BC и DC , взаимно противоположны. Уравнения равновесия части BD (рис. 31):

$$\begin{aligned}\sum X &= X'''_D + S - F_3 = 0, \\ \sum Y &= Y'''_D + Y_B = 0, \\ \sum M_B &= -4X'''_D - 2S + 2F_3 = 0.\end{aligned}\quad (1.27)$$

Все реакции прикладываем к оси в противоположную сторону. Для сходящейся системы сил на плоскости достаточно двух уравнений равновесия (рис. 32):

$$\begin{aligned}\sum X &= -X'_D - X''_D - X'''_D = 0, \\ \sum Y &= -Y'_D - Y''_D - Y'''_D = 0.\end{aligned}\quad (1.28)$$

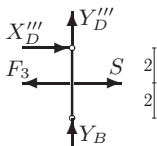


Рис. 31

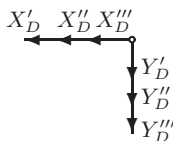


Рис. 32

Решаем систему 11 уравнений с 11 неизвестными (1.25)–(1.28). Получаем искомые реакции: $X_A = -9$ кН, $Y_A = -3$ кН, $Y_B = 7$ кН, $X_C = 9$ кН, $S = 2$ кН, $X'_D = -9$ кН, $Y'_D = 4$ кН, $X''_D = 9$ кН, $Y''_D = 3$ кН, $X'''_D = 0$, $Y'''_D = -7$ кН.

Проверка. Составим сумму моментов всех внешних сил, приложенных к нерасчлененной системе, относительно точки K (рис. 33):

$$\begin{aligned}\sum M_K &= -3X_A - 3Y_A + 3F_2 - 2F_3 + 4X_C - 11Y_B - M = \\ &= 27 + 9 + 12 - 4 + 36 - 77 - 3 = 0.\end{aligned}$$

Проверка выполнена. Решение найдено верно. Заметим, что эту задачу можно решить проще, составив всего пять, а не одиннадцать

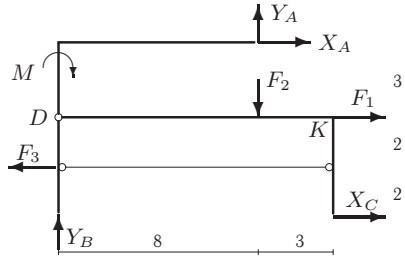


Рис. 33

уравнений. Первые два уравнения — система уравнений для реакций Y_B и X_C . Одно из них — сумма моментов всех сил, приложенных к целой системе относительно опоры A , другое — сумма моментов всех сил, приложенных к частям BD и DC относительно D . Остальные уравнения — уравнения проекций для всей системы, из которых легко находим две реакции опоры A . Для определения S можно использовать уравнение моментов системы (1.25).

Решение задачи в системе **Maple** дано на с. 340.

Задача 8. Плоская рама, состоящая из двух частей, закреплена в неподвижном шарнире B и опирается на подвижный шарнир A (рис. 34). Части рамы соединены шарниром C и двумя нитями. На

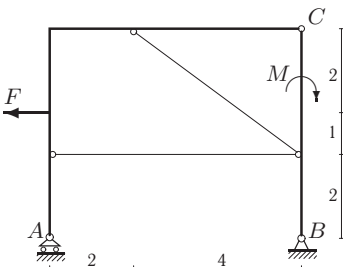


Рис. 34

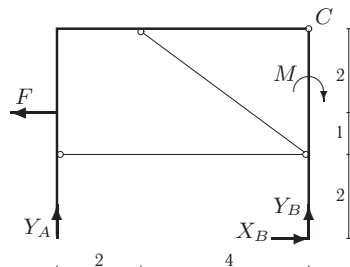


Рис. 35

раму действует сила $F = 4$ кН и момент $M = 2$ кНм. Определить наименьшее значение реакции внутреннего шарнира C , при котором рама находится в равновесии. Найти соответствующие реакции опор.

Решение

Прежде чем решать эту задачу, заметим странную для статики ситуацию. В задаче семь неизвестных реакций (четыре внутренних и три внешних), а для конструкции из двух тел можно составить только шесть независимых уравнений равновесия. Более того, задача усложняется из-за односторонних связей — натянутых нитей. Нити могут быть только растянуты. На первый взгляд решить такую задачу методами статики невозможно¹. В действительности же в условии скрыто дополнительное уравнение — минимизация реакции во внутреннем шарнире. Это позволяет из множества решений, допускаемых недоопределенной системой, выбрать единственное, соответствующее условию минимума реакции. При этом надо следить, чтобы в полученном решении реакции соответствовали натяжению нитей.

Заменяем действие связей реакциями. В неподвижной опоре B возникают две реакции — горизонтальная сила X_B и вертикальная Y_B . Реакция подвижного шарнира A вертикальная (рис. 35). Из условия равновесия рамы получаем три уравнения для трех неизвестных внешних реакций:

$$\sum X_i = X_B - F = 0, \quad (1.29)$$

$$\sum Y_i = Y_A + Y_B = 0, \quad (1.30)$$

$$\sum M_A = 6Y_B + 3F - M = 0. \quad (1.31)$$

Решаем систему уравнений и находим $Y_A = 5/3$ кН, $Y_B = -5/3$ кН, $X_B = 4$ кН. Эти реакции не зависят от натяжений нитей.

Для определения реакции шарнира C разобьем раму на две части. Действие частей друг на друга в точке C заменим реакциями X_C , Y_C . При этом реакции, приложенные к одной части, направлены в одну сторону, к другой — в противоположную. Нити предполагаются натянутыми. Натяжения направим вдоль нитей по направлению *от*

¹Системы, в которых число неизвестных больше, чем число уравнений статики, называются статически неопределимыми. В теоретической механике рассматриваются абсолютно твердые тела. В сопротивлении материалов и строительной механике, где учитываются деформации тел, существуют методы расчета таких систем, в частности, метод сил (с. 173).

точки крепления. Рассмотрим равновесие части AC (рис. 36):

$$\sum X_i = X_C + S_2 + S_1 \cos \alpha - F = 0, \quad (1.32)$$

$$\sum Y_i = Y_C + Y_A - S_1 \sin \alpha = 0, \quad (1.33)$$

$$\sum M_C = 3S_2 + 4S_1 \sin \alpha - 6Y_A - 2F = 0. \quad (1.34)$$

Здесь $\sin \alpha = 3/5$, $\cos \alpha = 4/5$.

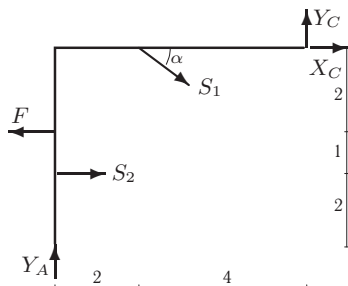


Рис. 36

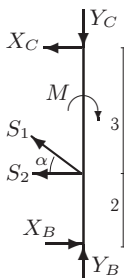


Рис. 37

Система шести уравнений (1.29)–(1.34) содержит семь неизвестных. Единственного решения такая система не имеет. Выберем неизвестную S_1 за переменный параметр. На этот параметр имеем ограничение $S_1 > 0$. Аналогично $S_2 > 0$. Нити должны быть натянуты. Из уравнения (1.34), в которое подставим найденное значение $Y_A = 5/3$ кН и заданную силу $F = 4$ кН, имеем

$$S_2 = 6 - (4/5)S_1. \quad (1.35)$$

С учетом найденного значения Y_A уравнение (1.33) дает $Y_C = -5/3 + (3/5)S_1$. Из уравнения (1.32) получаем $X_C = -2$ кН. Реакция в шарнире имеет вид

$$R_C = \sqrt{X_C^2 + Y_C^2} = \sqrt{4 + (-5/3 + (3/5)S_1)^2}. \quad (1.36)$$

Приравняв нулю производную $dR_C/dS_1 = 0$, получаем уравнение для усилия нити 1, при котором реакция в шарнире C минимальная: $S_1 = 25/9$ кН. Необходимо проверить знак усилия в другой нити. Согласно (1.35) получаем $S_2 = 34/9$ кН > 0 . Таким образом, решение найдено верно. Определяем окончательно реакцию: $Y_C = 0$, $R_C = |X_C| = 2$ кН.

Уравнения равновесия части AB (рис. 37) можно использовать как проверочные:

$$\sum X_i = -X_C + X_B - S_2 - S_1 \cos \alpha = 2 + 4 - \frac{34}{9} - \frac{20}{9} = 0,$$

$$\sum Y_i = -Y_C + Y_B + S_1 \sin \alpha = -\frac{5}{3} + \frac{25}{9} \cdot \frac{3}{5} = 0,$$

$$\sum M_B = 5X_C + 2S_2 + 2S_1 \cos \alpha - M = -10 + 2 \cdot \frac{34}{9} + 2 \cdot \frac{25}{9} \cdot \frac{4}{5} - 2 = 0.$$

Решение этой задачи в системе **Maple** с графической интерпретацией дано на с. 341¹.

1.2. Трение

Задача 9. Механическая система состоит из невесомого уголка, невесомого стержня и цилиндра весом $P = 5$ Н, радиусом $R = 1$ м (рис. 38). Стержень, ось цилиндра и уголок соединены шарнирно.

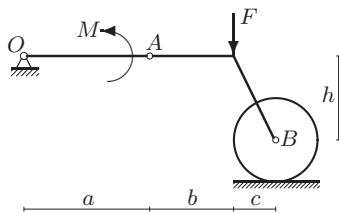


Рис. 38

Цилиндр может катиться без проскальзывания с трением качения $\delta = 5$ см. К уголку приложена вертикальная сила $F = 10$ Н. Дано: $a = 3$ м, $b = h = 2$ м, $c = 1$ м. В каких пределах меняется момент M при условии, что система находится в равновесии?

Решение

Система представляет собой механизм, находящийся в равновесии, но он может прийти в движение. Если сила F окажется достаточно велика, то момент M не удержит систему, и цилиндр покатится направо. И, наоборот, возможно движение налево, если момент пересилит остальные нагрузки и момент трения качения.

¹См. также: Зацепин М. Ф., Капустина О. М. Решение задач статики системы с неударивающими связями с помощью Mathematica — М.: МЭИ, 2006., <http://vuz.exponenta.ru/PDF/book/StKapZac.pdf>

Разобьем систему на части по шарнирам. Исследуемая система состоит из трех тел (рис. 39, 40, 41). Действие шарниров заменим их реакциями, не забывая о противоположном направлении реакций контактирующих тел. Зададим направление предполагаемого движения. Найдем предельное значение момента M , при котором цилиндр будет катиться *налево*. Момент трения качения, удерживающий цилиндр от поворота, будет направлен при этом по часовой

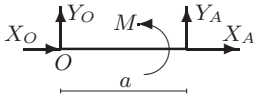


Рис. 39

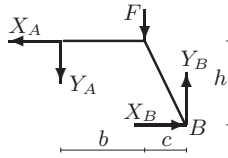


Рис. 40

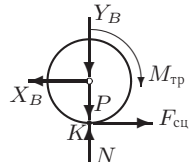


Рис. 41

стрелке. Для предельного случая (мгновение перехода от покоя к движению) момент трения качения будет

$$M_{\text{тр}} = N\delta. \quad (1.37)$$

Уравнение моментов для сил, действующих на стержень AO , относительно шарнира O имеет вид (рис. 39)

$$\sum M_O = Y_A \cdot a + M = 0. \quad (1.38)$$

Для уголка AB (рис. 40) записываем три уравнения равновесия:

$$\begin{aligned} \sum X_i &= -X_A + X_B = 0, \\ \sum Y_i &= -Y_A - F + Y_B = 0, \\ \sum M_B &= X_A \cdot h + Y_A \cdot (b + c) + F \cdot c = 0. \end{aligned} \quad (1.39)$$

Для цилиндра радиусом R , избегая включение в число неизвестных силы F_{cu} , составляем уравнения проекций на нормаль к плоскости и суммы моментов относительно точки касания K (рис. 41):

$$\begin{aligned} \sum Y_i &= N - P - Y_B = 0, \\ \sum M_K &= R \cdot X_B - M_{\text{тр}} = 0. \end{aligned} \quad (1.40)$$

Решаем систему семи уравнений (1.37)–(1.40) с семью неизвестными X_B , Y_B , X_A , Y_A , N , $M_{\text{тр}}$, M и получаем выражение момента через коэффициент трения качения δ и реакцию опоры:

$$M = a(FRc + h\delta(F + P))/(R(b + c) + h\delta), \quad (1.41)$$

$$N = R(P(b + c) + bF)/(R(b + c) + h\delta). \quad (1.42)$$

Подставляем заданные значения: $P = 5$ Н, $F = 10$ Н, $R = 1$ м, $a = 3$ м, $b = h = 2$ м, $c = 1$ м, и вычисляем

$$M = 30(3\delta + 1)/(2\delta + 3). \quad (1.43)$$

При $\delta = 0,05$ получаем $M = M_1 = 11,13$ Нм. Для того чтобы проверить возможность качения цилиндра в противоположную сторону, т. е. *направо*, необходимо направить $M_{\text{тр}}$ против часовой стрелки, составить систему уравнений равновесия и решить ее относительно M . Однако легко заметить, что эта система будет отличаться от уже решенной только знаком перед коэффициентом трения качения δ , следовательно, ее решение получается из предыдущего формальной заменой в (1.41) знака у δ . Таким образом, подставляя в (1.43) $\delta = -0,05$, получаем $M = M_2 = 8,79$ Нм. В итоге получаем область изменения момента в условии равновесия системы:

$$8,79 < M < 11,13 \text{ Нм}. \quad (1.44)$$

В этой задаче имеется односторонняя связь — плоскость опоры цилиндра. Условием реализуемости связи является отсутствие отрыва или неравенство $N > 0$. Решение (1.42) имеет вид $N = 35/(2\delta + 3) = 7M/(6(2\delta + 3))$. Так как найденный интервал (1.44) в условии равновесия системы соответствует $M > 0$, то при этих значениях момента реакция также положительна $N > 0$, следовательно, полученное решение удовлетворяет условию односторонней связи. Вне интервала (1.44) система движется. Очевидно, при $M \rightarrow \infty$ стержень OA будет поворачиваться против часовой стрелки, а цилиндр покатится налево (сила F не удержит это движение). При неограниченном уменьшении момента (со сменой его знака и изменением направления), $M \rightarrow -\infty$, цилиндр будет катиться направо.

Задача 10. Механическая система состоит из невесомого уголка, невесомого стержня и цилиндра весом $P = 20$ Н, радиусом $R = = 1$ м (рис. 42). Стержень, ось цилиндра и уголок соединены шарнирно.

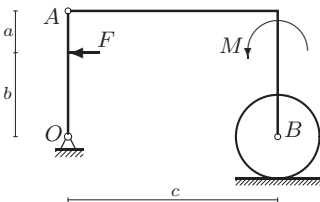


Рис. 42

Цилиндр может катиться без проскальзывания с коэффициентом трения качения $\delta = 2$ см. К стержню приложена горизонтальная сила F , к уголку — момент $M = 3$ Нм. Дано: $a = 1$ м, $b = 2$ м, $c = 5$ м. В каких пределах меняется сила F при условии, что система находится в равновесии?

Решение

Исследуемая система состоит из трех тел. Разобьем систему на части по шарнирам. Действие шарниров заменим их реакциями, не

забывая о противоположном направлении реакций контактирующих тел (рис. 43, 44, 45).

Зададим направление предполагаемого движения. Найдем предельное значение силы F , при котором цилиндр будет катиться налево. Момент трения качения, удерживающий цилиндр от поворота, будет направлен при этом по часовой стрелке.

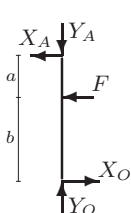


Рис. 43

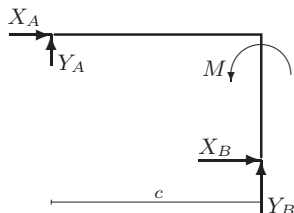


Рис. 44

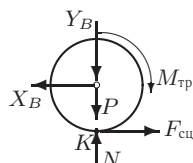


Рис. 45

Для предельного случая (мгновение перехода от покоя к движению) момент трения качения будет

$$M_{\text{тр}} = N\delta. \quad (1.45)$$

Уравнения равновесия стержня AO имеют вид (рис. 43)

$$\begin{aligned} \sum X_i &= X_O - X_A - F = 0, \\ \sum Y_i &= Y_O - Y_A = 0, \\ \sum M_O &= (a + b)X_A + bF = 0. \end{aligned} \quad (1.46)$$

Для уголка AB (рис. 44) записываем уравнения проекций и сумму моментов относительно шарнира A :

$$\begin{aligned} \sum X_i &= X_A + X_B = 0, \\ \sum Y_i &= Y_A + Y_B = 0, \\ \sum M_A &= (a + b)X_B + cY_B + M = 0. \end{aligned} \quad (1.47)$$

В общем случае для цилиндра необходимо составить три уравнения равновесия, однако в данном случае реакцию $F_{\text{сц}}$ можно исключить из системы, выбрав в качестве моментной точки точку касания K (рис. 45). Второе уравнение — сумма проекций на нормаль к плоскости, также не включает в себя эту силу:

$$\begin{aligned} \sum M_K &= R \cdot X_B - M_{\text{тр}} = 0, \\ \sum Y_i &= N - P - Y_B = 0. \end{aligned} \quad (1.48)$$

Решаем систему девяти уравнений (1.45)–(1.48) с девятью неизвестными X_O , Y_O , X_A , Y_A , X_B , Y_B , N , $M_{\text{тр}}$, F и получаем выражение

искомой силы и реакцию опоры через коэффициент трения качения:

$$F = \frac{\delta (a + b) (cP - M)}{b(Rc + \delta (a + b))}, \quad N = \frac{cP - M}{Rc + \delta (a + b)}. \quad (1.49)$$

Подставляем заданные значения: $P = 20$ Н, $M = 3$ Нм, $a = 1$ м, $b = 2$ м, $c = 5$ м, $R = 1$ м, $\delta = 0,02$ м, и вычисляем $F = 0,575$ Н, $N = 19,169$ Н.

Для того чтобы проверить возможность качения цилиндра в противоположную сторону, т. е. направо, необходимо направить $M_{\text{тр}}$ против часовой стрелки, составить систему уравнений равновесия и решить ее относительно M . Рассуждая как в предыдущей задаче на с. 28, заметим, что эта система будет отличаться от уже решенной только знаком перед коэффициентом трения качения δ , следовательно, ее решение получается из предыдущего формальной заменой знака δ . Подставляем в (1.49) $\delta = -0,02$ и вычисляем $F = -0,589$ Н, $N = 19,636$ Н. В итоге получаем область изменения силы F в условии равновесия системы: $-0,589 < F < 0,575$ Н. В этой задаче имеется односторонняя связь — опирание цилиндра на плоскость. Если $N > 0$ связь реализуется, при $N < 0$ происходит отрыв цилиндра и нарушение равновесия. Очевидно, здесь $N > 0$ и отрыва не происходит. Однако помимо возможности качения и отрыва цилиндра от плоскости есть еще одно состояние предельного равновесия, которое здесь не изучается. Это возможность проскальзывания цилиндра. Проскальзывание будет происходить при достижении силой сцепления $F_{\text{сц}}$ своего предельного значения, определяемого формулой Кулона¹: $F_{\text{сц}} = F_{\text{тр}} = Nf$, где f — коэффициент трения, зависящий от свойств контактирующих материалов, N — реакция опоры. По условию задачи полагается, что коэффициент трения достаточно большой.

1.3. Ферма

Задача 11. Плоская статически определимая ферма опирается на подвижный шарнир A и неподвижный B (рис. 46). В узле C ферма нагружена горизонтальной силой $F = 10$ кН. Размеры даны в метрах. Найти усилия в стержнях фермы.

Решение

Определяем реакции опор фермы. Отбрасываем связи (опорные шарниры) и заменяем их действие реакциями X_B , Y_A , Y_B (рис. 47). Систему координат выбираем с началом в точке A .

¹Charles-Augustin de Coulomb (1736–1806) — французский физик, открыл закон сухого трения, один из основателей электростатики.



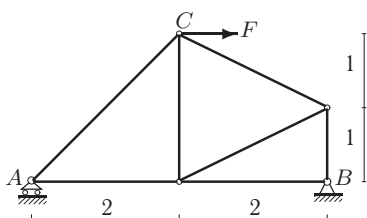


Рис. 46

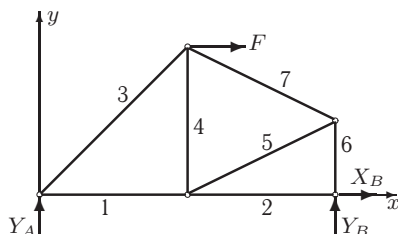


Рис. 47

Составляем три уравнения равновесия:

$$\begin{aligned}\sum X_i &= X_B + F = 0, \\ \sum M_A &= Y_B \cdot 4 - F \cdot 2 = 0, \\ \sum M_B &= -Y_A \cdot 4 - F \cdot 2 = 0.\end{aligned}$$

Решаем систему уравнений. Находим реакции опор: $X_B = -F = -10$ кН, $Y_A = -F/2 = -5$ кН, $Y_B = F/2 = 5$ кН. Для проверки вертикальных реакций составляем сумму проекций на ось y всех сил, действующих на ферму: $\sum Y_i = Y_A + Y_B = 5 - 5 = 0$. Уравнение удовлетворяется тождественно. Реакции Y_A и Y_B найдены верно¹.

Определяем усилия S_i в стержнях фермы. Нумеруем стержни фермы (рис. 47). Усилия S_1, S_3 найдем из условия равновесия узла A (рис. 48), а усилия S_2, S_6 — из условия равновесия узла B (рис. 49).

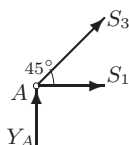


Рис. 48

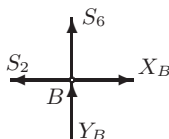


Рис. 49

Реакции рассеченных стержней направляем из узла. Это правило соответствует принятому соглашению, согласно которому в растянутых

¹Более надежная проверка, контролирующая также и реакцию X_B , состоит в проверке выполнения уравнения моментов относительно какой-либо точки, не обязательно совпадающей с узлом и не лежащей на линиях действия проверяемых реакций. В данной задаче это может быть центр средней стойки фермы.

стержнях положительные усилия, а отрицательные — в сжатых. Записываем уравнения равновесия узла A

$$\sum X_i = S_1 + S_3 \cos 45^\circ = 0,$$

$$\sum Y_i = S_3 \sin 45^\circ + Y_A = 0.$$

Решаем систему уравнений и находим $S_1 = -5$ кН, $S_3 = 7,07$ кН. Уравнения равновесия узла B :

$$\sum X_i = -S_2 + X_B = 0,$$

$$\sum Y_i = S_3 + Y_B = 0.$$

Отсюда получаем усилия $S_2 = X_B = -10$ кН, $S_6 = -Y_B = -5$ кН.

Для определения усилий S_5 , S_7 используем метод Риттера [27]. Сечение Риттера¹ должно пересекать три (не больше и не меньше) стержня фермы и делить ее на две части. Отсекаемая часть должна содержать по крайней мере один стержень. Произведем сечение I-I (рис. 50). Отбрасываем левую часть фермы (рис. 51). Рассматриваем

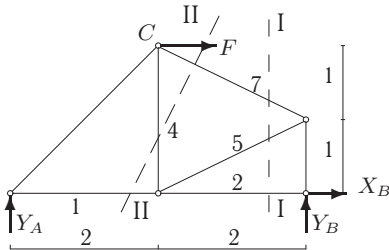


Рис. 50

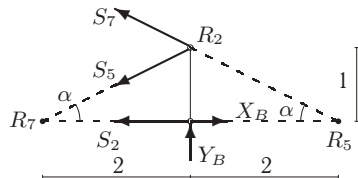


Рис. 51

равновесие оставшейся правой части, состоящей из одного вертикального стержня. Усилия в рассеченных стержнях направляем вдоль стержней в сторону сечения. Находим моментные точки R_5 , R_7 на пересечениях линий действия усилий в сечении. Составляем уравнения моментов относительно этих точек:

$$\sum M_{R_7} = S_7 \cdot 2 \sin \alpha + S_7 \cdot 1 \cos \alpha + Y_B \cdot 2 = 0,$$

$$\sum M_{R_5} = S_5 \cdot 2 \sin \alpha + S_5 \cdot 1 \cos \alpha - Y_B \cdot 2 = 0.$$

Здесь $\sin \alpha = 1/\sqrt{5}$, $\cos \alpha = 2/\sqrt{5}$. Решая уравнения, получаем $S_7 = -5,59$ кН, $S_5 = 5,59$ кН.

¹August Ritter (1826–1906) — немецкий механик.

Для определения усилия S_4 произведем сечение II-II по стержням 1, 4, 7 (рис. 52). Рассматриваем равновесие правой части фермы (стержневой треугольник). Разрезанные стержни заменяем усилиями, направленными по стержням в сторону сечения. То, что векторы с одним и тем же усилием для различных схем оказываются направленными в разные стороны, не является ошибкой.

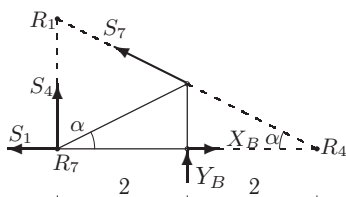


Рис. 52

Напротив, это соответствует аксиоме статики о действии и противодействии. Так, вектор с усилием S_1 на рис. 48 направлен направо, а на рис. 52 — налево. Точка Риттера R_4 для определения усилия S_4 находится на пересечении линий действия усилий S_9 и S_1 и совпадает с точкой R_5 (рис. 51).

Составляем уравнение моментов

$$\sum M_{R_4} = -S_4 \cdot 4 - Y_B \cdot 2 = 0.$$

Получаем $S_4 = -2,5$ кН. Для проверки решения можно найти усилие S_1 , составив уравнение моментов относительно точки R_1 . Ранее это усилие было найдено из уравнения проекции при рассмотрении равновесия узла A . Таким образом, усилия во всех стержнях фермы найдены. Стержни с положительными усилиями (3,5) растянуты, с отрицательными (1, 2, 4, 6, 7) — сжаты.

Маплет для расчета фермы дан на с. 342.

Задача 12 (диаграмма Максвелла–Кремоны¹). Дана ферма (рис. 53), на которую действует вертикальная сила $P = 10$ кН и горизонтальная $Q = 3$ кН. Размеры даны в метрах. Найти усилия в стержнях.

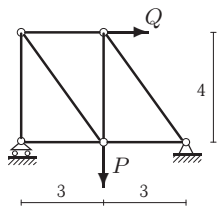


Рис. 53

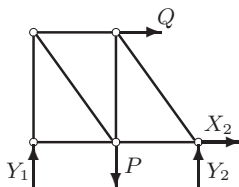


Рис. 54

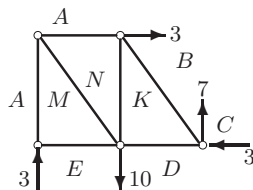


Рис. 55

¹James Clerk Maxwell (1831–1879) — шотландский физик, математик, астроном. Antonio Cremona (1830–1903) — итальянский математик.

Решение

Определяем реакции опор фермы (рис. 54). Составляем три уравнения равновесия:

$$\begin{aligned}\sum X_k &= X_2 + Q = 0, \\ \sum Y_k &= Y_1 + Y_2 - P = 0, \\ \sum M_1 &= Y_2 \cdot 6 - Q \cdot 4 - P \cdot 3 = 0.\end{aligned}$$

Решаем полученную систему уравнений. Находим значения реакций $X_2 = -3$ кН, $Y_1 = 3$ кН, $Y_2 = 7$ кН.

Изображаем все силы, действующие на ферму (включая найденные аналитически реакции опор), в виде векторов *вне* фермы. Реакция X_2 отрицательная, меняем ее направление на противоположное. Для графического способа требуются только реальные направления реакций (рис. 55).

Обозначаем буквами *внешние поля* — области чертежа, разделенные силами и стержнями фермы: A, B, C, D и E (рис. 55). Внешним нагрузкам и реакциям опор даем новые имена — по соседним с силой полям. Выбираем направление обхода фермы — по часовой стрелке. Имеем: $Q = AB$, $Y_1 = EA$, $Y_2 = BC$, $X_2 = CD$, $P = DE$.

Построение диаграммы Максвелла–Кремоны начинаем с многоугольника внешних сил. Начинаем с произвольной силы, например $Q = AB$. Откладывая ее в масштабе и соблюдая направление, отметим на диаграмме начальную и конечную точку буквами, соответствующими ее новому обозначению по направлению обхода (рис. 56). Следующую по выбранному направлению обхода силу $Y_2 = BC$ пристраиваем к первой, затем изображаем силы CD, DE . Замкнутость многоугольника внешних сил и реакций опор силой $Y_1 = EA$. Замкнутость многоугольника является проверкой правильности нахождения реакций опор.

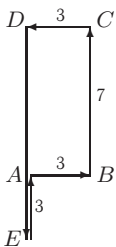


Рис. 56

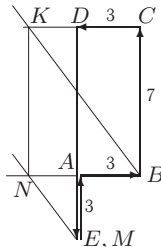


Рис. 57

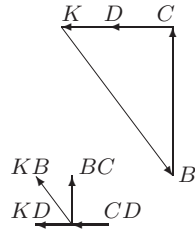


Рис. 58

Обозначаем буквами или цифрами *внутренние поля* — области, ограниченные стержнями фермы: M, N и K .

Усилиям в стержнях даем новые имена — по соседним со стержнем полям. В ферме семь стержней: AM, AN, ME, KB, MN, NK и KD .

Строим точки внутренних полей на диаграмме. Точку, соответствующую внутреннему полю, можно найти, если у этого поля построены точки двух соседних с ней полей. Таким образом, начинать графический расчет можно с поля, у которого имеется два соседних с ним внешних поля, уже отмеченных на диаграмме. Искомая точка лежит на пересечении прямых, параллельных стержням, имена которых состоят из имени искомой точки и точек найденных внешних полей. Стержень KB наклонный. Тангенс угла наклона равен $4/3$. Стержень KD горизонтальный. Проводим соответствующие прямые через точки B и D , получаем на их пересечении искомую точку K (рис. 57). Аналогично находим точку N и M . Из построения получаем, что точка M совпадает с точкой E . Модули усилий в стержнях равны длинам соответствующих отрезков на диаграмме. Следовательно, усилие в стержне ME нижнего пояса фермы равно нулю. Измеряя на чертеже или проводя несложные геометрические расчеты, получаем следующие величины: $|AM| = 3$ кН, $|AN| = 2,25$ кН, $|KB| = 8,75$ кН, $|MN| = 3,75$ кН, $|NK| = 7$ кН, $|KD| = 2,25$ кН.

Определяем знаки усилий. Рассматриваем шарнир фермы, к которому подходит какая-либо внешняя нагрузка или стержень с усилием известного знака, например правый неподвижный опорный шарнир (рис. 58). Равновесие шарнира изображено на диаграмме замкнутым силовым многоугольником с заданным направлением обхода. Сопоставляя направление усилия на диаграмме и его направление в вырезанном узле, определяем знак усилия. Если направление вектора на многоугольнике совпадает с направлением вектора, приложенного к узлу, то усилие больше нуля. В противном случае — усилие меньше нуля, т. е. стержень сжат. В случае узла 2 направление обхода многоугольника \vec{c} против часовой стрелки задают сразу два вектора — \vec{BC} (вверх) и \vec{CD} (налево). Остальные векторы в многоугольнике выстраиваются по этому направлению. В результате получаем усилия со знаком: $AM = -3$ кН, $AN = -2,25$ кН, $KB = -8,75$ кН, $MN = 3,75$ кН, $NK = 7$ кН, $KD = 2,25$ кН.

Задача 13 (метод замены стержней¹). Дана ферма, состоящая из девяти стержней (рис. 59), на которую действует нагрузка P . Считать, что тонкие стержни фермы не пересекаются, располагаясь в разных плоскостях. Найти усилие в стержне BC .

Решение

Обозначим усилие в стержне BC через X . Метод вырезания узлов здесь непосредственно результат не дает. В ферме нет ни одного узла, к которому подходят два стержня, поэтому пошаговый метод расчета

¹Метод Геннеберга. *Ernst Lebrecht Henneberg* (1850–1933) — немецкий математик и механик.

фермы переходом от одного узла к другому не годится в этом случае. Если не рассматривать прямой метод составления уравнений равновесия для всех узлов (в данном случае 12 уравнений для 9 стержней

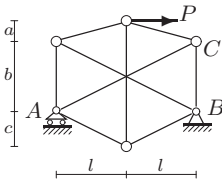


Рис. 59

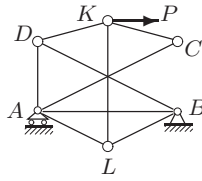


Рис. 60

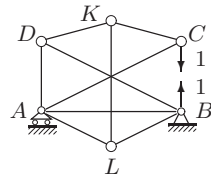


Рис. 61

и трех реакций опор) с последующим их решением и нахождением всех 12 неизвестных, то остается метод сечений¹. Однако и он здесь не подойдет. Сечения Риттера, пересекающего три стержня, включая стержень BC с искомым усилием, здесь нет.

Рассмотрим метод замены стержней (метод Геннеберга). Этот метод сходен по идее с методом сил для расчета статически неопределимых систем (с. 168). Состоит он в следующем.

Стержень BC заменяем стержнем AB ², т. е. разъединяем шарниры B и C и соединяем стержнем шарниры B и A . В новой ферме (рис. 60) определяем усилие во введенном стержне AB . Это легко сделать методом вырезания узлов. Вырезая последовательно узлы C , K , L и B , находим усилие $S_{AB}^{(P)}$. Верхний индекс P указывает происхождение этого усилия. Затем определяем усилие в AB только от действия усилия в отброшенном стержне BC . Само усилие неизвестно (оно как раз и разыскивается). Прикладываем вместо него единичные безразмерные силы (рис. 61). Как и ранее, последовательно вырезая узлы C , K , L и B , находим усилие $S_{AB}^{(1)}$. Если от действия единичных сил усилие в стержне AB равно $S_{AB}^{(1)}$, то от действительного усилия X в стержне BC получаем соответственно $X S_{AB}^{(1)}$. Пользуясь линейностью задачи статики, складываем результаты воздействия нагрузки P и усилия в стержне BC , получаем $S_{AB}^{(P)} + X S_{AB}^{(1)} = 0$, так как в действительности стержня AB нет или, что то же, усилие в нем равно нулю. Решая полученное уравнение относительно X , находим искомое усилие. Часто такой метод в затруднительных ситуациях, подобных описанной, дает более быстрое (и точное) решение, чем непосредственное составление системы уравнений равновесия для всех узлов.

¹Заметим интересную особенность конструкции. При $a = c$ определитель системы уравнений метода вырезания узлов обращается в ноль.

²Убирая из фермы один стержень и заменяя его другим, надо следить за кинематической неизменяемостью фермы. Подробнее об этом см. [49].

Задача 14 (метод замены стержней). Дана ферма из пяти стержней с тремя неподвижными опорами и одной подвижной (рис. 62), на которую действует горизонтальная сила P . Найти усилия в стержнях.

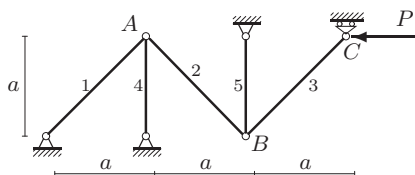


Рис. 62

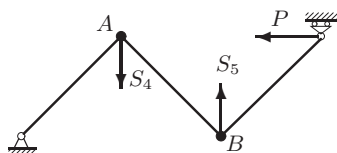


Рис. 63

Решение

Рассмотрим метод замены стержней (метод Геннеберга) для этой задачи¹. Заменяем сразу два стержня 4 и 5 неизвестными силами, а вместо шарниров A и B поставим в этих точках жесткие соединения. Убрав две связи, мы добавили также две. Ферма превратилась в одно твердое тело — изогнутый стержень, слева опертый на неподвижный шарнир и справа — на подвижный, под действием силы P и двух неизвестных сил S_4 и S_5 (рис. 63). Эти силы находим из условия равенства нулю моментов в дополнительных связях. Моменты в точках A и B , появившиеся в результате наложения жестких соединений на шарниры, являются внутренними усилиями в конструкции. Для их определения выполняем сечения в точках A и B и рассматриваем равновесие одной из образовавшихся частей.

Для удобства вычислений воспользуемся свойством линейности задачи и вычислим моменты в точках A и B отдельно для каждой нагрузки. Если на конструкцию действует сила P , то в правой опоре возникает вертикальная реакция $P/3$, значение которой находим из уравнения моментов относительно левой опоры. В левой опоре горизонтальная реакция равна P , вертикальная — $P/3$ (рис. 64). Рассматриваем всякий раз равновесие *правой* части. Внутренние моменты в дополнительных поставленных связях в точках A и B будут равны $M_A(P) = -2Pa/3$, $M_B(P) = Pa - Pa/3 = 2Pa/3$.

Для того чтобы найти моменты в точках A и B от действия только вертикальной силы S_4 , определяем реакции опор ($2S_4/3$ в левой опоре и $S_4/3$ — в правой), а затем вычисляем $M_A(S_4) = 2aS_4/3$, $M_B(S_4) = aS_4/3$ (рис. 65).

¹Эту задачу можно решить также методом вырезания узлов. Система уравнений будет содержать 12 неизвестных, в число которых входят 5 усилий в стержнях и 7 реакции опор.

Аналогично, рассматривая равновесие рамы под действием только S_5 , находим момент $M_A(S_5) = -aS_5 - (4a/3)S_5 = -5aS_5/3$, $M_B(S_5) = -2aS_5/3$ (рис. 66).

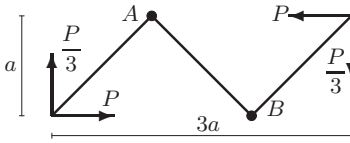


Рис. 64

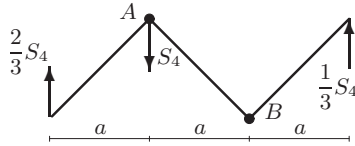


Рис. 65

В сумме моменты в указанных точках от действия всех трех нагрузок одновременно должны быть равны нулю. Получаем систему уравнений:

$$M_A = M_A(P) + M_A(S_4) + M_A(S_5) = -\frac{2Pa}{3} + \frac{2aS_4}{3} - \frac{aS_5}{3} = 0,$$

$$M_B = M_B(P) + M_B(S_4) + M_B(S_5) = \frac{2Pa}{3} + \frac{aS_4}{3} - \frac{2aS_5}{3} = 0.$$

Решаем систему уравнений и получаем искомые усилия $S_4 = S_5 = 2P$. Стержни растянуты. Для проверки решения воспользуемся методом вырезания узлов (рис. 67).

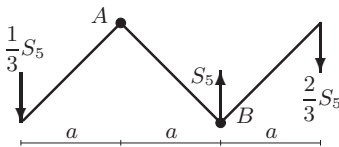


Рис. 66

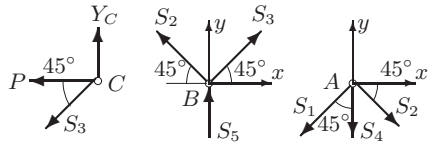


Рис. 67

Из уравнения равновесия узла C в проекции на горизонтальную ось имеем $S_3 \cos 45 + P = 0$, откуда $S_3 = -P\sqrt{2}$. Рассматриваем равновесие узла B . Имеем два уравнения равновесия в проекциях:

$$\sum X_k = S_3 \cos 45 - S_2 \cos 45 = 0,$$

$$\sum Y_k = S_2 \sin 45 + S_3 \sin 45 + S_5 = 0.$$

Из первого уравнения получаем усилие $S_2 = S_3$, из второго — $S_5 = 2P$. Аналогично из условия равновесия узла A получаем уравнения

$$\sum X_k = -S_1 \sin 45 + S_2 \cos 45 = 0,$$

$$\sum Y_k = -S_1 \cos 45 - S_2 \sin 45 - S_4 = 0.$$

В итоге определяем усилия $S_1 = -P\sqrt{2}$, $S_4 = 2P$. Решение этой задачи с помощью маплета дано на с. 348.

Задача 15 (кинематический метод). Дана ферма (рис. 68), на которую действует нагрузка P . Одна опора фермы — неподвижный шарнир, другая — подвижный. Найти усилия в стержнях 2–3 и 2–6.

Решение

Исключаем из фермы стержень, усилие которого разыскивается. Ферма превращается в механизм с одной степенью свободы (рис. 69). Составляем уравнение принципа возможных скоростей $\sum_i \vec{F}_i \cdot \vec{v}_i = 0$ в виде $\sum_i M_{K i_z}(F_i)\omega_{i_z} = 0$, где момент $M_{K i_z}(F_i)$ каждой силы вычисляется относительно мгновенного центра

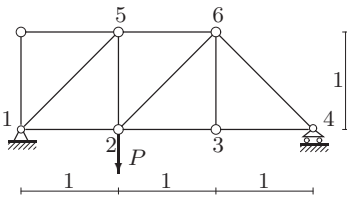


Рис. 68

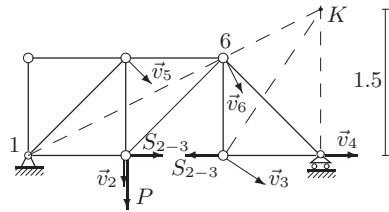


Рис. 69

скоростей (МЦС) тела, к которому эта сила приложена, а ω_{i_z} — это возможная угловая скорость этого тела¹. Индекс z у момента и угловой скорости показывает, что берутся проекции этих величин на ось, перпендикулярную плоскости чертежа. В данной задаче после исключения стержня 2–3 из фермы образуется механизм из двух тел. Левая часть (тело 1) совершает возможное вращение вокруг опоры 1, правая (тело 2) — вокруг K — своего мгновенного центра скоростей (МЦС). Напомним, что МЦС находится на пересечении перпендикуляров к скоростям тела². Для правой части фермы МЦС лежит на пересечении вертикальной прямой, проведенной перпендикулярно возможной скорости \vec{v}_4 подвижного шарнира 4 и прямой, перпендикулярной вектору \vec{v}_6 . Шарнир 6 принадлежит одновременно телу 1 и телу 2. Вектор \vec{v}_6 , как вектор скорости точки, принадлежащей телу 1, совершающему вращательное движение вокруг неподвижного шарнира 1, перпендикулярен радиусу 1–6. Приравняем нулю сумму мощностей

$$-P \cdot 1 \cdot \omega_{1_z} - S_{2-3} \cdot 1.5 \cdot \omega_{2_z} = 0. \quad (1.50)$$

В уравнение вошли только две силы, моменты обеих отрицательные. Момент силы S_{2-3} , приложенной к левой части, относительно опоры

¹Принцип возможных скоростей (или *перемещений*) см. с. 89.

²Задачи на определение скоростей с помощью МЦС см. на с. 56.

1 равен нулю. Пусть задана возможная угловая скорость ω_{1z} тела 1. Выражаем все скорости, входящие в уравнение принципа возможных скоростей через ω_{1z} . Модули угловых скоростей связывает выражение для скорости шарнира, сочленяющего части фермы: $v_6 = L_{16}\omega_1 = L_{6K}\omega_2$, где L_{16} и L_{6K} — расстояния между соответствующими точками. Отсюда получим $\omega_{1z} \cdot 2 = -\omega_{2z} \cdot 1$. Знак минус указывает на противоположные направления угловых скоростей. Подставляем это соотношение в (1.50) и сокращая его на ненулевую заданную возможную угловую скорость ω_{1z} , сразу же получаем $S_{2-3} = P/3$.

Заметим, что это же значение получается методом Риттера из уравнения моментов для левой или правой части вокруг шарнира 6. Однако в этом случае для решения необходимо заранее найти реакции опор. В принципе возможных скоростей, сформулированном для идеальных связей¹, реакции опор в уравнение не входят.

Кинематический метод (принцип возможных скоростей) не всегда дает такое простое решение. В тех случаях, когда после удаления стержня фермы получившиеся части не имеют общего шарнира (в описанном случае это был шарнир 6), кинематика задачи оказывается, как правило, сложной и для решения требуется привлечь аналитические методы, например метод кинематических графов. Рассмотрим случай, когда в подобной задаче можно обойтись графическим методом определения скоростей с помощью МЦС.

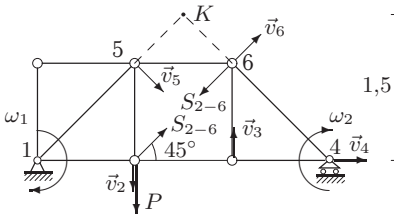


Рис. 70

Находим усилие в стержне 2–6. Удаляем стержень 2–6 из фермы (рис. 70). Левая часть приобретает возможную угловую скорость вращения ω_1 вокруг неподвижного опорного шарнира 1 (например, по часовой стрелке). Движение же правой части сразу не ясно. Очевидно только, что вектор скорости \vec{v}_4 горизонтален, так как шарнир

имеет горизонтальную подвижность. Отметим также, что вектор скорости \vec{v}_2 вертикален и перпендикулярен стержню 2–3, следовательно, из свойства проекций скоростей точек, принадлежащих отрезку, на ось, на которой этот отрезок расположен, вектор скорости \vec{v}_3 также вертикален. Отсюда МЦС тела 2 лежит в точке 4, т.е. $v_4 = 0$. Это позволяет определить направление скорости \vec{v}_6 . Пересечение перпендикуляров к скоростям \vec{v}_5 и \vec{v}_6 дает положение МЦС стержня 5–6. Точка K равноудалена от шарниров 5 и 6, следовательно, $v_6 = v_5$ и $\omega_{1z} = \omega_{2z}$.

¹См. с. 89.

Знаки угловых скоростей совпадают, так как вращение обеих частей происходит в одну сторону, в данном случае — по часовой стрелке. Уравнение принципа возможных скоростей имеет вид

$$S_{2-6} \sin 45^\circ \cdot \omega_{1z} \cdot 1 - P \cdot \omega_{1z} \cdot 1 + S_{2-6} \cdot \omega_{2z} \cdot \sqrt{2} = 0.$$

Отсюда, сокращая на $\omega_{1z} \neq 0$, получаем $S_{2-6} = \sqrt{2}P/3$.

Это же значение получается, если решить задачу методом Риттера. Рассечем ферму по стержням 2-3, 2-6, 5-6 и рассмотрим равновесие левой части (рис. 71). Составляя уравнение проекций на вертикаль

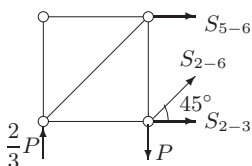


Рис. 71

$$S_{2-6} \sin 45^\circ + 2P/3 - P = 0,$$

получаем тот же результат. Конечно, предварительно из уравнения равновесия фермы в целом (сумма моментов относительно правой опоры) следует

найти реакцию левой опоры $2P/3$. Это отличает статические методы от кинематического, где реакции опор находить не надо.

Задача 16. Плоская ферма состоит из 12 стержней одинаковой длины и трех распорок длиной a (рис. 72). На ферму действуют три

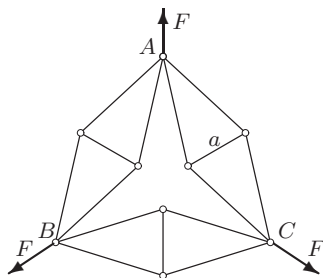


Рис. 72

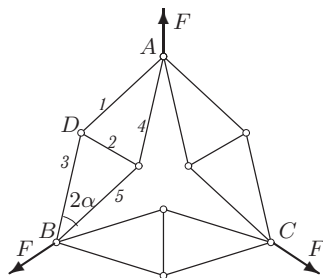


Рис. 73

силы F , линии действия которых направлены под одинаковым углом друг к другу и проходят через геометрический центр фермы, $AB = BC = AC = b$. Найти усилия в распорках.

Решение

Фактически ферма состоит из трех одинаковых шарнирно соединенных ферм. Из симметрии задачи следует, что усилия во всех распорках равны. Пронумеруем стержни одной из ферм (рис. 73). Распорка имеет номер 2.

В этой ферме не надо искать реакции опор — силы, действующие на ферму, уравновешены и демонстрируют теорему о трех силах. Для

определения усилий в стержнях фермы метод Риттера применить не удастся, так как нет сечения Риттера, пересекающего три стержня и разделяющего ферму на две части. Нельзя использовать и геометрический метод Максвелла–Кремоны, где должен быть хотя бы один узел фермы, к которому подходят только два стержня с неизвестными усилиями. По этой же причине нельзя найти усилие в нужном стержне последовательным вырезанием узлов. Можно, конечно, воспользоваться методом вырезания узлов в общей форме, рассмотрев равновесие всех узлов, но при этом придется решить сложную систему уравнений с большим числом неизвестных.

Используем метод разделения фермы на отдельные блоки. Рассматриваемая ферма содержит три одинаковых блока по пять стержней в каждом. На каждый выделенный блок действуют только две внешние силы (реакции). По аксиоме статики эти силы равны по величине и направлены по одной прямой в разные стороны (рис. 74).

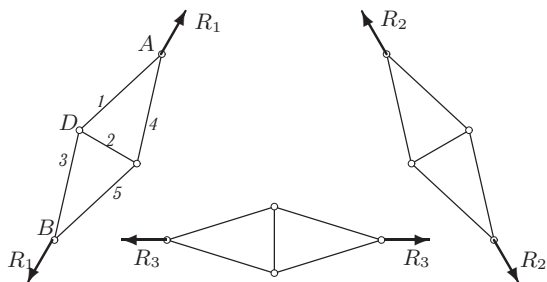


Рис. 74

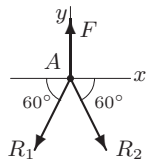


Рис. 75

Из условия равновесия узла A (рис. 75) получаем два уравнения:

$$\begin{aligned}\sum X_k &= -R_1 \cos 60^\circ + R_2 \cos 60^\circ = 0, \\ \sum Y_k &= -R_1 \sin 60^\circ - R_2 \sin 60^\circ + F = 0.\end{aligned}$$

Из первого уравнения находим $R_1 = R_2$, а из второго $R_1 = R_2 = F\sqrt{3}/3$. Аналогично можно получить $R_1 = R_3$. Равенство $R_1 = R_2 = R_3$ следует также просто из симметрии задачи. Теперь, когда известны силы, действующие на отдельный блок, можно найти усилие в нужном стержне. Усилие в стержне 2 можно найти сразу, из одного уравнения, по методу Риттера. Рассекая ферму сечением по стержням 1, 2, 5, рассматриваем равновесие одной из частей и составляем уравнение проекций на ось, перпендикулярную параллельным стержням 1 и 5 (рис. 76). Для определения угла между вектором силы S_2 и осью y опустим из точки D высоту DK треугольника BDE . Углы $\angle KDE$ и $\angle ABE$ равны как углы со взаимно перпендикулярными сторонами: $DE \perp AB$ (диагонали ромба), $DK \perp BE$ (по построению). Отсюда

искомый угол равен α . Очевидно также, что $\operatorname{tg} \alpha = a/b$. В уравнение войдут две силы: $\sum Y_i = -S_2 \cos \alpha - R_1 \sin \alpha = 0$. С учетом найденного значения R_1 получаем $S_2 = -R_1 \operatorname{tg} \alpha = -Fa\sqrt{3}/(3b)$.

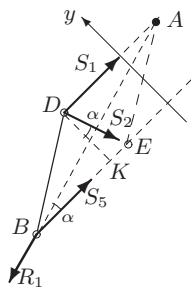


Рис. 76

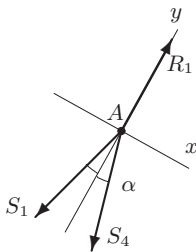


Рис. 77

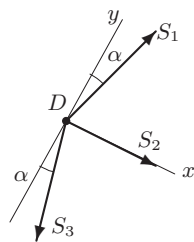


Рис. 78

Другой метод определения усилия S_2 — метод вырезания узлов. Он немного длиннее, но уравнения составляются проще. Опять используем то, что усилие R_1 найдено. Вырезаем узел A (рис. 77). Составляем уравнения равновесия:

$$\begin{aligned}\sum X_k &= -S_1 \sin \alpha + S_4 \sin \alpha = 0, \\ \sum Y_k &= R_1 - S_1 \cos \alpha - S_4 \cos \alpha = 0.\end{aligned}$$

Получаем $S_1 = S_4 = R_1/(2 \cos \alpha) = F\sqrt{3}/(6 \cos \alpha)$. Вырезаем узел D (рис. 78). Составляем уравнения равновесия:

$$\begin{aligned}\sum X_k &= S_2 + S_1 \sin \alpha + S_3 \sin \alpha = 0, \\ \sum Y_k &= S_1 \cos \alpha - S_3 \cos \alpha = 0.\end{aligned}$$

$$S_3 = S_1, S_2 = -2S_1 \sin \alpha = -R_1 \operatorname{tg} \alpha = -Fa\sqrt{3}/(3b).$$

Данное решение существенно упрощается за счет симметрии фермы, однако, используя алгоритм выделения отдельных блоков фермы, можно рассчитать аналогичную ферму с произвольными размерами. Следует помнить, что выделенный блок должен быть фермой, а не механизмом. В статически определимой ферме число стержней s и число узлов n связаны формулой $s = 2n - 3$.

1.4. Пространственная статика

Задача 17. Горизонтальная однородная полка весом $G = 12$ кН имеет в точке A сферическую опору и поддерживается двумя невесомыми, шарнирно закрепленными по концам стержнями (горизонтальным и вертикальным) и подпоркой в точке B (рис. 79). К этой же

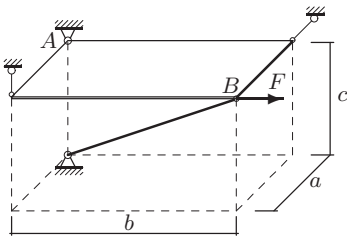


Рис. 79

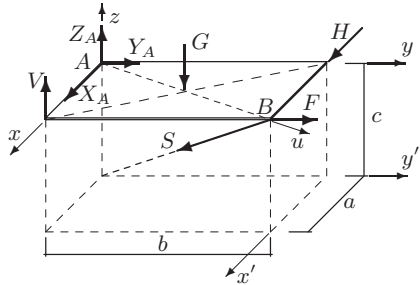


Рис. 80

точке приложена сила $F = 3$ кН, направленная вдоль ребра полки. Даны размеры $a = 2$ м, $b = 6$ м, $c = 3$ м. Определить реакции опор.

Решение

1. Рассматриваем равновесие полки. Начало системы координат xuz помещаем в точку A (рис. 80). Действие на тело опорных стержней заменяем их реакциями. Реакция \vec{V} — вертикальная, \vec{H} — горизонтальная вдоль бокового ребра полки. Усилие \vec{S} в подпорке направлено вдоль стержня. В сферическом шарнире A имеются три составляющие реакции $\vec{X}_A, \vec{Y}_A, \vec{Z}_A$, которые направляем по осям координат. Так как полка однородная, ее центр тяжести совпадает с геометрическим центром. Сюда, на пересечение диагоналей, приложен вес G .

2. Составляем систему уравнений равновесия, состоящую из трех уравнений проекций на оси координат всех сил, действующих на полку, и трех уравнений моментов относительно этих же осей:

$$\begin{aligned} \sum X_k &= X_A + H - S \cos \alpha_x = 0, \\ \sum Y_k &= Y_A - S \cos \alpha_y + F = 0, \\ \sum Z_k &= Z_A + V - S \cos \alpha_z - G = 0, \\ \sum M_{xi} &= -S \cdot b \cos \alpha_z - G \cdot b/2 = 0, \\ \sum M_{yi} &= -V \cdot a + S \cdot a \cos \alpha_z + G \cdot a/2 = 0, \\ \sum M_{zi} &= -H \cdot b + F \cdot a = 0. \end{aligned} \quad (1.51)$$

Здесь $\alpha_x, \alpha_y, \alpha_z$ — углы усилия \vec{S} с осями координат. Вычисляем значения тригонометрических функций: $\cos \alpha_x = a/l = 2/7$, $\cos \alpha_y = b/l = 6/7$, $\cos \alpha_z = c/l = 3/7$, где $l = \sqrt{a^2 + b^2 + c^2} = 7$ м — длина большой диагонали параллелепипеда. Усилие \vec{S} пересекает ось z , поэтому S не входит в последнее уравнение. Из него сразу получаем $H = Fa/b = 1$ кН.

Аналогично из уравнения моментов относительно оси x , находим $S = -G/(2 \cos \alpha_z) = -12 \cdot 7/6 = -14$ кН. Усилие меньше нуля —

стержень сжат. Решая систему (1.51), получаем и остальные реакции: $V = 0$, $X_A = -5$ кН, $Y_A = -15$ кН, $Z_A = 6$ кН, $H = 1$ кН.

3. Выполняем проверку решения, подставляя найденные значения в уравнение моментов относительно дополнительных осей x' и y' , проведенных параллельно соответствующим осям исходной системы координат:

$$\begin{aligned}\sum M_{x'i} &= -Y_A \cdot c - Z_A \cdot b - V \cdot b + S \cdot c \cdot \cos \alpha_y + G \cdot b/2 - F \cdot c = \\ &= 15 \cdot 3 - 6 \cdot 6 - 14 \cdot 3 \cdot (6/7) + 12 \cdot 3 - 3 \cdot 3 = 0, \\ \sum M_{y'i} &= X_A \cdot c - V \cdot a + G \cdot a/2 + H \cdot c = -5 \cdot 3 + 12 \cdot 1 + 1 \cdot 3 = 0.\end{aligned}$$

Суммы моментов равны нулю, проверка выполнена.

Замечание. Из решения системы (1.51) получается $V = 0$. В этом можно убедиться сразу из уравнения моментов относительно дополнительной оси u , лежащей на диагонали полки AB (рис. 80). Действительно, все векторы, кроме \vec{V} , пересекают эту ось, и их моменты равны нулю. Уравнение принимает простой вид $\sum M_u = V \cdot h = 0$, где h — некоторое плечо реакции \vec{V} относительно диагональной оси u . Не вычисляя $h \neq 0$, получаем $V = 0$.

Maple–программа расчета реакций опор полки дана на с. 350.

Задача 18. К точкам $A_1(0, 3, 0)$, $A_2(1, 3, 0)$ и $A_3(0, 0, 4)$ приложены, соответственно, силы $\vec{F}_1(0, 0, 1)$, $\vec{F}_2(0, -3, 0)$, $\vec{F}_3(0, 5, 0)$ и $\vec{F}_4(2, 0, 0)$ (рис. 81). Проекции сил даны в ньютонах, координаты точек — в метрах. Найдите статические инварианты системы сил.

Решение

Находим главный вектор системы — векторный инвариант:

$$\begin{aligned}R_x &= \sum_{k=1}^4 F_{kx} = 2 \text{ Н}, \\ R_y &= \sum_{k=1}^4 F_{ky} = 5 - 3 = 2 \text{ Н}, \\ R_z &= \sum_{k=1}^4 F_{kz} = 1 \text{ Н}.\end{aligned}$$

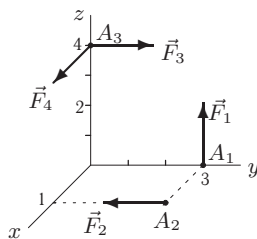


Рис. 81

Момент силы \vec{F} в точке $A(x, y, z)$ относительно начала координат вычисляется по формуле (1.1), с. 9. Проекции вектора момента на оси (или моменты силы относительно осей) имеют вид

$$M_x = yF_z - zF_y, \quad M_y = zF_x - xF_z, \quad M_z = xF_y - yF_x. \quad (1.52)$$

Находим проекции главного момента $\vec{M}_0 = \sum \vec{M}_0(\vec{F}_k)$ системы сил как суммы моментов сил \vec{F}_1 , \vec{F}_2 и \vec{F}_3 относительно осей:

$$M_x = \sum_{k=1}^4 M_{kx} = F_1 \cdot 3 - F_3 \cdot 4 = 1 \cdot 3 - 5 \cdot 4 = -17 \text{ Нм},$$

$$M_y = \sum_{k=1}^4 M_{ky} = F_4 \cdot 4 = 8 \text{ Нм},$$

$$M_z = \sum_{k=1}^4 M_{kz} = -F_2 \cdot 1 = -3 \text{ Нм}.$$

Находим скалярный инвариант системы сил:

$$I = \vec{M}_0 \cdot \vec{R} = M_x R_x + M_y R_y + M_z R_z = -21 \text{ Н}^2\text{м}.$$

Замечание. При вычислении моментов относительно осей не обязательно пользоваться аналитическими формулами (1.52). Воспользуемся определением момента силы относительно оси на с. 10, рис. 3. Найдем момент силы \vec{F}_1 относительно оси x . Из рисунка ясно, что вектор \vec{F}_1 стремится повернуть тело, к которому она приложена, вокруг оси x *против часовой* стрелки с плечом 3 м, следовательно, момент равен $3 \cdot 1 = 3$ Нм. Сила \vec{F}_3 стремится повернуть тело, к которому она приложена, вокруг оси x *по часовой* стрелке с плечом 4, и ее момент равен $-5 \cdot 4 = -20$ Нм. В сумме получаем $M_x = -17$ Нм. Относительно оси y момент имеет только сила \vec{F}_4 с плечом 4, силы \vec{F}_2 и \vec{F}_3 параллельны оси, а сила \vec{F}_1 пересекает ось¹. Вращательной способности у сил \vec{F}_1 , \vec{F}_2 , \vec{F}_3 нет, в результате $M_y = 2 \cdot 4 = 8$ Нм. Сила \vec{F}_2 стремится повернуть вокруг оси z *против часовой* стрелки с плечом 1 м, если смотреть с конца оси z , т. е. сверху. Вычисляем момент этой силы: $-3 \cdot 1 = -3$ Нм.

Проверка. Значение инварианта системы сил не зависит от выбора точки приведения. В данном решении точка приведения находилась в начале координат. Расположим для проверки точку приведения в другом месте, например на плоскости xy в точке $K(1, -1, 0)$ (рис. 82). Вычислим главный момент:

$$M_{x'} = \sum_{k=1}^4 M_{kx'} = F_1 \cdot 4 - F_3 \cdot 4 = -16 \text{ Нм},$$

$$M_{y'} = \sum_{k=1}^4 M_{ky'} = F_1 \cdot 1 + F_4 \cdot 4 = 9 \text{ Нм},$$

$$M_{z'} = \sum_{k=1}^4 M_{kz'} = -F_3 \cdot 1 - F_4 \cdot 1 = -7 \text{ Нм}.$$

¹ В общем, если вектор силы и ось лежат в одной плоскости, то момент силы относительно этой оси равен нулю.

Скалярный инвариант системы сил имеет вид $I = \vec{M}_K \cdot \vec{R} = M_{x'}R_x + M_{y'}R_y + M_{z'}R_z = -32 + 18 - 7 = -21 \text{ Н}\cdot\text{м}$. Проверка

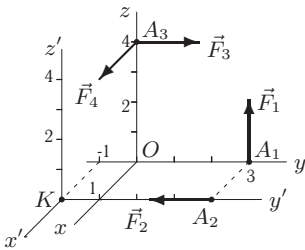


Рис. 82

выполнена. Значение инварианта получилось тем же. Интересно сравнить модули главных моментов относительно точек O и K . Имеем $M_O = \sqrt{17^2 + 8^2 + 3^2} = 19,026 \text{ Нм}$, $M_K = \sqrt{16^2 + 9^2 + 7^2} = 19,646 \text{ Нм}$. Модуль момента увеличился. Главный момент относительно различных центров приведения меняет и величину и направление, а скалярный инвариант I и главный вектор

$R = \sqrt{R_x^2 + R_y^2 + R_z^2} = \sqrt{2^2 + 2^2 + 1^2} = 3 \text{ Н}$ остаются постоянными.

Минимальное значение модуля главного момента вычисляется по формуле $M_* = |I|/R = 21/3 = 7 \text{ Нм}$. Точки, относительно которых главный момент минимальный, образуют прямую — центральную винтовую ось, уравнение которой имеет вид

$$\frac{M_{Ox} - yR_z + zR_y}{R_x} = \frac{M_{Oy} - zR_x + xR_z}{R_y} = \frac{M_{Oz} - xR_y + yR_x}{R_z} = p,$$

где $p = M_*/R$ — шаг винта. Индексы в уравнениях образуют круговую перестановку $\vec{x} \rightarrow \vec{y} \rightarrow \vec{z}$. Если систему привести к любой точке на центральной винтовой оси, то главный вектор и главный момент будут лежать на этой оси и образовывать так называемый силовой винт, или динаму.

Замечание. Для того чтобы определить знак момента силы относительно оси, есть простое правило. Начало и конец вектора силы

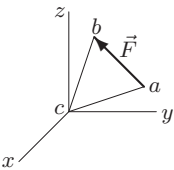


Рис. 83

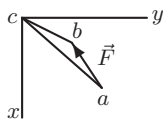


Рис. 84

соединяем с любой точкой оси. Образуется треугольник abc (рис. 83). Смотрим на этот треугольник с конца оси (рис. 84). Если вектор задает обход треугольника против часовой стрелки, то момент положительный, в противном случае — отрицательный.

Если треугольника не видно (вектор силы параллелен оси) или треугольник вырождается в отрезок (сила пересекает ось), то момент силы равен нулю.

Maple-программа для решения этой задачи дана на с. 352.

Глава 2

КИНЕМАТИКА

Кинематика — наука о движении геометрических тел. В ней рассматривается само движение без изучения причин, вызывающих это движение.

Простейшим объектом в кинематике является точка, более сложным — тело. Среди простейших движений тела различают поступательное, вращательное и плоское движение. Во вращательном движении вводятся понятия угла поворота тела $\varphi(t)$, угловой скорости и углового ускорения. Последние две величины векторные, но для вращательного движения их направление всегда постоянно — по оси вращения. Поэтому в решении часто используются скалярные величины $\omega_z(t) = \dot{\varphi}(t)$, $\varepsilon_z(t) = \dot{\omega}_z(t)$, являющиеся проекциями этих векторов на ось вращения z . Точкой будем обозначать производную по времени.

Модуль скорости точки тела при вращательном движении определяется по формуле $v = R\omega$, ускорение — $a = R\sqrt{\omega^4 + \varepsilon^2}$, где R — расстояние от точки до оси вращения.

В плоском движении тела каждая точка тела движется в плоскости, параллельной некоторой фиксированной плоскости. Само тело вовсе не обязательно должно быть плоским. Говорить о скорости тела или его ускорении в общем случае не имеет смысла: тело состоит из множества точек, каждая из которых может иметь свою скорость и ускорение. Исключение составляет поступательное движение тела, при котором равны скорости всех точек и равны их ускорения. Кроме того, в некоторых задачах иногда говорят, например, о скорости катящегося цилиндра или о скорости автомобиля, подразумевая при этом скорость точек центральной оси цилиндра или скорость кузова автомобиля, принимая его за точку.

Угловая скорость и ускорение для плоского движения — векторные величины. Направления этих векторов всегда перпендикулярны плоскости движения. Введем декартову систему координат, в которой плоскость xy совпадает с плоскостью движения. Тогда угловая скорость $\vec{\omega}$ и ускорение $\vec{\varepsilon}$ направлены вдоль оси z . В решении задач удобно использовать скалярные величины — проекции этих векторов на ось z : ω_z и ε_z . Вращение против часовой стрелки соответствует $\omega_z > 0$. Если вращение ускоренное, то проекции ω_z и ε_z имеют один знак. Основные методы вычислений скоростей точек тела при плоском движении см. в разделе 2.2.

2.1. Кинематика точки

Задача 19. Точка движется по закону

$$x = t, \quad y = 2t(t - 1)(t - 2). \quad (2.1)$$

Для момента времени $t = t_1 = 0,5$ найти скорость, ускорение точки и радиус кривизны траектории. Координаты x, y даны в сантиметрах, время — в секундах.

Решение

Определяем траекторию движения точки, исключая t из закона движения (2.1). Параметрическим представлением траектории является сам закон движения (2.1). Координатную форму уравнения движения точки получаем, исключая из закона движения (2.1) время:

$$y = 2x(x^2 - 3x + 2).$$

Траекторией является кубическая парабола (рис. 85). График строим по точкам (отмечены звездочками), через равные промежутки времени 0,1 с.

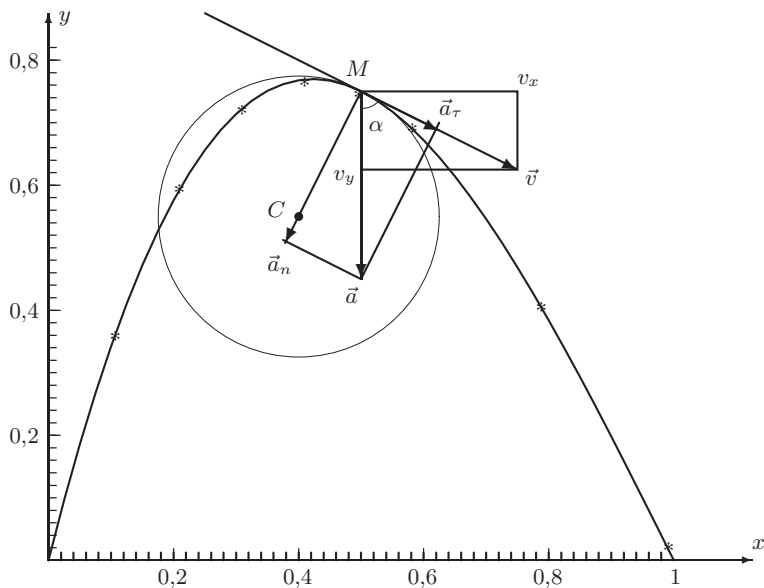


Рис. 85

Дифференцируя (2.1) по времени t , находим проекции скорости точки на оси x, y :

$$\begin{aligned}v_x &= \dot{x} = 1, \\v_y &= \dot{y} = 2(3t^2 - 6t + 2).\end{aligned}\quad (2.2)$$

При $t = 0,5$ имеем следующие численные значения компонентов скорости:

$$\begin{aligned}v_x &= 1 \text{ см/с}, \\v_y &= 2(3/4 - 3 + 2) = -0,5 \text{ см/с}.\end{aligned}$$

Модуль скорости вычисляем по формуле

$$v = \sqrt{v_x^2 + v_y^2} = \sqrt{5}/2 = 1,12 \text{ см/с}.$$

Вектор скорости \vec{v} строим на рисунке в масштабе по известным компонентам v_x и v_y . Если в вычислениях нет ошибок, то вектор скорости будет направлен по касательной к траектории (рис. 85).

Дифференцируя (2.2), находим компоненты вектора ускорения:

$$\begin{aligned}a_x &= \dot{v}_x = \ddot{x} = 0, \\a_y &= \dot{v}_y = \ddot{y} = 2(6t - 6).\end{aligned}$$

При $t = 0,5$ имеем значения

$$\begin{aligned}a_x &= 0, \\a_y &= -6 \text{ см/с}^2.\end{aligned}$$

Определяем модуль ускорения $a = \sqrt{a_x^2 + a_y^2} = 6 \text{ см/с}^2$. Вектор ускорения строим на чертеже в масштабе ускорений (не обязательно совпадающем с масштабом скоростей). Вектор ускорения направлен внутрь вогнутости кривой.

Вычисляем тангенциальное ускорение¹:

$$a_\tau = \frac{v_x a_x + v_y a_y}{v} = \frac{-0,5 \cdot (-6)}{\sqrt{5}/2} = \frac{6\sqrt{5}}{5} = 2,68 \text{ см/с}^2.$$

Заметим, что найденное значение a_τ — это не модуль вектора \vec{a}_τ , а проекция \vec{a} на \vec{v} . Если скорость и касательное ускорение направлены в одну сторону, то $a_\tau > 0$, а если в разные, то $a_\tau < 0$.

¹Наличие тангенциального (касательного) ускорения точки видно уже из рис. 85. После прохождения вершины параболы расстояние между точками увеличивается, хотя интервал времени одинаков, следовательно, скорость со временем увеличивается. Характеристикой такого изменения является величина $a_\tau = \dot{v}_\tau$.

Вычисляем нормальное ускорение:

$$a_n = \sqrt{a^2 - a_\tau^2} = \sqrt{36 - \frac{36}{5}} = \frac{12\sqrt{5}}{5} = 5,36 \text{ см/с}^2.$$

Нормальное ускорение можно вычислить иначе. Так как $a_n = a \sin \alpha$, где α — угол между скоростью, направленной по касательной к траектории, и вектором ускорения, и $|\vec{v} \times \vec{a}| = va \sin \alpha = va_n$, то

$$a_n = \frac{|\vec{v} \times \vec{a}|}{v} = \frac{|v_x a_y - v_y a_x|}{v} = \frac{6}{\sqrt{5}/2} = 5,36 \text{ см/с}^2.$$

Результат получается тот же.

В итоге находим радиус кривизны траектории в указанном положении точки:

$$R = \frac{v^2}{a_n} = \frac{5\sqrt{5}}{48} = 0,233 \text{ см}.$$

Центр кривизны траектории C лежит на нормали к кривой на расстоянии $R = MC = 0,233$ см внутри вогнутости кривой. Окружность радиусом R с центром в этой точке максимально близко совпадает с кривой в малой окрестности от нее.

Maple–программа, рассчитывающая скорость, ускорение и радиус кривизны, и иллюстрирующая движение точки, дана на с. 358.

Задача 20. Задан закон движения точки в полярных координатах:

$$\rho = t \sin(3t), \quad \varphi = t^3. \quad (2.3)$$

Найти скорость и ускорение точки в полярных, декартовых и естественных координатах при $t = 1$ с. Радиус дан в метрах.

Решение

Вычисляем полярные координаты точки в заданный момент времени¹: $\rho = 1 \sin(1) = 0,141$, $\varphi = 1$.

Дифференцируя (2.3) по времени t , находим производные полярного радиуса ρ и полярного угла:

$$\dot{\rho} = \sin(3t) + 3t \cos(3t), \quad \dot{\varphi} = 3t^2. \quad (2.4)$$

При $t = 1$ имеем $\dot{\rho} = -2,829$, $\dot{\varphi} = 3$.

Вычисляем компоненты скорости в полярных координатах:

$$v_\rho = \dot{\rho} = -2,829 \text{ м/с}, \quad v_\varphi = \rho \dot{\varphi} = 0,141 \cdot 3 = 0,423 \text{ м/с}.$$

Вычисляем модуль скорости: $v = \sqrt{v_\rho^2 + v_\varphi^2} = 2,860 \text{ м/с}.$

¹ Аргументы тригонометрических функций измеряются в радианах.

Вычисляем компоненты скорости в декартовых координатах:

$$v_x = v_\rho \cos \varphi - v_\varphi \sin \varphi = -1,885 \text{ м/с},$$

$$v_y = v_\rho \sin \varphi + v_\varphi \cos \varphi = -2,152 \text{ м/с}.$$

Делаем проверку, вычисляя модуль скорости по декартовым компонентам:

$$v = \sqrt{v_x^2 + v_y^2} = \sqrt{1,885^2 + 2,152^2} = 2,860 \text{ м/с}.$$

Дифференцируя (2.4), находим вторые производные полярного радиуса ρ и полярного угла:

$$\ddot{\rho} = d^2 \rho / dt^2 = 6 \cos(3t) - 9t \sin(3t),$$

$$\ddot{\varphi} = d^2 \varphi / dt^2 = 6t.$$

При $t = 1$ получаем $\dot{\rho} = -7,210$, $\dot{\varphi} = 6$.

Вычисляем компоненты ускорения в полярных координатах:

$$a_\rho = \ddot{\rho} - \rho \dot{\varphi}^2 = -8,480 \text{ м/с}^2, \quad a_\varphi = \rho \ddot{\varphi} + 2\dot{\rho}\dot{\varphi} = -16,126 \text{ м/с}^2.$$

Определяем модуль ускорения: $a = \sqrt{a_\rho^2 + a_\varphi^2} = 18,220 \text{ м/с}^2$.

Находим компоненты ускорения в декартовых координатах:

$$a_x = a_\rho \cos \varphi - a_\varphi \sin \varphi = 8,988 \text{ м/с}^2,$$

$$a_y = a_\rho \sin \varphi + a_\varphi \cos \varphi = -15,849 \text{ м/с}^2.$$

Делаем проверку, вычисляя модуль ускорения по декартовым компонентам:

$$a = \sqrt{a_x^2 + a_y^2} = \sqrt{8,988^2 + 15,849^2} = 18,220 \text{ м/с}^2.$$

Находим модуль тангенциального ускорения,

$$|a_\tau| = \frac{|v_x a_x + v_y a_y|}{v} = 6,000 \text{ м/с}^2,$$

и проверим его по формуле

$$|a_\tau| = \frac{|v_\rho a_\rho + v_\varphi a_\varphi|}{v} = \frac{|2,829 \cdot 8,48 - 0,423 \cdot 16,126|}{2,86} = 6,000 \text{ м/с}^2.$$

Вычисляем нормальное ускорение:

$$a_n = \sqrt{a^2 - a_\tau^2} = 17,204 \text{ м/с}^2.$$

Maple–программа, рассчитывающая скорость и ускорение точки в полярной системе координат, дана на с. 359.



2.2. Методы расчета кинематики плоского движения

Скорость точки B тела при плоском движении вычисляют через известную скорость какой-либо точки A того же тела, принимаемой за полюс (рис. 86):

$$\vec{v}_B = \vec{v}_A + \vec{v}_{BA}, \quad \vec{v}_{BA} = \vec{\omega}_1 \times \vec{AB}. \quad (2.5)$$

Ускорения точек тела при плоском движении связаны формулой¹

$$\vec{a}_B = \vec{a}_A + \vec{\varepsilon} \times \vec{AB} + \vec{\omega} \times \vec{v}_{BA}. \quad (2.6)$$

1. Метод графов. Для расчета скоростей точек многосвязного механизма, каждое звено которого совершает плоское движение, формулу (2.5) применяют последовательно для всех точек, переходя от одной точки, принимаемой за полюс, к другой. Схему вычислений в этом случае удобно записывать в виде структурных формул (графов [48]):

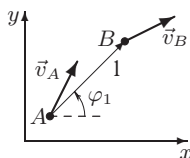


Рис. 86

$$A \xrightarrow[\varphi_1]{1} B, \quad (2.7)$$

где над стрелкой указан номер тела или наименование стержня, которому принадлежат точки, а снизу — угол φ между осью x и вектором \vec{AB} . В проекциях на оси x, y граф (2.7) дает уравнения

$$\begin{aligned} v_{Bx} &= v_{Ax} - AB \omega_{1z} \sin \varphi_1, \\ v_{By} &= v_{Ay} + AB \omega_{1z} \cos \varphi_1, \end{aligned} \quad (2.8)$$

где ω_{1z} — проекция угловой скорости тела 1 на ось z , перпендикулярную плоскости движения. Если вращение происходит против часовой стрелки, то $\omega_{1z} = |\omega_1|$, а если по часовой стрелке, то $\omega_{1z} = -|\omega_1|$.

В качестве вершин графа удобно брать точки механизма с заданными или искомыми скоростями. При этом скорость может быть задана частично, например только по направлению. Если в задаче имеется тело (обычно диск или цилиндр), катящееся без проскальзывания по какой-либо поверхности, то точка касания тела может быть вершиной графа, так как скорость ее равна нулю.

Для многосвязного механизма (рис. 87) из графов вида (2.7) можно образовать цепочку

$$A \xrightarrow[\varphi_1]{1} B \xrightarrow[\varphi_2]{2} C \xrightarrow[\varphi_3]{3} D,$$

¹Формула Ривальса.

особенно удобную для связи скоростей \vec{v}_A и \vec{v}_D , в тех случаях, когда скорости промежуточных точек B и C в задачу не входят.

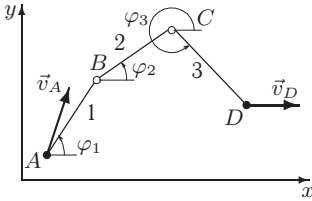


Рис. 87

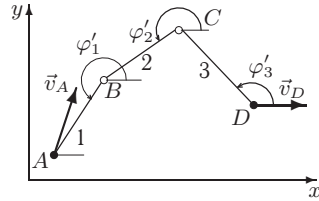


Рис. 88

В проекциях на оси этот граф дает соотношения:

$$\begin{aligned} v_{Dx} &= v_{Ax} - AB \omega_{1z} \sin \varphi_1 - BC \omega_{2z} \sin \varphi_2 - CD \omega_{3z} \sin \varphi_3, \\ v_{Dy} &= v_{Ay} + AB \omega_{1z} \cos \varphi_1 + BC \omega_{2z} \cos \varphi_2 + CD \omega_{3z} \cos \varphi_3. \end{aligned}$$

Многозвенный механизм можно пройти и в обратном направлении (рис. 88). Углы к направлениям стержней будут, как и ранее, отсчитываться от положительного направления оси x в начале стержня:

$$D \xrightarrow{\frac{3}{\varphi_3}} C \xrightarrow{\frac{2}{\varphi_2}} B \xrightarrow{\frac{1}{\varphi_1}} A,$$

где $\varphi'_k = \varphi_k + \pi$. Соотношение между скоростями точек при этом не изменится.

На с. 361 приведена **Maple**-программа для получения уравнений кинематического графа.

2. Метод мгновенных центров скоростей (МЦС). Легко показать, что в каждый момент при плоском движении тела, если его угловая скорость отлична от нуля, существует точка тела, скорость которой равна нулю и эта точка единственная. Точка МЦС в общем случае может и не принадлежать телу и, находясь за его пределами, геометрически быть связанной с ним. Доказано, что МЦС тела лежит на пересечении перпендикуляров, проведенных к скоростям точек, принадлежащих этому телу (рис. 89). У тех звеньев, у которых МЦС не существует (скорости двух точек параллельны и не перпендикулярны отрезку, их соединяющему), угловая скорость равна нулю, а скорости всех точек равны. Если векторы скоростей перпендикулярны отрезку, их соединяющему, то имеют место два частных случая положения МЦС (рис. 90, 91).

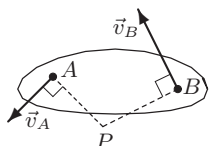


Рис. 89

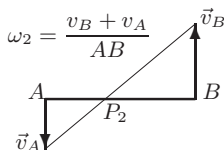


Рис. 90

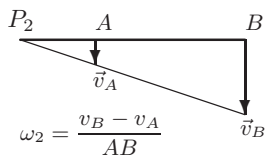


Рис. 91

Если тело (колесо, диск, цилиндр) катится по поверхности без проскальзывания, то МЦС этого тела находится в точке касания.

Рассчитывая скорости точек тела, определяем расстояния от точек до МЦС этого тела. Записываем систему уравнений для скоростей N точек тела i : $v_k = \omega_i R_{ik}$, $k = 1 \dots N$. Здесь ω_i — угловая скорость тела i , R_{ik} — расстояние от МЦС тела i до точки k . Решаем систему, в которой, как правило, одна из скоростей известна, определяем угловую скорость тела, а затем скорости всех его точек. При расчете многозвенного механизма шарнирные точки сочленения принадлежат одновременно двум (и более) звеньям. Это позволяет последовательно рассчитывать скорости точек механизма, передавая информацию о скоростях от одного звена к другому.

Недостатком метода МЦС является то, что здесь определяются только модули скоростей и угловых скоростей, а в некоторых задачах, например при составлении уравнения Лагранжа 2-го рода, с. 91, требуются проекции этих величин, вычисленные с учетом их знака. Кроме того, есть задачи, где метод МЦС не применим (с. 66).

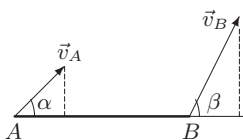


Рис. 92

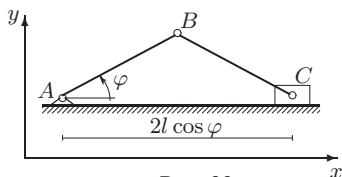


Рис. 93

Для определения величин и направлений скоростей можно пользоваться простым свойством векторов скоростей твердого тела: проекции скоростей неизменяемого отрезка на ось отрезка равны, $v_A \cos \alpha = v_B \cos \beta$ (рис. 92).

Кроме того, вычислять проекции скоростей удобно с помощью координатного метода. Рассмотрим кривошипно-шатунный механизм (рис. 93). Кривошип AB , вращаясь вокруг оси в подшипнике A , посредством шатуна BC сообщает возвратно-поступательное движение ползуну C . Дано: $AB = BC = l$. Силы, действующие на механизм, не указаны, для кинематического анализа они не требуются.

Пусть φ — обобщенная координата. Найдем скорость ползуна. Для этого определим его координату: $x_C = x_A + 2l \cos \varphi$. Дифференцируя это равенство, получаем $v_{Cx} = -2l\dot{\varphi} \sin \varphi$.

Легко проверить, что этот же результат получается методом графов.

Строим граф $A \xrightarrow{l} B \xrightarrow{-l} C$. Записываем оба уравнения графа в проекции на оси x и y . Вычисляем из второго уравнения графа (в проекции на y) скорость $v_{Cy} = v_{Ay} + l\dot{\varphi} \cos \varphi + l\omega_{3z} \cos(-\varphi)$. При $v_{Cy} = v_{Ay} = 0$ получаем отсюда интуитивно ясный результат: $\omega_{3z} = -\dot{\varphi}$. Стержни одинаковой длины вращаются с одинаковой угловой скоростью, но в разные стороны. Подставляем угловую скорость в первое уравнение графа,

$$v_{Cx} = v_{Ax} - l\dot{\varphi} \sin \varphi - l\omega_{3z} \sin(-\varphi),$$

и получаем тот же результат, но более сложным способом.

Заметим, что обычно в задачах зависимость координат от времени получается такой сложной, что продифференцировать ее можно только в системах компьютерной математики, в том числе **Maple** (оператор **diff**). Метод графов более универсальный — он применим для большинства задач.

3. Уравнение трех угловых скоростей. Одним из самых распространенных элементов стержневых механизмов является четырехзвенник, состоящий из трех шарнирно соединенных стержней на двух неподвижных опорах. Четвертым элементом механизма является основание, на котором он закреплен (рис. 94). Механизм приводится в движение вращением одного из звеньев. Найдем связь угловых скоростей звеньев. Составляем кинематический граф

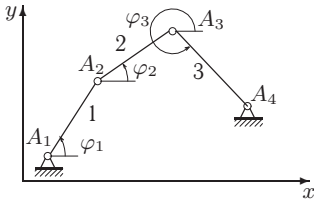


Рис. 94

$$A_1 \xrightarrow[\varphi_1]{1} A_2 \xrightarrow[\varphi_2]{2} A_3 \xrightarrow[\varphi_3]{3} A_4.$$

Записываем уравнения графа:

$$v_{A_4x} = v_{A_1x} - l_1 \omega_{1z} \sin \varphi_1 - l_2 \omega_{2z} \sin \varphi_2 - l_3 \omega_{3z} \sin \varphi_3,$$

$$v_{A_4y} = v_{A_1y} + l_1 \omega_{1z} \cos \varphi_1 + l_2 \omega_{2z} \cos \varphi_2 + l_3 \omega_{3z} \cos \varphi_3.$$

Координатная форма записи этих уравнений дает уравнения трех угловых скоростей

$$\begin{aligned} \sum_{i=1}^3 \omega_{m_{iz}} (x_{n_i} - x_{n_{i+1}}) &= 0, \\ \sum_{i=1}^3 \omega_{m_{iz}} (y_{n_i} - y_{n_{i+1}}) &= 0, \end{aligned} \quad (2.9)$$

где $\omega_{m_{iz}}$ — угловая скорость m_i -го звена, $x_{n_i}, y_{n_i}, x_{n_{i+1}}, y_{n_{i+1}}$ — координаты его концов.

Номера шарниров $n_i, i = 1, \dots, 4$, как и номера звеньев $m_i, i = 1, \dots, 3$, не обязательно должны быть последовательными числами. Если угловая скорость одного из звеньев задана, то угловые скорости двух других легко найти из полученной системы уравнений. В некоторых задачах [27] заданы все три угловые скорости, а определяется конфигурация механизма — положение звеньев, соответствующее этим угловым скоростям. В таких задачах метод МЦС не применим, метод графов в тригонометрической форме неэффективен, а уравнения трех угловых скоростей позволяют просто решить задачу.

Интересен один простой и наглядный частный случай.

Пусть $y_1 = y_4, y_2 = y_3$. Это означает, что четырехзвенник принимает форму трапеции (рис. 95). Из второго уравнения (2.9) имеем теорему трапеции, утверждающую, что *угловые скорости боковых звеньев четырехзвенника, имеющего в данный момент форму трапеции, равны: $\omega_{1z} = \omega_{3z}$.*

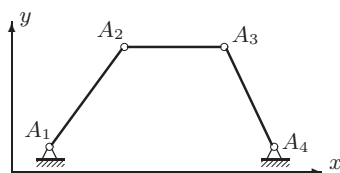


Рис. 95

Уравнения трех угловых ускорений четырехзвенника следуют непосредственно из формулы Ривальса и имеют вид

$$\begin{aligned} \sum_{i=1}^3 \varepsilon_{m_{iz}} (x_{n_i} - x_{n_{i+1}}) - \sum_{i=1}^3 \omega_{m_{iz}}^2 (y_{n_i} - y_{n_{i+1}}) &= 0, \\ \sum_{i=1}^3 \varepsilon_{m_{iz}} (y_{n_i} - y_{n_{i+1}}) + \sum_{i=1}^3 \omega_{m_{iz}}^2 (x_{n_i} - x_{n_{i+1}}) &= 0, \end{aligned} \quad (2.10)$$

где $\varepsilon_{m_{iz}}$ — угловое ускорение m_i -го звена. Если угловые скорости известны, то система уравнений (2.10) позволяет найти угловые ускорения двух звеньев по ускорению третьего, которое часто просто равно нулю, если ведущее звено вращается равномерно. При решении имеет смысл воспользоваться совпадением определителей систем (2.9) и (2.10).

Очевидно простое обобщение уравнений трех угловых скоростей и уравнений трех угловых ускорений на большее число звеньев. Для этого достаточно изменить предел суммирования с 3 на число звеньев.

Сравнивая методы, заметим, что аналитический метод, как универсальный, имеющий простую формализацию в виде графов и дающий результат в проекциях, безусловно наиболее

предпочтителен при решении задач теоретической механики. Однако в некоторых очевидных случаях метод кинематических графов применять нецелесообразно. Например, при определении скорости точки A цилиндра, катящегося по неподвижной поверхности, по заданной скорости центра O (рис. 96) проще использовать понятие мгновенного центра скоростей (МЦС). Цилиндр катится без проскальзывания, поэтому точка P касания плоскости неподвижна. Следовательно, учитывая линейное распределение скоростей, получаем $v_{Ox} = v_{Ax}/2$.

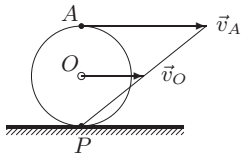


Рис. 96

2.3. Угловые скорости звеньев механизма

Задача 21. Плоский механизм состоит из пяти шарнирно соединенных стержней. Три шарнирные опоры крепят механизм к основанию. В указанном положении механизма (рис. 97) известна угловая скорость ведущего стержня OA : $\omega_{OA_z} = 12 \text{ с}^{-1}$. Даны длины стержней: $OA = 2 \text{ см}$, $AB = 9 \text{ см}$, $BC = 5 \text{ см}$, $DE = FE = 6 \text{ см}$, $AD = 3 \text{ см}$ и угол: $\cos \alpha = 4/5$. В данный момент стержень DE горизонтальный, стержни AB и FE вертикальные. Найти угловые скорости всех звеньев механизма.

Решение

Ведущий стержень OA и стержни FE и BC совершают вращательное движение, стержни AB и DE — плоское. Используем метод мгновенных центров скоростей. Вычисляем скорость точки A : $v_A = \omega_1 OA = 12 \cdot 2 = 24 \text{ см/с}$. Вектор скорости \vec{v}_A направляем перпендикулярно радиусу вращения OA вверх, принимая, что вращение ведущего звена происходит против часовой стрелки (рис. 98). В данной задаче определяем модули угловых скоростей, поэтому ответы не изменятся, если

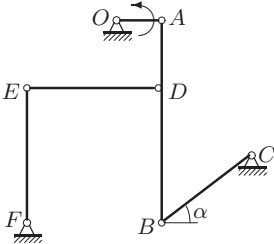


Рис. 97

предположить, что ведущее звено вращается по часовой стрелке. Направление вектора скорости \vec{v}_B известно — перпендикулярно стержню BC . Из двух возможных вариантов перпендикуляра к BC выбираем направление вектора вверх, что согласуется с теоремой о проекциях векторов скоростей неизменяемого отрезка (рис. 92, с. 57). Это поз-

воляет определить положение МЦС стержня AB . Находим точку P_1 пересечения перпендикуляров к векторам скоростей \vec{v}_A и \vec{v}_B .

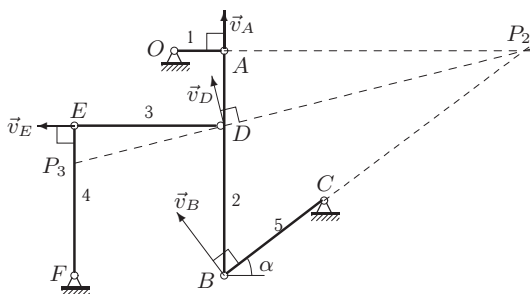


Рис. 98

Таким образом, стержень AB совершает мгновенное вращение вокруг P_2 с угловой скоростью ω_2 . Для скоростей точек этого стержня справедливы формулы скоростей точек при вращательном движении тела. Скорости пропорциональны расстояниям до МЦС:

$$v_A = \omega_2 AP_2, \quad (2.11)$$

$$v_B = \omega_2 BP_2, \quad (2.12)$$

$$v_D = \omega_2 DP_2. \quad (2.13)$$

Расстояния AP_2 , DP_2 , BP_2 легко вычислить. Сначала находим тригонометрические функции угла α : $\operatorname{tg} \alpha = 3/4$, $\sin \alpha = 3/5$. В прямоугольном треугольнике ABP_2 находим катет $AP_2 = AB \operatorname{ctg} \alpha = 9 \cdot 4/3 = 12$ см и гипотенузу $BP_2 = AB/\sin \alpha = 9 \cdot 5/3 = 15$ см. Имеем также $DP_2 = \sqrt{AP_2^2 + AD^2} = 3\sqrt{17}$ см. Далее определяем угловую скорость мгновенного вращения стержня AB вокруг его МЦС. Из уравнения (2.11) находим $\omega_2 = v_A/AP_2 = 24/12 = 2 \text{ с}^{-1}$. Согласно (2.12) $v_B = 2 \cdot 15 = 30$ см/с. Отсюда определяем угловую скорость вращения стержня BC :

$$\omega_1 = v_B/BC = 30/5 = 6 \text{ с}^{-1}.$$

По формуле (2.13) вычисляем также скорость

$$v_D = \omega_2 DP_2 = 6\sqrt{17} \text{ см/с}.$$

Переходим к стержню DE . Найдем МЦС стержня. Величина и направление вектора скорости точки D уже известны. Вектор \vec{v}_E перпендикулярен EF . Согласно его направлению с вектором \vec{v}_D , проекция которого на DE направлена налево, направляем \vec{v}_E в эту же

сторону. На пересечении перпендикуляров к векторам скоростей \vec{v}_E и \vec{v}_D находится точка P_3 — МЦС стержня DE . Скорости точек стержня удовлетворяют соотношениям

$$v_E = \omega_3 EP_3, \quad (2.14)$$

$$v_D = \omega_3 DP_3. \quad (2.15)$$

Расстояния EP_3 и DP_3 находим из подобия треугольников P_2DA и DP_3E : $AP_2/DE = 12/6 = DP_2/DP_3 = AD/EP_3$, откуда получаем расстояния до МЦС: $DP_3 = (3/2)\sqrt{17}$, $EP_3 = 1,5$ см. Из (2.15) находим угловую скорость: $\omega_3 = v_D/DP_3 = 4 \text{ с}^{-1}$. Вычисляем скорость: $v_E = \omega_3 EP_3 = 4 \cdot 1,5 = 6 \text{ см/с}$. Отсюда получаем угловую скорость стержня EF : $\omega_4 = v_E/EF = 6/6 = 1 \text{ с}^{-1}$.

Заметим, что с помощью метода МЦС находим только модули угловых скоростей. Знаки проекций угловых скоростей можно определить, например, методом кинематических графов, как это показано в следующем разделе.

Maple-программа расчета угловых скоростей методом графов дана на с. 361, а программа, иллюстрирующая движение рассчитываемого механизма, — на с. 363.

Задача 22. В указанном положении механизма (рис. 99) известна угловая скорость цилиндра $\omega_z = 2 \text{ с}^{-1}$. Радиус цилиндра $R = 4$ см, $OA = AB = 4$ см, $AD = BC = 8$ см, $EC = 4\sqrt{2}$ см, $\alpha = 45^\circ$. В данный момент стержни BC и AD горизонтальные, стержень OB вертикальный. Цилиндр катится по горизонтальной поверхности без проскальзывания. Найти угловые скорости всех звеньев механизма.

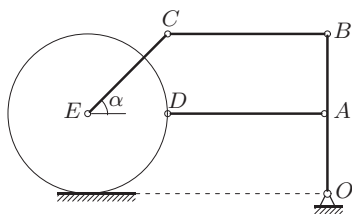


Рис. 99

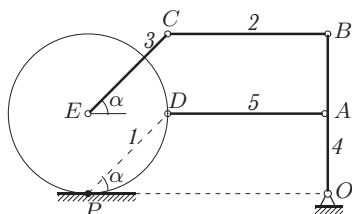


Рис. 100

Решение

Способ 1. Кинематические графы. Система состоит из пяти тел — четырех стержней и цилиндра. Пронумеруем тела (рис. 100). Цилиндр — 1, горизонтальные — 2 и 5, стержень CE — 3, вертикальный стержень OB — 4. Точка P касания цилиндра и поверхности является мгновенным центром скоростей цилиндра, если качение происходит без проскальзывания. Скорость этой точки равна нулю. Составляем

следующий кинематический граф: $P \xrightarrow{\frac{1}{\alpha}} D \xrightarrow{\frac{5}{0}} A \xrightarrow{\frac{4}{-\pi/2}} O$. Под стрелкой указываем угол к направлению перехода от точки к точке от положительного направления оси x против часовой стрелки (правая система координат — ось x направлена направо). Над стрелкой — номер тела¹. Ему соответствуют два уравнения:

$$v_{Ox} = v_{Px} - \omega_{1z} R \sqrt{2} \sin \alpha - \omega_{5z} AD \sin 0 - \omega_{4z} AO \sin(-\pi/2),$$

$$v_{Oy} = v_{Py} + \omega_{1z} R \sqrt{2} \cos \alpha + \omega_{5z} AD \cos 0 + \omega_{4z} AO \cos(-\pi/2).$$

С учетом $v_{Ox} = v_{Px} = 0$, $v_{Oy} = v_{Py} = 0$, $\omega_{1z} = 2 \text{ с}^{-1}$ получаем отсюда следующую линейную систему:

$$-8 + 4\omega_{4z} = 0,$$

$$8 + 4\omega_{5z} = 0.$$

Решаем систему уравнений и получаем $\omega_{4z} = 2 \text{ с}^{-1}$, $\omega_{5z} = -1 \text{ с}^{-1}$. Осталось найти еще две угловые скорости ω_{2z} и ω_{3z} . Для этого составляем граф $P \xrightarrow{\frac{1}{\pi/2}} E \xrightarrow{\frac{3}{\alpha}} C \xrightarrow{\frac{2}{0}} B \xrightarrow{\frac{4}{-\pi/2}} O$. Граф дает следующие уравнения в проекциях:

$$v_{Ox} = v_{Px} - \omega_{1z} R \sin(\pi/2) - \omega_{3z} R \sqrt{2} \sin \alpha - \omega_{2z} CB \sin 0 - \omega_{4z} BO \sin(-\pi/2),$$

$$v_{Oy} = v_{Py} + \omega_{1z} R \cos(\pi/2) + \omega_{3z} R \sqrt{2} \cos \alpha + \omega_{2z} CB \cos 0 + \omega_{4z} BO \cos(-\pi/2),$$

или, с учетом найденной угловой скорости $\omega_{4z} = 2 \text{ с}^{-1}$, получаем

$$8 - 4\omega_{3z} = 0,$$

$$4\omega_{3z} + 8\omega_{2z} = 0.$$

Решаем систему уравнений: $\omega_{2z} = -1 \text{ с}^{-1}$, $\omega_{3z} = 2 \text{ с}^{-1}$.

Способ 2. План скоростей. Вычисляем модули скоростей точек D и E . Эти точки принадлежат цилиндру, вращающемуся с известной угловой скоростью. Имеем: $v_E = \omega_1 R = 8 \text{ см/с}$, $v_D = \omega_1 R \sqrt{2} = 8\sqrt{2} \text{ см/с}$. Так как $\omega_{1z} > 0$, то вращение происходит против часовой стрелки. На плане скоростей (рис. 101) отмечаем произвольную точку o , от которой откладываем горизонтальный вектор длиной 8 и вектор \vec{v}_D под углом 45° к горизонту (в выбранном масштабе). Отмечаем точки e и d — концы векторов скоростей. Учитывая, что

¹Если пройти этот граф в обратном направлении, то соответствующая запись будет иметь вид $O \xrightarrow{\frac{4}{\pi/2}} A \xrightarrow{\frac{5}{\pi}} D \xrightarrow{\frac{1}{\pi+\alpha}} P$.



вектор \vec{v}_A перпендикулярен AO , проводим из точки o горизонтальную прямую. На этой прямой должна находиться точка A , при этом должно выполняться соотношение $ad \perp AD$. Проводим через d

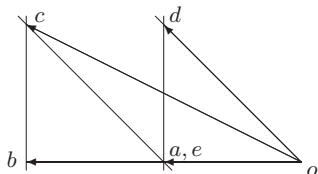


Рис. 101

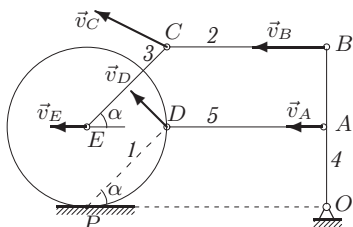


Рис. 102

вертикаль — получаем на пересечении двух прямых точку a . Она совпадает с точкой e . Таким образом, \vec{oa} есть скорость точки A . Очевидно, вектор скорости точки B горизонтальный и в два раза больше скорости точки A . Отмечаем точку b . По правилу плана скоростей $ec \perp EC$ и $bc \perp BC$. Проводим через b вертикаль, а через e прямую, перпендикулярную EC . Пересечение этих прямых определяет точку c . Все точки построены. Вектор \vec{v}_C — это \vec{oc} . Найдем модули угловых скоростей: $\omega_2 = \omega_{BC} = bc/BC = 1 \text{ с}^{-1}$, $\omega_3 = \omega_{EC} = ec/EC = 2 \text{ с}^{-1}$, $\omega_4 = \omega_{AO} = ao/AO = 2 \text{ с}^{-1}$, $\omega_5 = \omega_{AD} = ad/AD = 1 \text{ с}^{-1}$. Найденные векторы скоростей обозначим на механизме (рис. 102) и еще раз уточним: векторы скоростей (кроме \vec{v}_E и \vec{v}_D) на рисунке 102 получены по плану скоростей (рис. 101), а не наоборот.

Способ 3. Уравнения трех угловых скоростей. Если принять PD за условное звено, то фигура $PDAO$ образует четырехзвенник. Запишем уравнения трех угловых скоростей:

$$\begin{aligned}\omega_{1z}(x_D - x_P) + \omega_{5z}(x_A - x_D) + \omega_{4z}(x_O - x_A) &= 0, \\ \omega_{1z}(y_D - y_P) + \omega_{5z}(y_A - y_D) + \omega_{4z}(y_O - y_A) &= 0.\end{aligned}$$

Начало координат выберем в P . Координаты точек получаются следующие:

$$\begin{aligned}y_P = y_O = x_P = x_E = 0, \quad y_E = y_D = y_A = x_D = x_C = R, \\ y_C = y_B = 8 \text{ см}, \quad x_B = x_O = x_A = 12 \text{ см}.\end{aligned}$$

Из второго уравнения сразу получаем $\omega_{4z} = 2 \text{ с}^{-1}$, затем из первого находим $\omega_{5z} = -\omega_{1z}/2 = -1 \text{ с}^{-1}$.

Для того чтобы найти угловые скорости стержней 2 и 3, используем уравнения «трех угловых скоростей» в обобщенном виде — для *четырёх* угловых скоростей. Структура уравнений та же, просто добавится

еще одно слагаемое. Имеем для многозвенника *PECBO*:

$$\omega_{1z}(x_E - x_P) + \omega_{3z}(x_C - x_E) + \omega_{2z}(x_B - x_C) + \omega_{4z}(x_O - x_B) = 0,$$

$$\omega_{1z}(y_E - y_P) + \omega_{3z}(y_C - y_E) + \omega_{2z}(y_B - y_C) + \omega_{4z}(y_O - y_B) = 0.$$

Подставим значения координат. Получим простую систему

$$4\omega_{3z} + 8\omega_{2z} = 0,$$

$$4\omega_{1z} + 4\omega_{3z} - 16 = 0.$$

Решение этой системы получается, как и раньше: $\omega_{2z} = -1 \text{ с}^{-1}$, $\omega_{3z} = 2 \text{ с}^{-1}$.

Maple-программа решения этой задачи (а это уже четвертый способ решения!) дана на с. 365.

Задача 23. В указанном положении механизма, состоящего из цилиндра и четырех стержней (рис. 103), известна угловая скорость $\omega_{1z} = 20 \text{ с}^{-1}$ стержня *OA*. Радиус цилиндра $R = 3 \text{ см}$, $OA = 6 \text{ см}$, $AB = 4\sqrt{2} \text{ см}$, $AC = 10 \text{ см}$, $CD = 4 \text{ см}$, $R = 3 \text{ см}$. В данный момент стержни *CD* и *OA* вертикальные, стержень *CA* — горизонтальный. Цилиндр катится по горизонтальной поверхности без проскальзывания. Найти угловые скорости всех звеньев механизма.

Решение

Пронумеруем звенья (рис. 104).

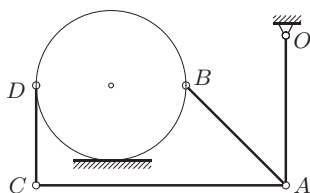


Рис. 103

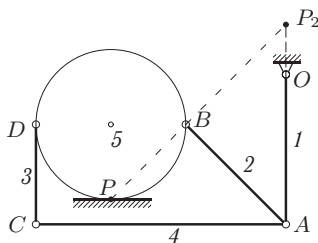


Рис. 104

Точка касания цилиндра и плоскости *P* является мгновенным центром скоростей цилиндра. Составляем граф $O \xrightarrow{-\pi/2} A \xrightarrow{2} B \xrightarrow{5} P$. Граф дает следующие уравнения в проекциях:

$$v_{Px} = v_{Ox} - \omega_{1z}OA \sin(-\pi/2) - \omega_{2z}AB \sin(3\pi/4) - \omega_{5z}R\sqrt{2} \sin(5\pi/4),$$

$$v_{Py} = v_{Oy} + \omega_{1z}OA \cos(-\pi/2) + \omega_{2z}AB \cos(3\pi/4) + \omega_{5z}R\sqrt{2} \cos(5\pi/4),$$

Отсюда с учетом известной угловой скорости $\omega_{1z} = 20 \text{ с}^{-1}$ получаем

$$\begin{aligned} 120 - 4\omega_{2z} + 3\omega_{5z} &= 0, \\ -4\omega_{2z} - 3\omega_{5z} &= 0. \end{aligned}$$

Решаем систему уравнений: $\omega_{2z} = 15 \text{ с}^{-1}$, $\omega_{5z} = -20 \text{ с}^{-1}$.

Составляем граф $P \xrightarrow{5/3\pi/4} D \xrightarrow{3/-\pi/2} C \xrightarrow{4/0} A \xrightarrow{1/\pi/2} O$. Граф дает следующие уравнения в проекциях:

$$\begin{aligned} v_{Ox} &= v_{Px} - \omega_{5z} R\sqrt{2} \sin(3\pi/4) - \omega_{3z} CD \sin(-\pi/2) - \omega_{4z} AC \sin(0) - \\ &\quad - \omega_{1z} OA \sin(\pi/2), \\ v_{Oy} &= v_{Py} + \omega_{5z} R\sqrt{2} \cos(3\pi/4) + \omega_{3z} CD \cos(-\pi/2) + \omega_{4z} AC \cos(0) + \\ &\quad + \omega_{1z} OA \cos(\pi/2). \end{aligned}$$

Получаем два уравнения:

$$\begin{aligned} -60 + 4\omega_{3z} &= 0, \\ 60 + 10\omega_{4z} &= 0. \end{aligned}$$

Решаем уравнения и находим: $\omega_{3z} = 15 \text{ с}^{-1}$, $\omega_{4z} = -6 \text{ с}^{-1}$. Заметим, что эту задачу нельзя решить с помощью МЦС. Если для звена AB легко найти МЦС (в точке P_2), то МЦС звеньев AC и CD найти непосредственно затруднительно.

Maple-программа решения этой задачи дана на с. 365.

Задача 24. Плоский механизм состоит из шести шарнирно соединенных стержней, закрепленных на трех опорах (рис. 105).

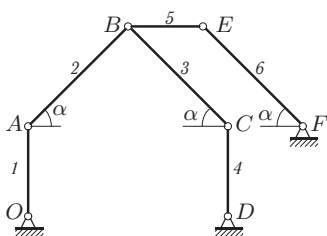


Рис. 105

Заданы угловые скорости двух его звеньев: $\omega_{CDz} = -3 \text{ с}^{-1}$, $\omega_{FEz} = 3 \text{ с}^{-1}$ и длины звеньев: $OA = DC = 4 \text{ см}$, $AB = BC = EF = 4\sqrt{2} \text{ см}$, $BE = 3 \text{ см}$. В указанном положении механизма стержни OA , CD вертикальные, BE — горизонтальный; $\alpha = 45^\circ$. Найти в этот момент угловые скорости всех звеньев механизма.

Решение

Применим аналитический метод расчета. Пронумеруем стержни механизма (рис. 105) и составим кинематический граф:

$$O \xrightarrow{1/\pi/2} A \xrightarrow{2/\alpha} B \xrightarrow{3/-\alpha} C \xrightarrow{4/-\pi/2} D.$$

Соответствующие уравнения для скоростей в проекциях на оси имеют вид

$$\begin{aligned} v_{Dx} &= v_{Ox} - OA\omega_{1z} \sin(\pi/2) - AB\omega_{2z} \sin(\pi/4) - BC\omega_{3z} \sin(-\pi/4) - \\ &\quad - CD\omega_{4z} \sin(-\pi/2), \\ v_{Dy} &= v_{Oy} + OA\omega_{1z} \cos(\pi/2) + AB\omega_{2z} \cos(\pi/4) + BC\omega_{3z} \cos(-\pi/4) + \\ &\quad + CD\omega_{4z} \cos(-\pi/2). \end{aligned}$$

Упростим уравнения, учитывая, что $v_{Dx} = v_{Ox} = v_{Dy} = v_{Oy} = 0$:

$$\begin{aligned} 0 &= -OA\omega_{1z} - AB\omega_{2z} \sin(\pi/4) + BC\omega_{3z} \sin(\pi/4) + CD\omega_{4z}, \\ 0 &= AB\omega_{2z} \cos(\pi/4) + BC\omega_{3z} \cos(\pi/4). \end{aligned} \quad (2.16)$$

Составляем граф $D \xrightarrow[\pi/2]{4} C \xrightarrow[3\pi/4]{3} B \xrightarrow[0]{5} E \xrightarrow[-\pi/4]{6} F$. Записываем соответствующие уравнения для скоростей:

$$\begin{aligned} v_{Fx} &= v_{Dx} - DC\omega_{4z} \sin(\pi/2) - CB\omega_{3z} \sin(3\pi/4) - BE\omega_{5z} \sin 0 - \\ &\quad - EF\omega_{6z} \sin(-\pi/4), \\ v_{Fy} &= v_{Dy} + DC\omega_{4z} \cos(\pi/2) + CB\omega_{3z} \cos(3\pi/4) + BE\omega_{5z} \cos 0 + \\ &\quad + EF\omega_{6z} \cos(-\pi/4). \end{aligned}$$

Упростим уравнения:

$$0 = -DC\omega_{4z} - CB\omega_{3z} \sin(3\pi/4) + EF\omega_{6z} \sin(\pi/4), \quad (2.17)$$

$$0 = CB\omega_{3z} \cos(3\pi/4) + BE\omega_{5z} + EF\omega_{6z} \cos(\pi/4). \quad (2.18)$$

В системе (2.16–2.18) известны угловые скорости $\omega_{4z} = -3 \text{ с}^{-1}$, $\omega_{6z} = 3 \text{ с}^{-1}$. После подстановки численных значений имеем

$$\begin{aligned} -\omega_{1z} - \omega_{2z} + \omega_{3z} - 3 &= 0, \\ \omega_{2z} + \omega_{3z} &= 0, \\ 6 - \omega_{3z} &= 0, \\ -4\omega_{3z} + 3\omega_{5z} + 12 &= 0. \end{aligned} \quad (2.19)$$

Решаем систему уравнений (2.19). Находим угловые скорости: $\omega_{1z} = 9 \text{ с}^{-1}$, $\omega_{2z} = -6 \text{ с}^{-1}$, $\omega_{3z} = 6 \text{ с}^{-1}$, $\omega_{5z} = 4 \text{ с}^{-1}$.

Другой способ решения задачи — построение плана скоростей [27]. Вычисляем модули скоростей точек E и C (рис. 106). Направления векторов выбираем с учетом знаков угловых скоростей (положительная угловая скорость соответствует вращению *против* часовой стрелки). На рис. 107 изображен план скоростей механизма.

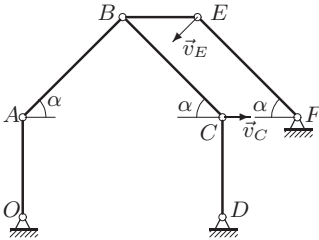


Рис. 106

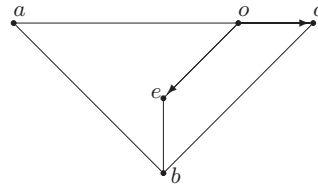


Рис. 107

План скоростей строим следующим образом. От произвольной точки o откладываем две известные скорости \vec{v}_C и \vec{v}_E . Их модули равны $v_C = \omega_{CD} \cdot CD = 12$ см/с и $v_E = \omega_{EF} \cdot EF = 12\sqrt{2}$ см/с. Направления скоростей перпендикулярны стержням CD и EF соответственно. С учетом знаков угловых скоростей отмечаем точки плана скоростей e и c (концы векторов). По правилу построения плана скоростей $BE \perp be$, $BC \perp bc$. На пересечении направлений be и bc получаем положение точки b . Отрезок ob (не обозначен на плане) определяет величину и направление скорости \vec{v}_B . Аналогично находим положение точки a . Имеем: $AO \perp ao$, $AB \perp ab$. Решая простую геометрическую задачу, находим: $ab = 24\sqrt{2}$, $be = 12$, $bc = 24\sqrt{2}$, $ao = 36$. Модули угловых скоростей получаем по простым формулам: $\omega_2 = ab/AB$, $\omega_{BE} = be/BE$, $\omega_3 = bc/BC$, $\omega_{AO} = ao/AO$.

Заметим, что с помощью мгновенных центров скоростей эту задачу решить нельзя.

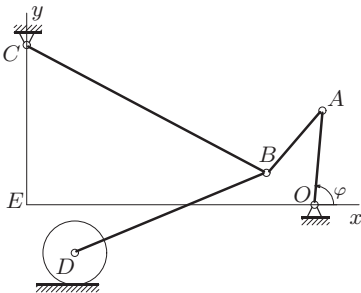


Рис. 108

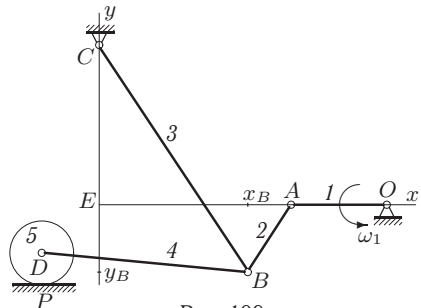


Рис. 109

Задача 25. Механизм, состоящий из четырех шарнирно соединенных стержней и диска, изображен в произвольном положении, определяемом некоторым углом φ (рис. 108). Диск катится без проскальзывания по горизонтальной плоскости. Заданы угловая скорость $\omega_{OA_z} = 56$ с⁻¹, длины стержней $OA = 6$ см, $AB = 5$ см, $BC = 17$ см, $BD =$

= 13 см и радиус диска $R = 1$ см. Известны координаты неподвижных шарниров: $x_C = 0$, $y_C = 10$ см, $x_O = 18$ см, $y_O = 0$ и $y_D = -3$ см в осях с началом в точке E . Найти угловые скорости всех звеньев механизма и скорость центра диска при $\varphi = \pi$.

Решение

Решение задачи состоит из двух частей¹. В первой части геометрическими методами определяем координаты шарниров механизма в заданном положении. Во второй части находим угловые скорости звеньев и скорость оси диска.

Изобразим механизм при $\varphi = \pi$ (рис. 109). Пронумеруем звенья механизма. Координаты шарнира A при $\varphi = \pi$ известны: $x_A = 18 - 6 = 12$ см, $y_A = 0$ см. Координаты шарнира B найдем из следующей системы уравнений:

$$\begin{aligned}x_B^2 + (y_C - y_B)^2 &= BC^2, \\(x_A - x_B)^2 + (y_A - y_B)^2 &= AB^2.\end{aligned}\tag{2.20}$$

Подставляя сюда известные координаты и длины, получим

$$\begin{aligned}(12 - x_B)^2 + y_B^2 &= 25, \\x_B^2 + (10 - y_B)^2 &= 289.\end{aligned}$$

Решаем эту систему уравнений. Уравнение имеет два решения²: $x_{B_1} = 15,666$, $y_{B_1} = 3,399$, $x_{B_2} = 9,317$, $y_{B_2} = -4,219$. Первое решение, где $y_B > 0$, не согласуется с условием. Берем второе решение.

Угловые скорости звеньев 2 и 3 находим из уравнений трех угловых скоростей³:

$$\begin{aligned}(x_C - x_B)\omega_{3_z} + (x_B - x_A)\omega_{2_z} + (x_A - x_O)\omega_{1_z} &= 0, \\(y_C - y_B)\omega_{3_z} + (y_B - y_A)\omega_{2_z} + (y_A - y_O)\omega_{1_z} &= 0.\end{aligned}$$

Подставляя сюда известные координаты и угловую скорость $\omega_{1_z} = \omega_1 = 56$, получим решение $\omega_{2_z} = -61,679 \text{ с}^{-1}$, $\omega_{3_z} = -18,302 \text{ с}^{-1}$.

В точке P касания плоскости диска (тело 5) скорость равна нулю. Это позволяет рассмотреть диск как часть условного четырехзвенника $CBDP$ и записать уравнения трех угловых скоростей:

$$\begin{aligned}(x_C - x_B)\omega_{3_z} + (x_B - x_D)\omega_{4_z} + (x_D - x_P)\omega_{5_z} &= 0, \\(y_C - y_B)\omega_{3_z} + (y_B - y_D)\omega_{4_z} + (y_D - y_P)\omega_{5_z} &= 0.\end{aligned}$$

¹При $\varphi = \pi/2$ эта задача решена в [28].

²Задача о пересечении двух окружностей: одна — радиусом 17 см и с центром в C , другая — радиусом 5 см с центром в A .

³См. с. 73.

Находим координату $x_D = x_B - \sqrt{BD^2 - (y_B - y_D)^2} = 9,317 - \sqrt{13^2 - (-4,219 + 3)^2} = -3,625$ см. Решаем систему и находим $\omega_{4z} = -13,175 \text{ c}^{-1}$, $\omega_{5z} = 244,182 \text{ c}^{-1}$.

Скорость центра диска $v_D = R\omega_{5z} = 244,182$ см/с.

Замечание. Систему уравнений (2.20) для определения координат легче решить в системе **Maple**:

```
> eq1 := (Xa-Xb)^2 + (Ya-Yb)^2 = AB^2;
```

```
> eq2 := (Xc-Xb)^2 + (Yc-Yb)^2 = CB^2;
```

```
> solve({eq1, eq2, Yb < 0}, {Xb, Yb});
```

Дополнительное условие $Yb < 0$ позволяет сразу же получить единственное геометрически допустимое решение. Найти точки пересечения окружностей можно также в пакете **geometry**. Подключив этот пакет с помощью команды **with**, надо задать центры окружностей **point(A, 12, 0)**, **point(C, 0, 10)**, окружности **circle(o1, [A, 5.])**, **circle(o2, [C, 17.])**. Затем надо определить точки пересечения **intersection(B, o1, o2)** и вычислить их координаты **coordinates(B[1])**, **coordinates(B[2])**. Ответ будет тем же. Для изображения геометрических объектов служит оператор **draw([o2, o1])**.

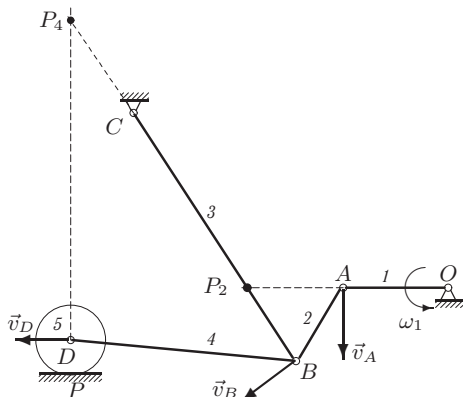


Рис. 110

Проверим решение с помощью мгновенных центров скоростей [27]. Вектор скорости точки A перпендикулярен радиусу OA и направлен вниз, так как по условию вращение ведущего звена № 1 направлено против часовой стрелки (рис. 110). Значение скорости: $v_A = \omega_1 OA$. Вектор скорости точки B перпендикулярен радиусу BC . Перпендикуляры к векторам скоростей концов отрезка AB определяют

положение МЦС звена 2. Точка P_2 пересечения перпендикуляров есть МЦС звена 2. Отсюда имеем два уравнения: $v_A = \omega_2 P_2 A$, $v_B = \omega_2 P_2 B$. Из первого уравнения находим угловую скорость $\omega_2 = v_A / P_2 A$, из второго — скорость $v_B = v_A (P_2 B / P_2 A)$. Расстояния до мгновенных центров скоростей можно найти и приближенно, измерив их на чертеже, выполненном в масштабе. Точность будет, конечно, невысокой, но для проверочного решения этого обычно достаточно. Аналогично

находим P_4 — МЦС звена 4 — точку пересечения перпендикуляров к векторам скоростей \vec{v}_D и \vec{v}_B . Записываем соответствующие уравнения: $v_B = \omega_4 P_4 B$, $v_D = \omega_4 P_4 D$, из которых легко найти угловую скорость $\omega_4 = v_B / P_4 B = v_A P_2 B / (P_4 B \cdot P_2 A)$ и скорость оси D : $v_D = \omega_4 P_4 D$. Другой способ проверить решение — построение плана скоростей. Этот метод, как и метод МЦС, также является графическим (см. примеры на с. 64, 68).

2.4. Угловые ускорения звеньев механизма

Задача 26. Механизм состоит из трех шарнирно скрепленных стержней. В указанном положении механизма (рис. 111) известна постоянная угловая скорость $\omega_{OA_z} = 4 \text{ с}^{-1}$ стержня OA . Известны длины стержней $OA = 5 \text{ см}$, $AB = BC = 4 \text{ см}$. Размеры на рисунке даны в сантиметрах. Найти угловые ускорения звеньев AB и BC .

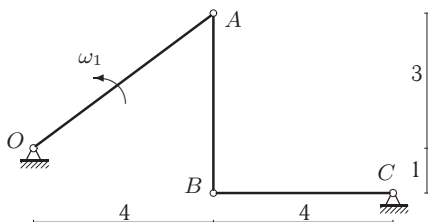


Рис. 111

Решение

Дадим два метода решения этой задачи.

1. *Метод МЦС и многоугольника ускорений.* Определим угловые скорости звеньев. Пронумеруем стержни (рис. 112). Начнем с вычисления скорости точки A , принадлежащей вращающемуся телу OA .

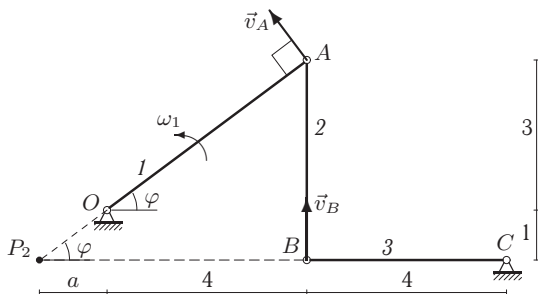


Рис. 112

Известна угловая скорость ведущего звена $\omega_1 = \omega_{OA_z} = 4 \text{ с}^{-1}$. Имеем $v_A = \omega_1 OA = 4 \cdot 5 = 20 \text{ см/с}$. Найдем положение МЦС звена AB .

Стержень OA вращается вокруг шарнира O , стержень BC — вокруг шарнира C . Векторы скоростей точек A и B направлены перпендикулярно радиусам вращения. Направление вектора \vec{v}_A задано по условию против часовой стрелки, вверх (т.к. $\omega_{1z} > 0$), вектор \vec{v}_B направляем вверх, исходя из теоремы о проекциях векторов скоростей неизменяемого отрезка¹. МЦС звена AB лежит на пересечении перпендикуляров к скоростям точек отрезка AB , обозначим его P_2 .

Найдем расстояние a , определяющее положение МЦС. Очевидно, $\operatorname{tg} \varphi = 3/4 = 1/a$. Следовательно, $a = 4/3 = 1,33$ м.

Скорости точек звена AB удовлетворяют соотношениям $v_A = \omega_2 AP_2$, $v_B = \omega_2 BP_2$. Из первого соотношения при известной скорости $v_A = 20$ см/с с учетом $AP_2 = \sqrt{4^2 + (4 + 4/3)^2} = 20/3$ см найдем $\omega_2 = v_A/AP_2 = 20/(20/3) = 3$ с⁻¹. Вычислим скорость точки B : $v_B = \omega_2 BP_2 = 3 \cdot (4 + 4/3) = 16$ см/с. Угловая скорость стержня BC равна $\omega_3 = v_B/BC = 16/4 = 4$ с⁻¹.

Вычислим угловые ускорения стержней. Начнем с ускорения точки A . Стержень OA вращается равномерно. Полагая в формуле $a_A = OA\sqrt{\omega_1^4 + \varepsilon_1^2}$ для ускорения точки тела при вращательном

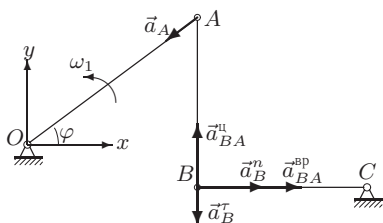


Рис. 113

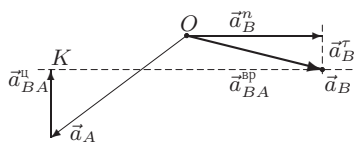


Рис. 114

движении $\varepsilon_1 = 0$, получаем $a_A = OA\omega_1^2 = 5 \cdot 16 = 80$ см/с². Вектор ускорения направлен к центру O (рис. 113). Для вычисления ускорения точки B воспользуемся формулой Ривальса

$$\vec{a}_B = \vec{a}_A + \vec{a}_{BA}^{\text{BP}} + \vec{a}_{BA}^{\text{u}}. \quad (2.21)$$

Здесь $a_{BA}^{\text{u}} = \omega_2^2 AB = 3^2 \cdot 4 = 36$ см/с², $a_{BA}^{\text{BP}} = \varepsilon_2 AB$. Направление векторов показано на рисунке 113. С учетом того, что точка B движется по окружности радиусом BC с центром в C , вектор \vec{a}_B представим в

¹ **Теорема.** Проекции скоростей точек отрезка твердого тела на ось, лежащую на отрезке, совпадают (рис. 92, с. 57).

виде суммы $\vec{a}_B = \vec{a}_B^n + \vec{a}_B^r$, где $a_B^n = \omega_3^2 BC = 16 \cdot 4 = 64 \text{ см/с}^2$, $a_B^r = \varepsilon_3 BC$. Запишем (2.21) в проекциях на оси координат:

$$\begin{aligned} a_B^n &= -a_A \cos \varphi + a_{BA}^{BP}, \\ -a_B^r &= -a_A \sin \varphi + a_{BA}^u. \end{aligned}$$

Из системы уравнений получаем $a_{BA}^{BP} = 64 + 80 \cdot 0,8 = 128 \text{ см/с}^2$, $a_B^r = 80 \cdot 0,6 - 36 = 12 \text{ см/с}^2$. Отсюда находим угловые ускорения: $\varepsilon_2 = a_{BA}^{BP}/AB = 128/4 = 32 \text{ с}^{-2}$, $\varepsilon_3 = a_B^r/BC = 12/4 = 3 \text{ с}^{-2}$.

Векторное уравнение (2.21) относительно ε_2 , ε_3 можно решить графически. Выберем на чертеже произвольную точку O , от которой будем откладывать векторные суммы (рис. 114). Начнем с векторов \vec{a}_A и \vec{a}_{BA}^u , известных и по величине, и по направлению. Последовательно откладывая их от O (начало одного к концу другого¹), получим точку K . Осталось добавить вектор \vec{a}_{BA}^{BP} известного направления (перпендикулярно \vec{a}_{BA}^u) и результат, т. е. вектор \vec{a}_B , будет получен. Однако длина вектора \vec{a}_{BA}^{BP} неизвестна. Поэтому обозначим пунктиром место возможного положения конца вектора \vec{a}_B и отложим от той же точки O сумму $\vec{a}_B^n + \vec{a}_B^r = \vec{a}_B$. Вектор \vec{a}_B^n известен и по величине, и по направлению, а вектор \vec{a}_B^r — только по направлению, $\vec{a}_B^r \perp \vec{a}_B^n$. Проводим вертикальную пунктирную линию к концу \vec{a}_B^n . Пересечение двух пунктирных линий определяет конец искомого вектора \vec{a}_B и одновременно величины \vec{a}_B^r и \vec{a}_{BA}^{BP} , по которым можно найти угловые ускорения.

2. Уравнения трех угловых

скоростей и ускорений. Вводим систему координат, нумеруем стержни и шарниры. Номера шарниров приписываем нижним индексом к соответствующим буквам (рис. 115). Определяем координаты шарниров (в сантиметрах): $x_1 = 0$, $y_1 = 0$, $x_2 = 4$, $y_2 = 3$, $x_3 = 4$, $y_3 = -1$, $x_4 = 8$, $y_4 = -1$.

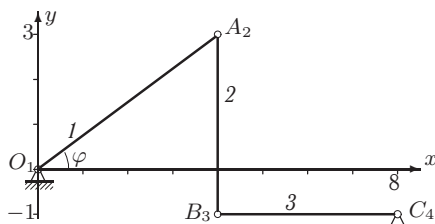


Рис. 115

Записываем уравнения трех угловых скоростей (2.9):

$$\begin{aligned} \omega_{1z}(x_2 - x_1) + \omega_{2z}(x_3 - x_2) + \omega_{3z}(x_4 - x_3) &= 0, \\ \omega_{1z}(y_2 - y_1) + \omega_{2z}(y_3 - y_2) + \omega_{3z}(y_4 - y_3) &= 0, \end{aligned} \quad (2.22)$$

где x_i, y_i и x_{i+1}, y_{i+1} , $i = 1, 2, 3$, — координаты шарниров на концах звена, имеющего угловую скорость ω_{iz} . Шарниры 1 и 4 — опорные.

¹Метод "Head and tail", или метод векторного многоугольника.

С учетом известных координат имеем отсюда

$$\begin{aligned} 4\omega_{1z} + 4\omega_{3z} &= 0, \\ 3\omega_{1z} - 4\omega_{2z} &= 0. \end{aligned}$$

Решаем систему с учетом известной угловой скорости $\omega_{1z} = 4 \text{ с}^{-1}$ и находим $\omega_{2z} = 3 \text{ с}^{-1}$, $\omega_{3z} = -4 \text{ с}^{-1}$.

Записываем уравнения трех угловых ускорений (2.10):

$$\begin{aligned} \varepsilon_{1z}(x_2 - x_1) + \varepsilon_{2z}(x_3 - x_2) + \varepsilon_{3z}(x_4 - x_3) - \\ - \omega_{1z}^2(y_2 - y_1) - \omega_{2z}^2(y_3 - y_2) - \omega_{3z}^2(y_4 - y_3) &= 0, \\ \varepsilon_{1z}(y_2 - y_1) + \varepsilon_{2z}(y_3 - y_2) + \varepsilon_{3z}(y_4 - y_3) + \\ + \omega_{1z}^2(x_2 - x_1) + \omega_{2z}^2(x_3 - x_2) + \omega_{3z}^2(x_4 - x_3) &= 0. \end{aligned}$$

С учетом известных координат и $\varepsilon_{1z} = 0$ эти уравнения примут вид

$$\begin{aligned} 4\varepsilon_{3z} - 3\omega_{1z}^2 + 4\omega_{2z}^2 &= 0, \\ -4\varepsilon_{2z} + 4\omega_{1z}^2 + 4\omega_{3z}^2 &= 0. \end{aligned}$$

Подставляя найденные угловые скорости, находим $\varepsilon_{2z} = 32 \text{ с}^{-2}$, $\varepsilon_{3z} = 3 \text{ с}^{-2}$. Ответы получаются те же.

Заметим, что вместо уравнений трех угловых скоростей можно использовать граф $O \xrightarrow[1]{\varphi} A \xrightarrow[2]{-\pi/2} B \xrightarrow[3]{0} C$.

Maple-программа для решения этой задачи дана на с. 366.

2.5. Механизм с муфтой

Задача 27. Стержень BD шарнирно соединен с горизонтальным ползуном B и проходит сквозь муфту C , вращающуюся на стержне

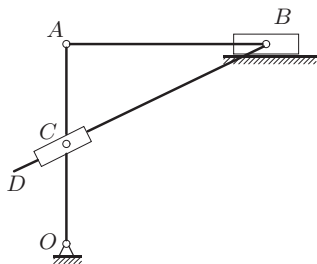


Рис. 116

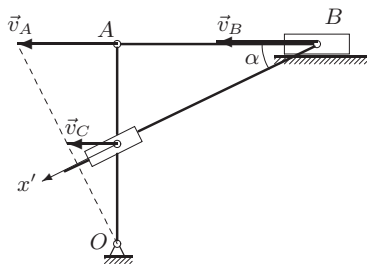


Рис. 117

OA (рис. 116). Угловая скорость OA постоянна и равна ω . Дано: $OC = CA = l$, $AB = 2l$, $AB \perp AO$. Найти скорость и ускорение стержня BD относительно муфты в указанном положении.

Решение

Муфту C будем рассматривать как точку, совершающую сложное движение: относительное движение по стержню и переносное движение вместе со стержнем. Абсолютное движение — движение точки C по окружности с центром в точке O с постоянной скоростью ωl .

1. *Скорость.* Абсолютная скорость C (скорость точки C на стержне OA) равна по модулю ωl (рис. 117). Переносной скоростью является скорость точки C стержня BD , совпадающей в данный момент с муфтой. Она выражается по формуле для скоростей точек тела при плоском движении через скорость полюса B : $\vec{v}_C = \vec{v}_B + \vec{v}_{CB}$, где $\vec{v}_{CB} = \vec{\omega}_{CB} \times \vec{BC}$. Так как AB в данный момент движется мгновенно-поступательно (МЦС стержня — в бесконечности), то $v_B = v_A = 2\omega l$. С другой стороны, по формуле сложения скоростей для сложного движения точки

$$\vec{v}_C = \vec{v}_n + \vec{v}_{от} = \vec{v}_B + \vec{v}_{CB} + \vec{v}_{от}. \quad (2.23)$$

В этом векторном уравнении две неизвестных величины — модули векторов $\vec{v}_{от}$ и \vec{v}_{CB} . Направления всех векторов известны. Это позволяет решить уравнение графически.

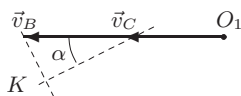


Рис. 118

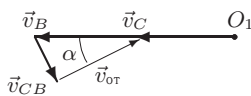


Рис. 119

Начнем с суммы векторов в *правой* части уравнения (2.23). Отложим от некоторой точки O_1 вектор \vec{v}_B известной длины (рис. 118). От его конца проведем прямую, перпендикулярную BC , — на этой прямой лежит вектор \vec{v}_{CB} . Обозначим ее пунктиром. К концу вектора \vec{v}_{CB} надо провести прямую, параллельную BC . Эта прямая дает направление последнего слагаемого в правой части (2.23) — относительной скорости $\vec{v}_{от}$. *Левая* часть уравнения — это вектор \vec{v}_C . Конец вектора $\vec{v}_{от}$ должен попасть в конец вектора \vec{v}_C , отложенного от той же точки O_1 , что и \vec{v}_B . Тем самым утверждается равенство левой и правой части в (2.23). Проведем вектор \vec{v}_C . Его длина известна и равна ωl . Через конец \vec{v}_C проведем пунктир под углом α до пересечения его с первым пунктиром. Точка пересечения двух пунктирных линий K определяет величины скоростей $\vec{v}_{от}$ и \vec{v}_{CB} (рис. 119). Из прямоугольного треугольника получаем значения

$$v_{от} = |\vec{v}_B - \vec{v}_C| \cos \alpha = \omega l \cos \alpha = (2/\sqrt{5})\omega l,$$

$$v_{CB} = |\vec{v}_B - \vec{v}_C| \sin \alpha = \omega l \sin \alpha = (1/\sqrt{5})\omega l.$$

Уравнение (2.23) можно решить аналитически. Спроецируем его на ось x' , направленную по стержню BC . В полученное соотношение не войдет слагаемое $\vec{v}_{CB} \perp x'$, содержащее неизвестную угловую скорость ω_{CB} . Получим уравнение $\omega l \cos \alpha = -v_{от} + 2l\omega \cos \alpha$, откуда получаем тот же результат: $v_{от} = (2/\sqrt{5})\omega l$.

Для определения ускорения потребуется значение угловой скорости стержня BC (переносная угловая скорость). Так как $BC = l\sqrt{5}$, то

$$\omega_{п} = v_{CB}/BC = \omega/5. \quad (2.24)$$

2. Ускорение. Так как угловая скорость стержня OA постоянна, то $\varepsilon = 0$, и ускорение точки C в абсолютном движении имеет только центростремительную компоненту: $a_C = \omega^2 l$. По теореме о сложении ускорений (теорема Кориолиса¹)

$$\vec{a}_C = \vec{a}_{от} + \vec{a}_{п} + \vec{a}_K, \quad (2.25)$$

где $\vec{a}_{от}$ — искомое ускорение, $\vec{a}_{п}$ — переносное ускорение,

$$\vec{a}_K = 2\vec{\omega}_{п} \times \vec{v}_{от} \quad (2.26)$$

— ускорение Кориолиса. Рассмотрим переносное ускорение — ускорение точки C стержня, совершающего плоское движение (рис. 120).

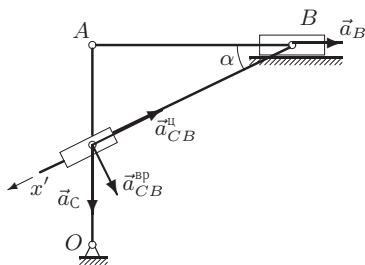


Рис. 120

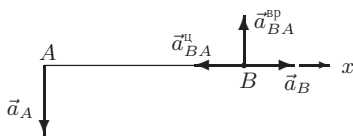


Рис. 121

Возьмем точку B за полюс. Согласно формуле Ривальса (2.6) имеем

$$\vec{a}_{п} = \vec{a}_B + \vec{a}_{CB}^п + \vec{a}_{CB}^вп. \quad (2.27)$$

Заметим, что в этой задаче получается две точки C и, соответственно, два ускорения \vec{a}_C . Будем различать точку C — ось, на которой вращается муфта, и точку C , принадлежащую стержню. Ускорение первой точки — абсолютное ускорение, ускорение второй — переносное. Ускорение \vec{a}_B вычислим через ускорение \vec{a}_A . Вектор ускорения ползуна, движущегося прямолинейно, направлен по линии скольжения

¹Gaspar Gustav de'Coriolis (1792–1843) — французский математик.

горизонтально. Очевидно, при $\omega = \text{const}$ имеем $a_A = 2l\omega^2$ и $\vec{a}_A \perp AB$ (рис. 121). Пользуясь формулой Ривальса, записываем

$$\vec{a}_B = \vec{a}_A + \vec{a}_{BA}^{\text{н}} + \vec{a}_{BA}^{\text{вп}}. \quad (2.28)$$

Здесь $a_{BA}^{\text{н}} = \omega_{AB}^2 AB = 0$, так как стержень движется мгновенно поступательно и $\omega_{AB} = 0$. Из проекции (2.28) на ось x сразу же получаем $a_B = 0$.

Вернемся к уравнению (2.27). Центростремительное ускорение $a_{CB}^{\text{н}}$ можно легко найти. С учетом уже известной переносной угловой скорости (2.24) имеем $a_{CB}^{\text{н}} = \omega_{\text{п}}^2 CB = (\omega/5)^2 l\sqrt{5}$. Величину ускорения Кориолиса вычисляем по формуле (2.26). Направление этого вектора определяем по правилу Жуковского¹ поворотом на 90° вектора относительной скорости по направлению переносного вращения. В данном случае вращение происходит против часовой стрелки (это определяется направлением вектора \vec{v}_{CB} — он оказался направлен вниз (рис. 119), указывая тем самым поворот стержня вокруг B против часовой стрелки). Перепишем (2.25) в виде $\vec{a}_C = \vec{a}_{\text{от}} + \vec{a}_K + \vec{a}_{CB}^{\text{н}} + \vec{a}_{CB}^{\text{вп}}$. Проецируем это уравнение на ось x' (рис. 122) и получаем $a_C \sin \alpha = a_{\text{от}} - a_{CB}^{\text{н}}$. Отсюда находим $a_{\text{от}} = (6\sqrt{5}/25)\omega^2 l$. Еще один способ вычисления скоростей точек механизмов с муфтой, справедливый в некоторых частных случаях, описан в решении задачи 39, с. 104. Отметим, что вычисление величины ускорения Кориолиса не потребовалось.

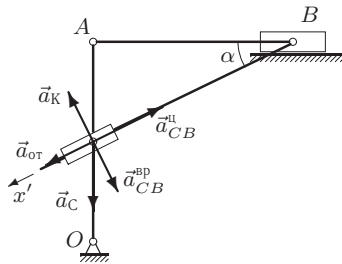
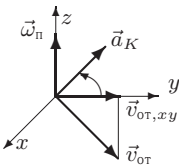


Рис. 122

¹Жуковский Николай Егорович (1847–1921) — русский ученый, основоположник гидроаэродинамики. Преподавал теоретическую механику в МГУ. Правило Жуковского определения направления ускорения Кориолиса основано на простой интерпретации правила векторного произведения. Согласно этому правилу, надо спроецировать вектор относительной скорости на плоскость, перпендикулярную вектору переносной угловой скорости. Затем получившуюся проекцию $\vec{v}_{\text{от},xy}$ следует повернуть в указанной плоскости по направлению переносного вращения. Это и будет искомое направление ускорения Кориолиса. В задаче 27 движение точки происходит в плоскости, поэтому проецировать вектор скорости не требуется. Для определения направления ускорения Кориолиса достаточно только повернуть его на 90° .



Maple–программа для решения этой задачи координатным методом дана на с. 369.

Задача 28. Механизм состоит из четырехзвенника $OABC$ и муфты, скользящей по кривошипу OA . Муфта соединена стержнем DB с шарниром B (рис. 123). Кривошип вращается с постоянной угловой скоростью $\omega_{OAz} = 3 \text{ с}^{-1}$. Даны размеры $OA = 40 \text{ см}$, $AB = 111 \text{ см}$, $BC = 43 \text{ см}$, $DB = 99,3 \text{ см}$. Найти скорости и ускорения шарниров A , B , а также скорость и ускорение муфты D относительно кривошипа OA в момент, когда $\alpha = 45^\circ$.

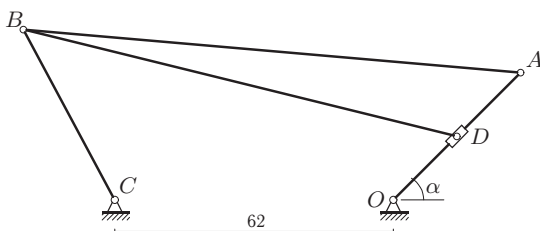


Рис. 123

Решение

1. План решения

Абсолютное движение муфты представим в виде суммы относительного движения по звену OA и переносного вместе с ним. Траекторией относительного движения является прямая, переносного движения — окружность с центром в точке O . По теореме сложения скоростей абсолютная скорость $\vec{v} = \vec{v}_{от} + \vec{v}_n$. Абсолютное ускорение вычисляется по теореме Кориолиса $\vec{a} = \vec{a}_{от} + \vec{a}_n + \vec{a}_k$. Неизвестные абсолютные скорость и ускорение выражаются через соответствующие величины полюса B

$$\vec{v} = \vec{v}_D = \vec{v}_B + \vec{v}_{DB},$$

$$\vec{a} = \vec{a}_D = \vec{a}_B^n + \vec{a}_B^t + \vec{a}_{DB}^{nl} + \vec{a}_{DB}^{bp}.$$

В результате для $\vec{v}_{от}$ и $\vec{a}_{от}$ имеем два основных векторных уравнения:

$$\vec{v}_B + \vec{v}_{DB} = \vec{v}_{от} + \vec{v}_n, \quad (2.29)$$

$$\vec{a}_B^n + \vec{a}_B^t + \vec{a}_{DB}^{nl} + \vec{a}_{DB}^{bp} = \vec{a}_{от} + \vec{a}_n + \vec{a}_k. \quad (2.30)$$

Решение этих уравнений можно выполнить *графически* и *аналитически*. Таким образом, имеем следующий план решения:

- 1) расчет скоростей четырехзвенника $OABC$;
- 2) определение $\vec{v}_{от}$ по формуле (2.29);

- 3) расчет ускорений четырехзвенника $OABC$;
 4) определение $\vec{a}_{от}$ по формуле (2.30).

2. Графическое решение

1) Расчет скоростей.

Существуют два способа графического расчета скоростей механизма: построение *плана скоростей* и расчет с помощью *мгновенных центров* скоростей. Используем первый способ. Сначала с помощью циркуля, линейки и транспортира на миллиметровой бумаге построим в масштабе сам механизм. Пронумеруем стержни.

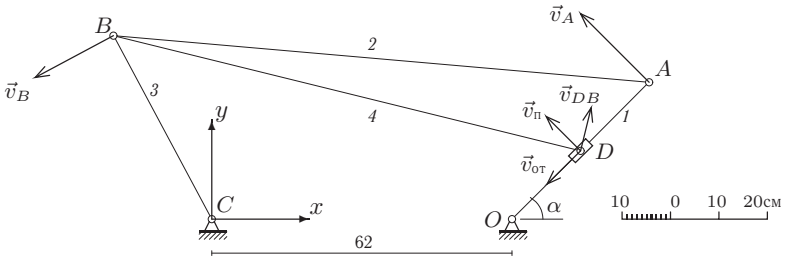


Рис. 124. Направление векторов скоростей и оси координат

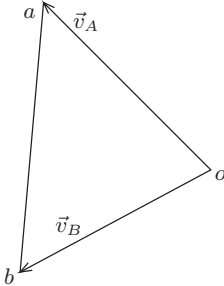


Рис. 125. План скоростей

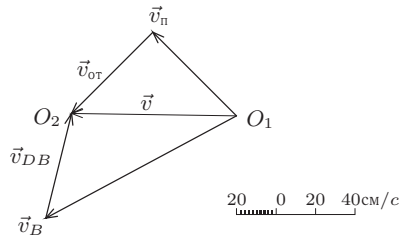


Рис. 126. Определение $\vec{v}_{от}$

План скоростей механизма начнем с построения вектора скорости точки A . Величина скорости равна

$$v_A = \omega_1 OA = 3 \cdot 40 = 120 \text{ см/с.}$$

Направление вектора — перпендикулярно радиусу OA (рис. 124) против часовой стрелки (т. к. $\omega_{1z} > 0$). Откладываем этот вектор от произвольной точки, обозначенной буквой o (рис. 125). Конец вектора отмечаем буквой a . Через точку o проводим прямую, параллельную направлению вектора скорости шарнира B , перпендикулярно радиусу CB траектории вокруг C . На этой прямой должна лежать точка b — конец вектора \vec{v}_B .

По основному свойству плана скоростей $\vec{ab} \perp \vec{AB}$. Через точку a перпендикулярно AB проводим вторую прямую. Пересечение проведенных прямых дает искомую точку b и, следовательно, длину вектора $v_B = = 109,7$ см/с (измеряем в масштабе) и угловую скорость звена AB

$$\omega_2 = ab/AB = 137,2/111 = 1,24 \text{ с}^{-1}.$$

Непосредственное измерение на чертеже дает $OD = 20$ см. Точка в этот момент оказалась на середине стержня. Найдем переносную скорость — скорость той точки звена OA , которая совпадает в данный момент с положением муфты: $v_n = v_D = 3 \cdot 20 = 60$ см/с. Вектор \vec{v}_n направлен перпендикулярно звену OA в сторону вращения.

2) Определение $\vec{v}_{от}$.

Равенство (2.29) представим графически. От некоторой точки O_1 отложим вектор \vec{v}_n , известный по величине и направлению, рис. 126. От его конца мы должны отложить вектор $\vec{v}_{от}$, известный лишь направлением (вдоль OA). Проведем через конец \vec{v}_n прямую, параллельную OA . Рассмотрим теперь левую часть равенства (2.29). От точки O_1 проведем вектор v_B , а через его конец прямую, перпендикулярную DB (на ней лежит неизвестный пока вектор \vec{v}_{DB}). Точка O_2 пересечения двух построенных прямых определяет вектор абсолютной скорости \vec{v} и величины векторов $v_{от} = 58,4$ см/с и $v_{DB} = 54,4$ см/с (измеряем в масштабе). Отсюда найдем необходимую в дальнейшем угловую скорость

$$\omega_4 = v_{DB}/DB = 54,4/99,3 = 0,55 \text{ с}^{-1}.$$

3) Расчет ускорений четырехзвенника $OABC$.

Ускорение точки A , лежащей на стержне OA , совершающем вращательное движение с постоянной угловой скоростью 3 с^{-1} , определяется аналитически: $a_A = \omega_1^2 OA = 9 \cdot 40 = 360 \text{ см/с}^2$.

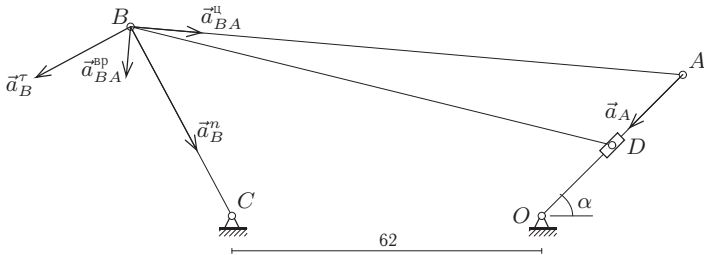


Рис. 127. Направления векторов ускорения

Ускорение точки B определим из векторного уравнения

$$\vec{a}_B = \vec{a}_B^n + \vec{a}_B^\tau = \vec{a}_A + \vec{a}_{BA}^u + \vec{a}_{BA}^{BP},$$

решение которого найдем графически. Для этого вычислим предварительно модули векторов $a_B^n = v_B^2/BC = 109,7^2/43 = 280 \text{ см/с}^2$,

$$a_{BA}^u = \omega_4^2 DB = 0,55^2 \cdot 99,3 = 29,8 \text{ см/с}^2.$$

От некоторой точки O_1 (рис. 128) строим план ускорений. Откладываем один за другим известные вектора \vec{a}_A и \vec{a}_{BA}^u . К концу последнего проводим перпендикуляр к AB — на нем должен лежать вектор \vec{a}_{BA}^{BP} . От той же точки O_1 откладываем \vec{a}_B^n и перпендикуляр к нему (\vec{a}_B^τ). Пересечение двух перпендикуляров дает точку O_2 — конец искомого вектора \vec{a}_B . Измерим длины

построенных векторов. Получим $a_B = 380 \text{ см/с}^2$ и касательное ускорение $a_B^\tau = 257 \text{ см/с}^2$. Ускорение точки D (переносное для муфты) определим по формуле $a_D = a_n = \omega_1^2 OD = 9 \cdot 20 = 180 \text{ см/с}^2$.

4) Определение $\vec{a}_{от}$.

Направления всех векторов, входящих в уравнение (2.30), изобразим на чертеже механизма.

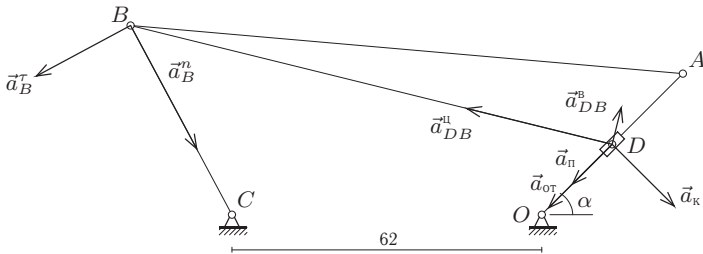


Рис. 129

Направление вектора ускорения Кориолиса получается поворотом вектора относительной скорости $\vec{v}_{от}$ по часовой стрелке ($\omega_n = \omega_{1z} > 0$). Модули некоторых векторов из (2.30) можно вычислить. Так как вектор переносной угловой скорости перпендикулярен плоскости чертежа, а следовательно, и относительной скорости, то $a_k = 2\omega_n v_{от} \sin 90^\circ = 2 \cdot 3 \cdot 58,4 = 350,2 \text{ см/с}^2$. Зная ω_4 , найдем $a_{DB}^u = \omega_4^2 DB = 0,55^2 \times 99,3 = 29,9 \text{ см/с}^2$. Величины $a_n = 180 \text{ см/с}^2$, $a_B^n = 280 \text{ см/с}^2$

и $a_B^T = 257 \text{ см/с}^2$ найдены ранее. По аналогии с предыдущими графическими построениями, от некоторой точки O_1 (рис. 130) отложим отдельно известные вектора из левой и правой части уравнения

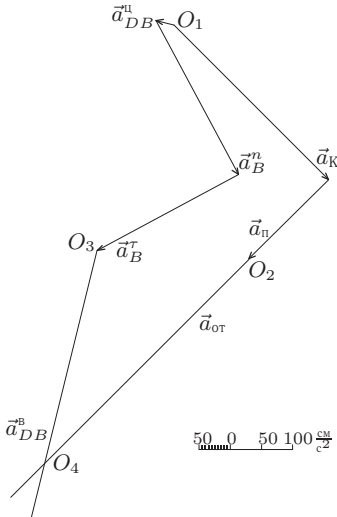


Рис. 130. Определение $a_{от}$

(2.30). Сначала последовательно построим векторы \vec{a}_K и \vec{a}_n . Получим точку O_2 , от которой надо провести прямую по направлению искомого ускорения $\vec{a}_{от}$. Направление задает положение кривошипа под углом $\alpha = 45^\circ$ к горизонту. Сумму векторов \vec{a}_{DB}^{nl} , \vec{a}_B^T и \vec{a}_B^{nl} в левой части (2.30) также построим от точки O_1 до точки O_3 . Точка O_4 пересечения направлений $\vec{a}_{от}$, направленного вдоль \vec{a}_n , и \vec{a}_{DB}^{np} , перпендикулярного DB , определяет абсолютное ускорение \vec{a} и искомое $\vec{a}_{от}$. Измеряя длину вектора $\vec{O_2O_4}$ на чертеже, приближенно получим $a_{от} = 464 \text{ см/с}^2$. Заметим, что графическое решение весьма чувствительно к точности построения.

Например, небольшая ошибка в направлении вектора \vec{a}_{DB}^{nl} или \vec{a}_K приводит к существенному изменению ответа.

3. Аналитическое решение

Для аналитического решения необходимо выбрать систему координат и определить координаты всех шарниров. Поместим начало координат в точку C (рис. 124). Таким образом, $x_C = 0$, $y_C = 0$, $x_O = 62 \text{ см}$, $y_O = 0$, $x_A = 62 + 40 \cos 45^\circ = 90,28 \text{ см}$, $y_A = 40 \sin 45^\circ = 28,28 \text{ см}$. Точку B найдем на пересечении окружности радиусом AB с центром в A и окружности радиусом CB с центром в начале координат:

$$\begin{aligned} x_B^2 + y_B^2 &= BC^2, \\ (x_A - x_B)^2 + (y_A - y_B)^2 &= AB^2. \end{aligned} \quad (2.31)$$

Получаем два решения, одно из них — $x_B = 4,96 \text{ см}$, $y_B = -42,71 \text{ см}$ — не соответствует рисунку. Выбираем второе: $x_B = -20,30 \text{ см}$, $y_B = 37,91 \text{ см}$. Этот же алгоритм использован в решении задачи 25. На с. 70 приведена соответствующая **Maple**-программа. Положение муфты D определяем из решения системы трех уравнений (задача

аналитической геометрии — пересечение окружности и прямой)

$$\begin{aligned}(x_D - x_B)^2 + (y_D - y_B)^2 &= DB^2, \\ x_D &= OC + S \cos \alpha, \quad y_D = S \sin \alpha.\end{aligned}\quad (2.32)$$

Система имеет два решения. Берем решение при $y_D > 0$. Таким образом, получаем $S = OD = 20$ см, $x_D = 76,14$ см, $y_D = 14,14$ см.

1) *Расчет скоростей.*

Уравнения *трех угловых скоростей* (2.9) примет вид

$$\begin{aligned}(y_A - y_O)\omega_{1z} + (y_B - y_A)\omega_{2z} + (y_C - y_B)\omega_{3z} &= 0, \\ (x_A - x_O)\omega_{1z} + (x_B - x_A)\omega_{2z} + (x_C - x_B)\omega_{3z} &= 0,\end{aligned}$$

или, при заданной угловой скорости $\omega_{1z} = 3$ с⁻¹,

$$\begin{aligned}9,63\omega_{2z} - 37,91\omega_{3z} &= -28,28 \cdot 3, \\ -110,58\omega_{2z} + 20,30\omega_{3z} &= -28,28 \cdot 3.\end{aligned}$$

Решаем систему и находим $\omega_{2z} = 1,236$ с⁻¹, $\omega_{3z} = 2,552$ с⁻¹.

2) *Определение $\vec{v}_{от}$.*

Предположим, что вектор относительной скорости $\vec{v}_{от}$ направлен от точки D к O . Уравнение (2.29) $\vec{v}_B + \vec{v}_{DB} = \vec{v}_{от} + \vec{v}_п$ запишем в координатной форме, в проекциях на оси x и y . Для этого воспользуемся представлением формулы Эйлера для скорости в виде

$$\begin{aligned}\vec{v}_B = \vec{\omega}_3 \times \vec{CB} &= \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ 0 & 0 & \omega_{3z} \\ x_B - x_C & y_B - y_C & 0 \end{vmatrix}, \\ \vec{v}_{DB} = \vec{\omega}_4 \times \vec{BD} &= \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ 0 & 0 & \omega_{4z} \\ x_D - x_B & y_D - y_B & 0 \end{vmatrix}, \\ \vec{v}_п = \vec{v}_D = \vec{\omega}_1 \times \vec{OD} &= \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ 0 & 0 & \omega_{1z} \\ x_D - x_O & y_D - y_O & 0 \end{vmatrix}.\end{aligned}$$

Отсюда следуют формулы для проекций

$$\begin{aligned}v_{Bx} &= -\omega_{3z}(y_B - y_C), & v_{By} &= \omega_{3z}(x_B - x_C), \\ v_{DBx} &= -\omega_{4z}(y_D - y_B), & v_{DBy} &= \omega_{4z}(x_D - x_B), \\ v_{пx} &= -\omega_{1z}(y_D - y_O), & v_{пy} &= \omega_{1z}(x_D - x_O).\end{aligned}\quad (2.33)$$

В результате получаем систему

$$-\omega_{3z}(y_B - y_C) - \omega_{4z}(y_D - y_B) = -v_{от} \cos 45^\circ - \omega_{1z}(y_D - y_O),$$

$$\omega_{3z}(x_B - x_C) + \omega_{4z}(x_D - x_B) = -v_{от} \sin 45^\circ + \omega_{1z}(x_D - x_O).$$

Подставив сюда известные координаты и найденные угловые скорости, запишем систему уравнений для $v_{от}$ и ω_{4z} :

$$\begin{aligned} 0,71v_{от} + 23,77\omega_{4z} &= 54,32, \\ 0,71v_{от} + 96,44\omega_{4z} &= 94,23. \end{aligned} \quad (2.34)$$

Находим решение системы $v_{от} = 58,36$ см/с, $\omega_{4z} = 0,549$ с⁻¹.

3) Расчет ускорений четырехзвенника OABC.

При $\varepsilon_{1z} = 0$ запишем уравнения трех угловых ускорений (2.10):

$$\begin{aligned} \varepsilon_{3z}(x_C - x_B) + \varepsilon_{2z}(x_B - x_A) &= \\ &= \omega_{3z}^2(y_C - y_B) + \omega_{2z}^2(y_B - y_A) + \omega_{1z}^2(y_A - y_O), \\ \varepsilon_{3z}(y_C - y_B) + \varepsilon_{2z}(y_B - y_A) &= \\ &= -\omega_{3z}^2(x_C - x_B) - \omega_{2z}^2(x_B - x_A) - \omega_{1z}^2(x_A - x_O). \end{aligned}$$

Координаты всех точек и угловые скорости звеньев уже найдены. При $\omega_{3z} = 2,552$ с⁻¹, $\omega_{2z} = 1,236$ с⁻¹, $\omega_{1z} = 3$ с⁻¹ получаем

$$20,30\varepsilon_{3z} - 110,58\varepsilon_{2z} = -2,552^2 \cdot 37,91 + 1,236^2 \cdot 9,63 + 3^2 \cdot 28,28 = 22,33,$$

$$9,63\varepsilon_{2z} - 37,91\varepsilon_{3z} = -2,552^2 \cdot 20,3 + 1,236^2 \cdot 110,58 - 3^2 \cdot 28,28 = -217,89.$$

Решение этой системы: $\varepsilon_{2z} = 0,895$ с⁻², $\varepsilon_{3z} = 5,975$ с⁻².

4) Определение $\vec{a}_{от}$.

Для того чтобы уравнение (2.30) записать в координатной форме, потребуются проекции векторов ускорений. Имеем

$$\begin{aligned} \vec{a}_B^n &= \vec{\omega}_3 \times \vec{v}_B = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ 0 & 0 & \omega_{3z} \\ v_{Bx} & v_{By} & 0 \end{vmatrix}, \\ \vec{a}_B^r &= \vec{\varepsilon}_3 \times \vec{CB} = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ 0 & 0 & \varepsilon_{3z} \\ x_B - x_C & y_B - y_C & 0 \end{vmatrix}, \\ \vec{a}_{DB}^{II} &= \vec{\omega}_4 \times \vec{v}_{DB} = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ 0 & 0 & \omega_{4z} \\ v_{DBx} & v_{DBy} & 0 \end{vmatrix}, \\ \vec{a}_{DB}^{вп} &= \vec{\varepsilon}_4 \times \vec{BD} = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ 0 & 0 & \varepsilon_{4z} \\ x_D - x_B & y_D - y_B & 0 \end{vmatrix}, \end{aligned}$$



$$\vec{a}_n = \vec{a}_D^n = \vec{\omega}_1 \times \vec{v}_D = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ 0 & 0 & \omega_{1z} \\ v_{Dx} & v_{Dy} & 0 \end{vmatrix},$$

$$\vec{a}_K = 2\vec{\omega}_n \times \vec{v}_{от} = 2 \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ 0 & 0 & \omega_{1z} \\ v_{отx} & v_{отy} & 0 \end{vmatrix}.$$

Раскрывая определители по элементам верхней строки (орты \vec{i} , \vec{j} координатных осей x и y), с учетом уже найденных проекций скоростей (2.33) получим выражения

$$\begin{aligned} a_{Bx}^n &= -\omega_{3z}^2(x_B - x_C), & a_{Bx}^r &= -\varepsilon_{3z}(y_B - y_C), \\ a_{DBx}^n &= -\omega_{4z}^2(x_D - x_B), & a_{DBx}^{BP} &= -\varepsilon_{4z}(y_D - y_O), \\ a_{nx} &= -\omega_1^2(x_D - x_O), & a_{Kx} &= -2\omega_{1z}v_{отy} = 2\omega_{1z}v_{от} \sin 45^\circ. \end{aligned}$$

Аналогично получают проекции на ось y . В итоге

$$\begin{aligned} & -\omega_{3z}^2(x_B - x_C) - \varepsilon_{3z}(y_B - y_C) - \omega_{4z}^2(x_D - x_B) - \varepsilon_{4z}(y_D - y_B) = \\ & = -a_{от} \cos 45^\circ - \omega_1^2(x_D - x_O) + 2 \cdot \omega_{1z}v_{от} \sin 45^\circ, \\ & -\omega_{3z}^2(y_B - y_C) + \varepsilon_{3z}(x_B - x_C) - \omega_{4z}^2(y_D - y_B) + \varepsilon_{4z}(x_D - x_B) = \\ & = -a_{от} \sin 45^\circ - \omega_1^2(y_D - y_O) - 2 \cdot \omega_{1z}v_{от} \cos 45^\circ. \end{aligned}$$

Подставив сюда все известные величины, получим систему уравнений для $a_{от}$ и ε_{4z} :

$$\begin{aligned} 0,71a_{от} + 23,77\varepsilon_{4z} &= 243,7, \\ 0,71a_{от} + 96,44\varepsilon_{4z} &= -13,87. \end{aligned} \quad (2.35)$$

Решение системы¹ дает искомое относительное ускорение $a_{от} = 463,77$ см/с² и угловое ускорение стержня $\varepsilon_{4z} = -3,544$ с⁻².

Maple-программа для решения этой задачи координатным методом дана на с. 370.

¹В процессе аналитического решения приходится 4 раза решать систему двух линейных уравнений. Эту математическую процедуру лучше выполнить в системе **Maple**, тем более что матрицы систем (2.34) и (2.35) совпадают.

ДИНАМИКА

3.1. Кинетическая энергия системы

В следующей задаче при вычислении кинетической энергии системы тел потребуются формула для момента инерции цилиндра радиусом R относительно его оси $J = mR^2/2$, выражение для момента инерции тела через его радиус инерции $J = mi^2$ и три основные формулы для кинетической энергии:

1. Вращательное движение: $T = J_z\omega^2/2$, где J_z — момент инерции тела относительно оси вращения, ω — угловая скорость.

2. Поступательное движение тела массой m : $T = mv^2/2$, где v — скорость какой-либо точки тела.

3. Плоское движение:

$$T = mv_C^2/2 + J_C\omega^2/2, \quad (3.1)$$

где v_C — скорость центра масс тела, J_C — момент инерции тела относительно оси, проходящей через центр масс, m — масса.

В частном случае кинетическая энергия однородного цилиндра массой m радиусом R , катящегося без проскальзывания по какой-либо неподвижной поверхности (рис. 131), имеет вид

$$T = 3mv_C^2/4, \quad (3.2)$$

что следует из (3.1) при $\omega = v_C/R$, $J_C = m_C R^2/2$. К этому случаю относится и движение цилиндра, падающего вертикально с разматыванием

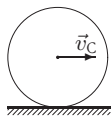


Рис. 131

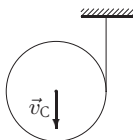


Рис. 132

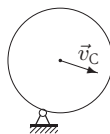


Рис. 133

нитью на его обод нити, один конец которой закреплен ("качение по нити", рис. 132) и вращение цилиндра вокруг оси на его ободу ("качение по точке", рис. 133).

Задача 29. Механическая система состоит из грузов A , E , блоков B , C и цилиндра D . Блок B вращается вокруг неподвижной оси, блок C и цилиндр D катятся по поверхности. Груз A движется вертикально (рис. 134). Нить, прикрепленная к оси цилиндра D , навита на меньший обод блока. Нить к грузу A вертикальная, остальные горизонтальные. Даны радиусы ободов и радиусы инерции блоков: $R_B = 5$ см, $r_B = 4$ см, $i_B = 3$ см, $R_C = 8$ см, $r_C = 2$ см, $i_C = 6$ см. Массы тел $m_A = 7$ кг, $m_B = 25$ кг, $m_C = 8$ кг, $m_D = 16$ кг, $m_E = 50$ кг. Найти приведенную массу μ системы в формуле $T = \mu v_A^2/2$, где v_A — скорость груза.

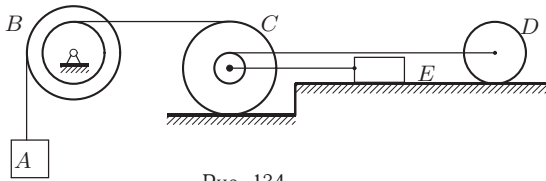


Рис. 134

Решение

Грузы A и E совершают поступательное движение, блок B — вращательное, блок C и цилиндр D — плоское. Предполагается, что груз A опускается и нити натянуты. Записываем выражения для соответствующих кинетических энергий:

$$\begin{aligned} T_A &= m_A v_A^2/2, & T_B &= J_B \omega_B^2/2, \\ T_C &= m_C v_C^2/2 + J_C \omega_C^2/2, \\ T_D &= (3/4)m_D v_D^2, & T_E &= m_E v_E^2/2, \end{aligned} \quad (3.3)$$

где v_D и v_C — скорости осей цилиндра и блока соответственно. Кинетическая энергия всей системы имеет вид

$$T = T_A + T_B + T_C + T_D + T_E. \quad (3.4)$$

Переходя от одного тела к другому, последовательно выражаем все кинематические величины, входящие в (3.3), через скорость груза A . Используем метод мгновенных центров скоростей¹. Выражаем угловую скорость блока B через v_A : $\omega_B = v_A/R_B$. Отсюда легко получить скорость точки K на внутреннем ободе блока B : $v_K = \omega_B r_B = v_A r_B/R_B$ (рис. 135). Нить нерастяжима, следовательно $v_N = v_K$.

¹Метод мгновенных центров скоростей удобно применять при определении модулей скоростей, как, например здесь, где в кинетическую энергию входят только квадраты скоростей. Там, где требуются знаки проекций, например в задачах на принцип возможных перемещений (задача 30) или при составлении уравнения Лагранжа 2-го рода (задачи 31–40), лучше использовать метод кинематических графов.

Мгновенный центр скоростей P_1 блока C находится в точке касания поверхности. Получаем угловую скорость блока: $\omega_C = v_N/(2R_C) = v_{ArB}/(2R_B R_C)$.

Определяем скорость центра масс блока

$$v_C = v_N/2 = \omega_C R_C = v_{ArB}/(2R_B)$$

и скорость точки L меньшего обода блока

$$v_L = \omega_C(R_C + r_C) = v_{ArB}(R_C + r_C)/(2R_B R_C).$$

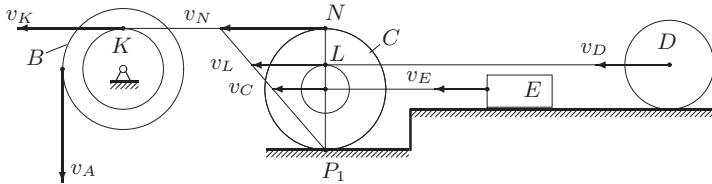


Рис. 135

Очевидно, опять же из-за нерастяжимости нити, $v_E = v_C$, $v_D = v_L$.

Таким образом, все кинематические величины, входящие в кинетическую энергию системы, выражены через v_A . Для моментов инерций имеем формулы

$$J_B = i_B^2 m_B, \quad J_C = i_C^2 m_C.$$

Подставляем скорости, угловые скорости и моменты инерции в (3.3). С учетом числовых данных получаем

$$T_A = 7 \frac{v_A^2}{2}, \quad T_B = \frac{m_B i_B^2 v_A^2}{2 R_B^2} = \frac{25 \cdot 9 v_A^2}{2 \cdot 25} = 9 \frac{v_A^2}{2},$$

$$T_C = \frac{m_C r_B^2 (i_C^2 + R_C^2) v_A^2}{8 R_B^2 R_C^2} = \frac{8 \cdot 16 (36 + 64) v_A^2}{8 \cdot 25 \cdot 64} = v_A^2,$$

$$T_D = \frac{3 m_D v_A^2 (R_C + r_C)^2 r_B^2}{16 R_B^2 R_C^2} = \frac{3 \cdot 16 v_A^2 (8 + 2)^2 \cdot 16}{16 \cdot 25 \cdot 64} = 3 v_A^2,$$

$$T_E = \frac{m_E v_A^2 r_B^2}{8 R_B^2} = \frac{50 \cdot v_A^2 \cdot 16}{8 \cdot 25} = 4 v_A^2.$$

Отсюда имеем приведенные массы тел (коэффициенты при $v_A^2/2$): $\mu_A = 7$ кг, $\mu_B = 9$ кг, $\mu_C = 2$ кг, $\mu_D = 6$ кг, $\mu_E = 8$ кг. Приведенная масса всей системы, согласно (3.4), равна $\mu = 7 + 9 + 2 + 6 + 8 = 32$ кг.

Заметим, что радиус цилиндра D по условию не задан и для решения не потребовался.

Maple–программа для решения этой задачи дана на с. 371.

3.2. Принцип возможных перемещений

Для систем с идеальными¹ стационарными² связями справедлив принцип возможных перемещений, согласно которому для равновесия необходимо и достаточно, чтобы при любом виртуальном перемещении сумма элементарных работ всех активных сил равнялась нулю:

$$\sum_i \vec{F}_i \cdot \delta \vec{r}_i = 0,$$

где \vec{r}_i — радиус-вектор точки приложения силы \vec{F}_i .

Задача 30. Плоский механизм с идеальными стационарными связями находится в равновесии под действием силы \vec{F} и моментов $M_1 = = 5$ Нсм, $M_2 = 10$ Нсм. Радиус диска $R = 4$ см, $CB = AB = = AD = 8$ см, $DK = 5$ см, $OA = 12\sqrt{2}$ см, $\alpha = 45^\circ$. Стержень BC горизонтальный, стержень AB в данный момент вертикальный. Деталь $ADKE$ состоит из жестко соединенных в D взаимно перпендикулярных стержней. Диск может кататься по горизонтальной поверхности без проскальзывания (рис. 136). Найти силу \vec{F} .

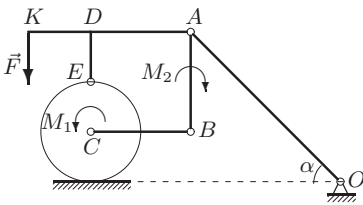


Рис. 136

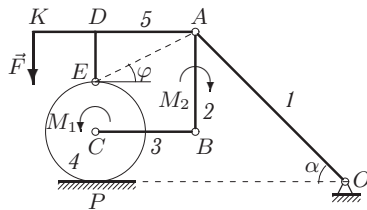


Рис. 137

Решение

Система состоит из пяти тел — трех стержней, уголка и диска. Пронумеруем тела (рис. 137). Уравнение принципа возможных скоростей для механизма имеет вид

$$-F v_{K,y} + M_{1z} \omega_{4z} + M_{2z} \omega_{2z} = 0. \quad (3.5)$$

Скалярное произведение здесь записываем в проекциях. Решим сначала задачу кинематики. Выразим угловые скорости ω_{4z} , ω_{2z} и скорость $v_{K,y}$ через ω_{1z} . Точка P касания диска и поверхности является мгновенным

¹Под идеальными связями понимают такие связи, у которых сумма элементарных работ реакций на любом возможном перемещении системы равна нулю.

²Стационарная связь — связь, не зависящая явно от времени.

центром скоростей диска, если качение происходит без проскальзывания. Скорость этой точки равна нулю. Составляем следующий кинематический граф: $P \xrightarrow{\frac{4}{\pi/2}} E \xrightarrow{\frac{5}{\varphi}} A \xrightarrow{\frac{1}{-\alpha}} O$. Отсюда следуют два уравнения¹:

$$\begin{aligned} v_{Ox} &= v_{Px} - \omega_{4z} 2R \sin(\pi/2) - \omega_{5z} EA \sin \varphi - \omega_{1z} OA \sin(-\alpha), \\ v_{Oy} &= v_{Py} + \omega_{4z} 2R \cos(\pi/2) + \omega_{5z} EA \cos \varphi + \omega_{1z} OA \cos(-\alpha). \end{aligned}$$

Перепишем эту систему при $EA \sin \varphi = DE = AB - R = 4$ см, $EA \cos \varphi = AD = 8$ см, $v_{Ox} = v_{Px} = 0$, $v_{Oy} = v_{Py} = 0$:

$$\begin{aligned} -8\omega_{4z} - 4\omega_{5z} + 12\omega_{1z} &= 0, \\ 8\omega_{5z} + 12\omega_{1z} &= 0. \end{aligned}$$

Решаем систему уравнений. Получаем зависимости: $\omega_{4z} = (9/4)\omega_{1z}$, $\omega_{5z} = -(3/2)\omega_{1z}$. Для того чтобы аналогичным образом выразить угловую скорость ω_{2z} стержня AB , входящую в (3.5), через ω_{1z} составляем граф $P \xrightarrow{\frac{4}{\pi/2}} C \xrightarrow{\frac{3}{0}} B \xrightarrow{\frac{2}{\pi/2}} A \xrightarrow{\frac{1}{-\alpha}} O$. Граф дает следующие уравнения:

$$\begin{aligned} v_{Ox} &= v_{Px} - \omega_{4z} R \sin(\pi/2) - \omega_{3z} CB \sin 0 - \omega_{2z} AB \sin(\pi/2) - \\ &\quad - \omega_{1z} OA \sin(-\alpha), \\ v_{Oy} &= v_{Py} + \omega_{4z} R \cos(\pi/2) + \omega_{3z} CB \cos 0 + \omega_{2z} AB \cos(\pi/2) + \\ &\quad + \omega_{1z} OA \cos(-\alpha). \end{aligned}$$

С учетом уже найденной угловой скорости $\omega_{4z} = (9/4)\omega_{1z}$ получим

$$\begin{aligned} -8\omega_{2z} + 3\omega_{1z} &= 0, \\ 8\omega_{3z} + 12\omega_{1z} &= 0. \end{aligned}$$

Решаем систему уравнений: $\omega_{2z} = (3/8)\omega_{1z}$, $\omega_{3z} = -(3/2)\omega_{1z}$.

Для выражения вертикальной скорости точки K через ω_{1z} составляем граф $O \xrightarrow{\frac{1}{\pi-\alpha}} A \xrightarrow{\frac{5}{\pi}} K$, из которого потребуется только одно уравнение: $v_{Ky} = v_{Oy} + \omega_{1z} OA \cos(3\pi/4) + \omega_{5z} AK \cos \pi$, или $v_{Ky} = -12\omega_{1z} - 13\omega_{5z} = 7,5\omega_{1z}$. Подставляем этот результат и угловые скорости $\omega_{4z} = (9/4)\omega_{1z}$ и $\omega_{2z} = (3/8)\omega_{1z}$ в (3.5). Сокращаем на $\omega_{1z} \neq 0$. Получаем уравнение $-7,5F + (9/4)M_{1z} + (3/8)M_{2z} = 0$. При $M_{1z} = 5$ Нсм, $M_{2z} = -10$ Нсм (момент M_2 направлен по часовой стрелке, отсюда и минус), вычисляем $F = 1$ Н.

¹ Можно составить немного более длинный граф, не вводя в рассмотрение дополнительный угол φ : $P \xrightarrow{\frac{4}{\pi/2}} E \xrightarrow{\frac{5}{\pi/2}} D \xrightarrow{\frac{5}{0}} A \xrightarrow{\frac{1}{-\alpha}} O$. Уравнения для проекций скоростей будут при этом те же.



3.3. Уравнение Лагранжа 2-го рода

3.3.1. Системы с одной степенью свободы

Ставится задача о составлении уравнения движения механической системы. Заданы обобщенная координата, массы отдельных тел системы, внешние силы и моменты. Решение задачи состоит из четырех частей.

1. Определение кинематических зависимостей. Выражение скоростей (угловых скоростей) через обобщенную скорость и обобщенную координату.

2. Нахождение кинетической энергии системы.

3. Определение обобщенной силы.

4. Составление уравнения Лагранжа¹ 2-го рода для системы с одной степенью свободы:

$$\frac{d}{dt} \frac{\partial T}{\partial \dot{q}} - \frac{\partial T}{\partial q} = Q, \quad (3.6)$$

где T — кинетическая энергия системы, q — обобщенная координата, Q — обобщенная сила.

Разберем отдельно каждый пункт плана.

1. Определение кинематических зависимостей

Для решения задачи кинематики рекомендуем аналитический метод — метод графов, с. 55. Метод МЦС связывает *модули* угловых скоростей, поэтому для решения задачи на составление уравнения Лагранжа 2-го рода он непригоден, так как при вычислении обобщенной силы необходимы *проекции* скоростей. В программах **Maple** удобно использовать процедуру **Граф**, с. 361. Аргументами процедуры являются названия вершин графа, расстояния между ними, углы, отсчитываемые от положительного направления оси x против часовой стрелки и угловые скорости. На выходе получаем два уравнения для компонент скоростей.

2. Нахождение кинетической энергии системы

Нахождение кинетической энергии системы разобрано на с. 86 и в решении задачи 29. Напомним также выражение для момента инерции

¹Лагранж Жозеф Луи (1736–1813) — французский математик, механик, астроном.

стержня длиной $AB = l$ и массой m относительно середины¹, $J_O = = ml^2/12$, и относительно конца, $J_A = ml^2/3$.

3. Определение обобщенной силы

Обобщенная сила находится с использованием скалярного произведения при вычислении мощности. Пусть q — обобщенная координата. Тогда для стационарных связей справедливо выражение

$$Q = \frac{1}{\dot{q}}(\vec{F}_1 \cdot \vec{v}_1 + \vec{F}_2 \cdot \vec{v}_2 + \dots + \vec{F}_k \cdot \vec{v}_k + \vec{M}_1 \cdot \vec{\omega}_1 + \vec{M}_2 \cdot \vec{\omega}_2 + \dots + \vec{M}_n \cdot \vec{\omega}_n),$$

где \vec{F}_i , $i = 1, \dots, k$ и \vec{M}_j , $j = 1, \dots, n$ — действующие на систему силы и моменты; \vec{v}_i — скорости точек приложения сил, $\vec{\omega}_j$ — угловые скорости тел, к которым приложены моменты. Скалярное произведение рекомендуем находить в координатной форме, например $\vec{F} \cdot \vec{v}_A = F_x v_{Ax} + + F_y v_{Ay} + F_z v_{Az}$. Вычисление через модули векторов и косинус угла α между ними в форме $\vec{F} \cdot \vec{v}_A = F v_A \cos \alpha$ обычно затруднительно, так как угол не задан.

4. Составление уравнения

При заданных выражениях для кинетической энергии и обобщенной силы составление уравнения движения (3.6) сводится к дифференцированию кинетической энергии по обобщенной координате и обобщенной скорости. Эта процедура, в основном, одинаковая для всех задач. Например, если кинетическая энергия имеет вид

$$T = \frac{\dot{q}^2}{2} (C_1 + C_2 \sin^2 q + C_3 \sin 2q),$$

то частные производные легко вычисляются:

$$\frac{\partial T}{\partial \dot{q}} = \dot{q}(C_1 + C_2 \sin^2 q + C_3 \sin 2q), \quad \frac{\partial T}{\partial q} = \frac{\dot{q}^2}{2}(C_2 \sin 2q + 2C_3 \cos 2q).$$

Полная производная по времени определяется по формуле

$$\frac{d}{dt} \frac{\partial T}{\partial \dot{q}} = \ddot{q}(C_1 + C_2 \sin^2 q + C_3 \sin 2q) + \dot{q}^2(C_2 \sin 2q + 2C_3 \cos 2q).$$

Окончательно уравнение примет вид

$$\ddot{q}(C_1 + C_2 \sin^2 q + C_3 \sin 2q) + (1/2)\dot{q}^2(C_2 \sin 2q + 2C_3 \cos 2q) = Q.$$

¹Когда говорится, что момент инерции вычисляется относительно середины, то подразумевается момент инерции относительно оси, перпендикулярной стержню, проходящей через его середину.



Приведем краткие решения некоторых задач. При этом, как правило, будем ограничиваться наиболее трудной частью решения — нахождением кинетической энергии и обобщенной силы. Дифференцирование же кинетической энергии по обобщенной координате и скорости обычно не вызывает проблем. Более того, в общем случае если кинетическая энергия имеет вид $T = (\dot{q}^2/2)f(q)$, то получается следующее уравнение

$$\ddot{q}f(q) + \frac{1}{2}\dot{q}^2 \frac{df}{dq} = Q. \quad (3.7)$$

Во всех задачах качение происходит без проскальзывания и трения качения, механизм расположен в вертикальной плоскости (кроме задачи 32). Если масса тела не указана, то она принимается равной нулю.

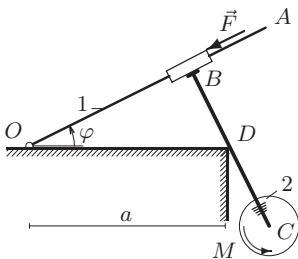


Рис. 138

Составить уравнение движения системы. За обобщенную координату принять угол поворота кривошипа φ .

Решение

Кривошип совершает вращательное движение и его кинетическая энергия равна $T_1 = J_1 \dot{\varphi}^2/2$, где $J_1 = m_1 b^2/3$. Движение диска 2

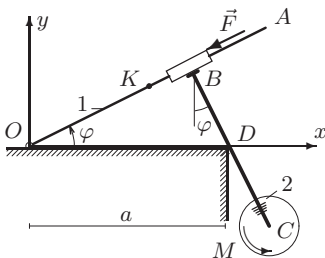


Рис. 139

Задача 31. К муфте B , скользящей по кривошипу OA , под прямым углом жестко прикреплен стержень BC (рис. 138). На конце стержня расположен диск радиусом R , образуя со стержнем одно тело. Стержень BC скользит без отрыва по гладкому углу, O — неподвижный шарнир. Масса кривошипа m_1 , диска — m_2 . К муфте вдоль кривошипа приложена сила \vec{F} , к диску — момент M . Дано: $OA = b$, $BC = l$.

Для определения его кинетической энергии необходимо знать выражение для скорости точки C . Прямой угол между OA и BC не меняется, поэтому угловая скорость диска равна угловой скорости кривошипа $\dot{\varphi}$. Кинетическая энергия диска: $T_2 = J_2 \dot{\varphi}^2/2 + m_2 v_C^2/2$. Скорость точки C найдем координатным способом. Выберем систему координат с началом в неподвижном шарнире O (рис. 139). Найдем расстояние

до муфты: $OB = a \cos \varphi$. Координаты точки B имеют вид

$$x_B = OB \cos \varphi = a \cos^2 \varphi, \quad y_B = OB \sin \varphi = a \sin \varphi \cos \varphi. \quad (3.8)$$

Учитывая, что $BC \perp OA$, получим

$$\begin{aligned} x_C &= x_B + BC \sin \varphi = a \cos^2 \varphi + l \sin \varphi, \\ y_C &= y_B - BC \cos \varphi = a \cos \varphi \sin \varphi - l \cos \varphi. \end{aligned}$$

Дифференцируем по времени:

$$\begin{aligned} v_{Cx} &= \dot{x}_C = \dot{\varphi}(-a \sin 2\varphi + l \cos \varphi), \\ v_{Cy} &= \dot{y}_C = \dot{\varphi}(a \cos 2\varphi + l \sin \varphi), \end{aligned}$$

и получаем $v_{Cx}^2 + v_{Cy}^2 = \dot{\varphi}^2(a^2 + l^2 - 2al \sin \varphi)$.

Запишем кинетическую энергию системы

$$T = T_1 + T_2 = \frac{m_1 b^2 \dot{\varphi}^2}{6} + \frac{m_2 R^2 \dot{\varphi}^2}{4} + \frac{m_2 v_C^2}{2} = \frac{\dot{\varphi}^2}{2} (C_1 + C_2 \sin \varphi),$$

где $C_1 = m_1 b^2/3 + m_2(a^2 + l^2 + R^2/2)$, $C_2 = -2alm_2$. Найдем выражение обобщенной силы:

$$Q = (1/\dot{\varphi})(M_z \dot{\varphi} + (-m_1 g)v_{Ky} + (-m_2 g)v_{Cy} + F_x v_{Bx} + F_y v_{By}).$$

Здесь $M_z = M$, $F_x = -F \cos \varphi$, $F_y = -F \sin \varphi$. Вертикальную компоненту скорости центра масс кривошипа v_{Ky} найдем из графа $O \frac{b/2}{\varphi} K$. Получим $v_{Ky} = \dot{\varphi}(b/2) \cos \varphi$. Скорости v_{Bx} , v_{By} вычислим, дифференцируя координаты (3.8):

$$v_{Bx} = \dot{x}_B = -\dot{\varphi} a \sin 2\varphi, \quad v_{By} = \dot{y}_B = \dot{\varphi} a \cos 2\varphi.$$

Таким образом, мощность силы F примет вид

$$F_x v_{Bx} + F_y v_{By} = F \dot{\varphi} a (\sin 2\varphi \cos \varphi - \cos 2\varphi \sin \varphi) = F a \dot{\varphi} \sin \varphi.$$

Это значение можно получить проще, с учетом, что скорость муфты v относительно стержня в проекции на ось $u = OA$ равна $v_u = dOB/dt = -a\dot{\varphi} \sin \varphi$. Так как $F_u = -F$, то мощность равна $v_u F_u = F a \dot{\varphi} \sin \varphi$.

Замечание. Для определения скорости точки C кинематический граф $O \frac{OB}{\varphi} B \xrightarrow{\varphi \rightarrow \pi/2} C$ использовать нельзя. В вершинах графа скорости контактирующих тел должны совпадать, а здесь в точке B происходит проскальзывание — разрыв скоростей.

Задача 32. Диск радиусом r и прямоугольный груз массой $m_1 = 1$ кг движутся между двумя параллельными плоскостями (рис. 140). Ползун массой $m_2 = 4$ кг соединен с грузом стержнем длиной $l = 1$ м. К стержню приложен момент $M = 4$ Нм. Масса диска $m_3 = 8$ кг. Механизм расположен в горизонтальной плоскости. Найти угловое ускорение стержня.

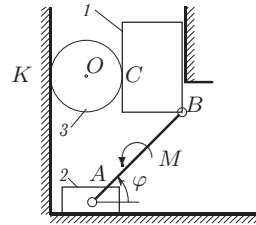


Рис. 140

Решение

Для вычисления кинетической энергии потребуются выражения скорости центра диска, его угловой скорости и скорости груза через обобщенную скорость $\dot{\varphi}$. Рассмотрим сначала кинематический граф $A \xrightarrow{l/\varphi} B$. Запишем соответствующие уравнения для проекций скоростей:

$$v_{Bx} = v_{Ax} - l\dot{\varphi} \sin \varphi,$$

$$v_{By} = v_{Ay} + l\dot{\varphi} \cos \varphi.$$

Учитывая, что $v_{Ay} = v_{Bx} = 0$, получаем скорость скольжения ползуна и скорость поступательного движения груза:

$$v_{Ax} = l\dot{\varphi} \sin \varphi, \quad v_{By} = l\dot{\varphi} \cos \varphi.$$

Составим граф $K \xrightarrow{2r/0} C$ для определения угловой скорости диска и получим

$$v_{Cy} = v_{Ky} + 2r\omega_z \cos \theta.$$

Здесь C и K — точки касания диска с грузом и стеной. Так как диск катится по плоскости без проскальзывания, то $v_{Kx} = v_{Ky} = 0$, $v_C = v_B$, поэтому угловая скорость диска $\omega_{3z} = (l/(2r))\dot{\varphi} \cos \varphi$. Далее, из графа

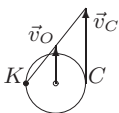


Рис. 141

от МЦС диска к его центру $K \xrightarrow{r/0} O$ следует $v_{Oy} = r\omega_{3z} \cos \theta$, или $v_{Oy} = (1/2)l\dot{\varphi} \cos \varphi$.

Заметим, что скорость центра O вдвое меньше скорости точки C . Это легче получить, используя понятие МЦС. У диска МЦС — точка K .

Вычислим кинетическую энергию груза, совершающего поступательное движение:

$$T_1 = \frac{m_1 v_{By}^2}{2} = \frac{m_1}{2} l^2 \dot{\varphi}^2 \cos^2 \varphi.$$

Кинетическая энергия ползуна $T_2 = m_2 v_A^2 / 2 = (1/2) m_2 l^2 \dot{\varphi}^2 \sin^2 \varphi$. Кинетическая энергия плоского движения однородного диска, катящегося без проскальзывания по неподвижной поверхности, согласно формуле (3.2), имеет вид $T_3 = (3/4) m_3 v_O^2 = (3/16) m_3 l^2 \dot{\varphi}^2 \cos^2 \varphi$. Суммарная кинетическая энергия имеет вид

$$T = T_1 + T_2 + T_3 = \frac{\dot{\varphi}^2}{2} l^2 ((m_1 + 3m_3/8) \cos^2 \varphi + m_2 \sin^2 \varphi).$$

Подставляем сюда массы $m_1 = 1$ кг, $m_2 = 4$ кг, $m_3 = 8$ кг и обнаруживаем, что кинетическая энергия системы в этом случае не зависит от обобщенной координаты: $T = 2l^2 \dot{\varphi}^2$. Это существенно упрощает решение.

Обобщенная сила вычисляется только по мощности момента M , так как механизм расположен в горизонтальной плоскости, и силы тяжести не имеют мощности: $Q = M$. Уравнение Лагранжа (3.6) при $\partial T / \partial \varphi = 0$ сразу дает ответ: $\ddot{\varphi} = M / (4l^2) = 1 \text{ с}^{-2}$.

Задача 33. Диск массой m_1 и радиусом r соединен с горизонтально скользящим ползуном стержнем AB длиной $3r$. Диск катится по цилиндрической поверхности радиусом $R = 2r$. К стержню приложен момент M (рис. 142). Масса ползуна m_2 . Составить уравнение движения системы. За обобщенную координату принять угол поворота стержня φ .

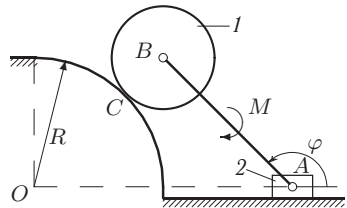


Рис. 142

Решение

Заметим, что по условию задачи треугольник OBA равнобедренный. Составляем кинематический граф $A \xrightarrow{3r} B \xrightarrow{-r} C$, где C — точка касания диска и цилиндрической поверхности. Запишем соответствующие уравнения для проекций скоростей:

$$\begin{aligned} v_{Cx} &= v_{Ax} - 3r\dot{\varphi} \sin \varphi - r\omega_z \sin(-\varphi), \\ v_{Cy} &= v_{Ay} + 3r\dot{\varphi} \cos \varphi + r\omega_z \cos(-\varphi). \end{aligned}$$

Учитывая, что $v_{Ay} = 0$ и $v_{Cx} = v_{Cy} = 0$, получаем угловую скорость диска и скорость ползуна: $\omega_z = -3\dot{\varphi}$, $v_{Ax} = 6r\dot{\varphi} \sin \varphi$.

Из графа $A \xrightarrow{3r} B$ следует:

$$\begin{aligned} v_{Bx} &= 6r\dot{\varphi} \sin \varphi - 3r\dot{\varphi} \sin \varphi = 3r\dot{\varphi} \sin \varphi, \\ v_{By} &= 3r\dot{\varphi} \cos \varphi \end{aligned}$$

и $v_B^2 = 9r^2\dot{\varphi}^2$. Используя формулу (3.2) для кинетической энергии плоского движения однородного диска, катящегося без проскальзывания по неподвижной поверхности, получим

$$T_1 = \frac{3m_1v_B^2}{4} = \frac{27}{4}m_1r^2\dot{\varphi}^2.$$

Вычислим кинетическую энергию ползуна A , совершающего поступательное движение:

$$T_2 = \frac{m_2v_{Ax}^2}{2} = 18m_2r^2\dot{\varphi}^2 \sin^2 \varphi.$$

Кинетическая энергия системы имеет вид

$$T = \frac{\dot{\varphi}^2}{2} (C_1 + C_2 \sin^2 \varphi),$$

где $C_1 = (27/2)m_1r^2$, $C_2 = 36m_2r^2$. Обобщенная сила вычисляется по моменту M и силе тяжести тела I :

$$Q = \frac{1}{\dot{\varphi}}((-M)\dot{\varphi} + (-m_1g)v_{By}) = -M - 3m_1gr \cos \varphi.$$

Задача 34. Механизм состоит из диска массой m_1 и радиусом r , шарнирно соединенных стержня AB массой m_2 и кривошипа CB длиной $3r$. Диск катится по цилиндрической поверхности радиусом $R = 2r$, $AB = OC$. К стержню AB применен момент M (рис. 143). Составить уравнение движения системы. За обобщенную координату принять угол φ .

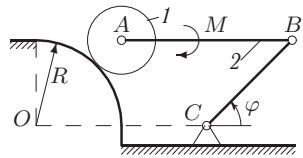


Рис. 143

Решение

По условию задачи $OA = BC = 3r$, $AB = OC$, поэтому $ABCO$ — параллелограмм, следовательно, стержень AB движется поступательно. При поступательном движении тела скорости всех его точек равны: $\vec{v}_A = \vec{v}_B$, а угловая скорость равна нулю: $\omega_{2z} = 0$. Составляем кинематический граф $C \xrightarrow[3r]{\varphi} B$, из которого сразу же следует

$$v_{Bx} = -3r\dot{\varphi} \sin \varphi,$$

$$v_{By} = 3r\dot{\varphi} \cos \varphi$$

и $v_A^2 = v_B^2 = 9r^2\dot{\varphi}^2$. Кинетическая энергия плоского движения однородного диска, катящегося без проскальзывания по неподвижной

поверхности, равна

$$T_1 = (3/4)m_1 v_A^2 = (27/4)m_1 r^2 \dot{\varphi}^2.$$

Кинетическая энергия поступательного движения стержня AB :

$$T_2 = m_2 v_A^2 / 2 = 9m_2 r^2 \dot{\varphi}^2 / 2.$$

Суммарная кинетическая энергия имеет вид

$$T = C_1 \dot{\varphi}^2 / 2,$$

где $C_1 = 9r^2(1,5m_1 + m_2)$ — константа (обобщенный момент инерции механизма). Обобщенная сила:

$$Q = \frac{1}{\dot{\varphi}} ((-M)\omega_{2z} + (-m_1 g)v_{Ay} + (-m_2 g)v_{Ay}) = -3(m_1 + m_2)gr \cos \varphi.$$

Для $C_1 = \text{const}$ уравнение Лагранжа имеет простой вид, $C_1 \ddot{\varphi} = Q$, и не отличается от уравнения прямолинейного движения точки массой C_1 под действием силы Q .

Задача 35. Уголок OAB из двух жестко скрепленных под прямым углом стержней вращается на шарнире O . Цилиндр радиусом R , шарнирно закрепленный на конце кривошипа CD , катится по уголку (рис. 144). $OA = R$, $AB = l$, $OD = DC = b$. Масса кривошипа m_1 , масса цилиндра m_2 . К уголку приложена горизонтальная сила \vec{F} , к цилиндру — момент M . Составить уравнение движения системы. За обобщенную координату принять угол φ поворота кривошипа.

Решение

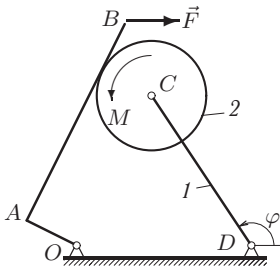


Рис. 144

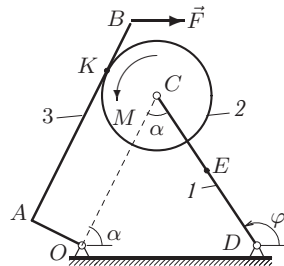


Рис. 145

По условию задачи треугольник OCD равнобедренный, поэтому $\alpha = \varphi/2$. Для определения угловых скоростей цилиндра и уголка составим следующий граф:

$$O \xrightarrow[\alpha + \pi/2]{R} A \xrightarrow[\alpha]{AK} K \xrightarrow[\alpha - \pi/2]{R} C \xrightarrow[\varphi + \pi]{b} D,$$

где $AK = OC = 2b \cos \alpha = 2b \cos(\varphi/2)$. Получим два уравнения для скоростей:

$$\begin{aligned} v_{Dx} &= v_{Ox} - \omega_{3z} R \sin(\varphi/2 + \pi/2) - \omega_{3z} AK \sin(\varphi/2) - \\ &\quad - \omega_{2z} R \sin(\varphi/2 - \pi/2) - \dot{\varphi} b \sin(\varphi + \pi), \\ v_{Dy} &= v_{Oy} + \omega_{3z} R \cos(\varphi/2 + \pi/2) + \omega_{3z} AK \cos(\varphi/2) + \\ &\quad + \omega_{2z} R \cos(\varphi/2 - \pi/2) + \dot{\varphi} b \cos(\varphi + \pi). \end{aligned}$$

С учетом $v_{Dx} = v_{Ox} = v_{Dy} = v_{Oy} = 0$ преобразуем эти уравнения:

$$\begin{aligned} -R\omega_{3z} \cos(\varphi/2) - b\omega_{3z} \sin(\varphi) + R\omega_{2z} \cos(\varphi/2) + b\dot{\varphi} \sin(\varphi) &= 0, \\ -R\omega_{3z} \sin(\varphi/2) + 2b\omega_{3z} \cos^2(\varphi/2) + R\omega_{2z} \sin(\varphi/2) - b\dot{\varphi} \cos(\varphi) &= 0. \end{aligned}$$

Решаем систему двух уравнений и получаем значения угловых скоростей

$$\omega_{2z} = \dot{\varphi}(1/2 - (b/R) \sin(\varphi/2)), \quad \omega_{3z} = \dot{\varphi}/2. \quad (3.9)$$

Последнее значение можно найти значительно проще, заметив, что угол отрезка OA с осью x равен $\varphi/2 + \pi/2$. Дифференцируя это выражение, сразу получаем $\omega_{3z} = \dot{\varphi}/2$.

Кинетическая энергия вращательного движения кулисы: $T_1 = J_1 \dot{\varphi}^2/2 = m_1 b^2 \dot{\varphi}^2/6$. Кинетическая энергия плоского движения цилиндра равна $T_2 = m_2 v_C^2/2 + J_2 \omega_{2z}^2/2$, где $J_2 = m_2 R^2/2$ — момент инерции цилиндра. Скорость центра масс цилиндра найдем из графа $D \xrightarrow{\frac{b}{\varphi}} C$. Получим

$$\begin{aligned} v_{Cx} &= v_{Dx} - b\dot{\varphi} \sin \varphi = -b\dot{\varphi} \sin \varphi, \\ v_{Cy} &= v_{Dy} + b\dot{\varphi} \cos \varphi = b\dot{\varphi} \cos \varphi, \end{aligned}$$

откуда $v_C^2 = b^2 \dot{\varphi}^2$. Кинетическая энергия системы приобретает вид

$$T = \frac{\dot{\varphi}^2}{2} (C_1 + C_2 \sin(\varphi/2) + C_3 \sin^2(\varphi/2)),$$

где $C_1 = m_1 b^2/3 + m_2 (R^2/8 + b^2)$, $C_2 = -m_2 b R/2$, $C_3 = m_2 b^2/2$.

Найдем обобщенную силу:

$$Q = (1/\dot{\varphi})(M\omega_{2z} + Fv_{Bx} - m_1 g v_{Ey} - m_2 g v_{Cy}).$$

Здесь угловая скорость ω_{2z} определена выражением (3.9). Вертикальную компоненту скорости центра масс кулисы определяем по уже найденной скорости точки C : $v_{Ey} = v_{Cy}/2$, а скорость точки B —

из графа $O \xrightarrow[\varphi/2+\pi/2]{R} A \xrightarrow[\varphi/2]{l} B$, из которого имеем

$$\begin{aligned} v_{Bx} &= v_{Ox} - \omega_{3z} R \sin(\varphi/2 + \pi/2) - \omega_{3z} l \sin(\varphi/2) = \\ &= -(\dot{\varphi}/2)(R \cos(\varphi/2) + l \sin(\varphi/2)). \end{aligned}$$

В итоге

$$Q = M \left(\frac{1}{2} - \frac{b}{R} \sin \frac{\varphi}{2} \right) - \frac{F}{2} \left(R \cos \frac{\varphi}{2} + l \sin \frac{\varphi}{2} \right) - \left(\frac{m_1}{2} + m_2 \right) g b \cos \varphi.$$

Задача 36. Невесомый уголок, составленный из двух жестко соединенных взаимно перпендикулярных стержней, опирается без проскальзывания на диск массой m_1 , радиусом R , с неподвижной осью и

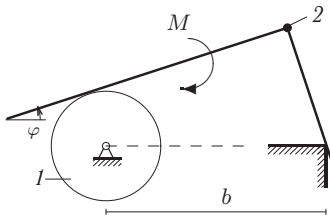


Рис. 146

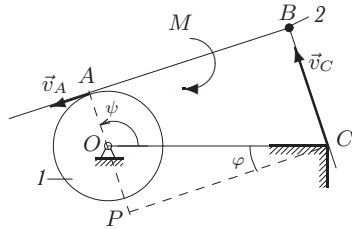


Рис. 147

гладкий угол (рис. 146). На уголке, к которому приложен момент M , находится точка массой m_2 . Составить уравнение движения системы. За обобщенную координату принять угол поворота φ .

Решение

Для определения угловой скорости цилиндра составим граф:

$$O \xrightarrow[\psi]{R} A \xrightarrow[\varphi]{AB} B \xrightarrow[\varphi - \pi/2]{BC} C,$$

где $\psi = \varphi + \pi/2$, $AB = PC = b \cos \varphi$, $BC = AP = R + b \sin \varphi$, P — мгновенный центр скоростей уголка, который располагается на пересечении перпендикуляров к векторам скоростей точек A и C (рис. 147). Получим два уравнения для скоростей:

$$\begin{aligned} v_{Cx} &= v_{Ox} - \omega_{1z} R \sin(\varphi + \pi/2) - \dot{\varphi} AB \sin \varphi - \dot{\varphi} BC \sin(\varphi - \pi/2), \\ v_{Cy} &= v_{Oy} + \omega_{1z} R \cos(\varphi + \pi/2) + \dot{\varphi} AB \cos \varphi + \dot{\varphi} BC \cos(\varphi - \pi/2). \end{aligned}$$

С учетом $v_{Ox} = v_{Oy} = 0$, $v_{Cx} = -v_C \sin \varphi$, $v_{Cy} = v_C \cos \varphi$ преобразуем эти уравнения:

$$\begin{aligned} -v_C \sin \varphi &= -\omega_{1z} R \cos \varphi - \dot{\varphi} b \sin \varphi \cos \varphi + \dot{\varphi} (R + b \sin \varphi) \cos \varphi, \\ v_C \cos \varphi &= -\omega_{1z} R \sin \varphi + \dot{\varphi} b \cos^2 \varphi + \dot{\varphi} (R + b \sin \varphi) \sin \varphi. \end{aligned}$$

Получилась система двух уравнений относительно скорости v_C и угловой скорости ω_{1z} . Умножаем первое уравнение на $\cos \varphi$, а второе — на $\sin \varphi$ и складываем их. Получаем значение угловой скорости

$$\omega_{1z} = \dot{\varphi}(R + b \sin \varphi)/R. \quad (3.10)$$

Скорость точки B массой m_2 найдем из графа $O \xrightarrow{\frac{R}{\psi}} A \xrightarrow{\frac{AB}{\varphi}} B$.

Этому графу соответствуют уравнения

$$\begin{aligned} v_{Bx} &= v_{Ox} - \omega_{1z}R \sin(\varphi + \pi/2) - \dot{\varphi}AB \sin \varphi = -\dot{\varphi}(b \sin 2\varphi + R \cos \varphi), \\ v_{By} &= v_{Oy} + \omega_{1z}R \cos(\varphi + \pi/2) + \dot{\varphi}AB \cos \varphi = \dot{\varphi}(b \cos 2\varphi - R \sin \varphi), \end{aligned}$$

откуда $v_B^2 = \dot{\varphi}^2(b^2 + R^2 + 2Rb \sin \varphi)$. Заметим, что скорости точек этого механизма можно также найти с помощью МЦС уголка (рис. 147). Из соотношения $v_A = R\omega_1 = AP|\dot{\varphi}|$ с учетом того, что направления поворотов цилиндра и уголка совпадают и $AP = R + b \sin \varphi$, получаем (3.10). Кроме того, $v_B^2 = \dot{\varphi}^2 PB^2 = \dot{\varphi}^2(AB^2 + AP^2) = \dot{\varphi}^2(b^2 + R^2 + 2Rb \sin \varphi)$.

Кинетическая энергия вращательного движения цилиндра: $T_1 = J_1 \omega_{1z}^2/2 = m_1 \dot{\varphi}^2(R + b \sin \varphi)^2/4$. Кинетическая энергия точки B равна $T_2 = m_2 v_B^2/2 = m_2 \dot{\varphi}^2(b^2 + R^2 + 2Rb \sin \varphi)/2$. Кинетическая энергия системы приобретает вид

$$T = \frac{\dot{\varphi}^2}{2} (C_1 + C_2 \sin \varphi + C_3 \sin^2 \varphi),$$

где $C_1 = m_1 R^2/2 + m_2(R^2 + b^2)$, $C_2 = bR(m_1 + 2m_2)$, $C_3 = m_1 b^2/2$.

Найдем обобщенную силу:

$$Q = (1/\dot{\varphi})(-M\dot{\varphi} - m_2 g v_{By}) = -M - m_2 g(b \cos 2\varphi - R \sin \varphi).$$

Задача 37. Оси цилиндров соединены невесомым спарником.

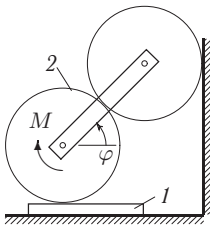


Рис. 148

Верхний цилиндр катится без проскальзывания по вертикальной плоскости. Нижний цилиндр, к которому приложен момент M , находится в зацеплении с верхним и катится по пластинке, скользящей по горизонтальной плоскости. Радиусы цилиндров r . Масса пластинки m_1 , нижнего цилиндра — m_2 (рис. 148). Составить уравнение движения системы. За обобщенную координату принять угол поворота спарника φ .

Решение

Рассмотрим кинематический граф (рис. 149) $A \xrightarrow{\frac{2r}{\varphi}} B$ вдоль спарника AB . Соответствующие уравнения для проекций скоростей имеют вид $v_{Bx} = v_{Ax} - 2r\dot{\varphi} \sin \varphi$, $v_{By} = v_{Ay} + 2r\dot{\varphi} \cos \varphi$.

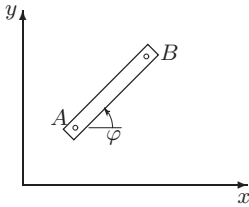


Рис. 149

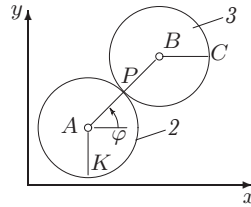


Рис. 150

Учитывая, что $v_{Ay} = v_{Bx} = 0$, имеем

$$v_{Ax} = 2r\dot{\varphi} \sin \varphi, \quad v_{By} = 2r\dot{\varphi} \cos \varphi.$$

Рассмотрим проекции скоростей на ось y для графа $B \xrightarrow{r} C$ (рис. 150): $v_{Cy} = v_{By} + r\omega_{3z} \cos 0$. Откуда получаем угловую скорость верхнего цилиндра

$$\omega_{3z} = -2\dot{\varphi} \cos \varphi. \quad (3.11)$$

Проекция скоростей на ось x для графа $K \xrightarrow{\frac{r}{\pi/2}} A \xrightarrow{\frac{2r}{\varphi}} B$, где K — точка касания нижнего цилиндра и пластинки (рис. 150):

$$v_{Bx} = v_{Kx} - r\omega_{2z} \sin(\pi/2) - 2r\dot{\varphi} \sin \varphi.$$

Найдем скорость поступательного движения пластинки:

$$v_{Kx} = r\omega_{2z} + 2r\dot{\varphi} \sin \varphi. \quad (3.12)$$

Запишем граф по цилиндрам: $K \xrightarrow{\frac{r}{\pi/2}} A \xrightarrow{\frac{r}{\varphi}} P \xrightarrow{\frac{r}{\varphi}} B$, где P — точка соприкосновения цилиндров. Запишем соответствующее уравнение для проекции скорости: $v_{Bx} = v_{Kx} - r\omega_{2z} \sin(\pi/2) - r\omega_{2z} \sin \varphi - r\omega_{3z} \sin \varphi$, или

$$v_{Kx} = r\omega_{2z} + r\omega_{2z} \sin \varphi + r\omega_{3z} \sin \varphi. \quad (3.13)$$

Из (3.11)–(3.13) следует выражение для угловой скорости нижнего цилиндра $\omega_{2z} = 2\dot{\varphi}(1 + \cos \varphi)$ и скорость пластинки $v_{Kx} = 2r\dot{\varphi}(1 + \cos \varphi + \sin \varphi)$.

Запишем кинетическую энергию поступательного движения пластинки и плоского движения цилиндра:

$$T_1 = \frac{m_1 v_{Kx}^2}{2} = 2m_1 r^2 \dot{\varphi}^2 (1 + \cos \varphi + \sin \varphi)^2,$$

$$T_2 = \frac{m_2 v_{Ax}^2}{2} + \frac{J_2 \omega_{2z}^2}{2},$$

где $v_{Ax} = 2r\dot{\varphi} \sin \varphi$, $J_2 = m_2 r^2/2$. Суммарная кинетическая энергия имеет вид

$$T = \frac{\dot{\varphi}^2}{2} (C_1 + C_2 \sin^2 \varphi + C_3 \sin 2\varphi + C_4 \sin \varphi + C_5 \cos \varphi),$$

где C_k , $k = 1, \dots, 5$ — константы, зависящие от размеров и масс. В выражение обобщенной силы войдет только момент:

$$Q = \frac{-M\omega_{2z}}{\dot{\varphi}} = -2M(1 + \cos \varphi).$$

Maple-программа для решения этой задачи дана на с. 374.

Задача 38. На оси обода радиусом R , массой m_1 шарнирно закреплен стержень AB длиной l , скользящий одним концом

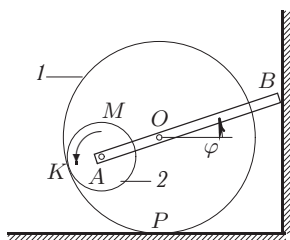


Рис. 151

по вертикальной плоскости. На другом конце стержня шарнирно закреплен диск радиусом r , катящийся по внутренней поверхности обода. К диску приложен момент M . Качение обода по горизонтальной плоскости происходит без проскальзывания (рис. 151). Масса диска m_2 . Составить уравнение движения системы. За обобщенную координату принять угол поворота стержня φ .

Решение

Найдем кинематические величины, входящие в выражение для кинетической энергии и обобщенной силы. Введем обозначение $a = OB = l - R + r$. Составим граф $O \xrightarrow{\frac{a}{\varphi}} B$, из которого следует, что

$$v_{Bx} = v_{Ox} - a\dot{\varphi} \sin \varphi.$$

Получим скорость центра, необходимую для вычисления кинетической энергии обода: $v_{Ox} = a\dot{\varphi} \sin \varphi$. Точку касания обода земли (МЦС обода) обозначим P . В соответствии с графом $P \xrightarrow{\frac{R}{\pi/2}} O$ запишем уравнение

$$v_{Ox} = v_{Px} - R\omega_{1z} \sin(\pi/2),$$

из которого при $v_{Px} = 0$ сразу следует выражение для угловой скорости обода: $\omega_{1z} = -(a/R)\dot{\varphi} \sin \varphi$. Найдем скорость центра диска. Составим граф $A \xrightarrow{\frac{R-r}{\varphi}} O$ и запишем два уравнения:

$$v_{Ox} = v_{Ax} - (R - r)\dot{\varphi} \sin \varphi; \quad v_{Oy} = v_{Ay} + (R - r)\dot{\varphi} \cos \varphi.$$

Получим

$$\begin{aligned} v_{Ax} &= l\dot{\varphi} \sin \varphi, \\ v_{Ay} &= -(R-r)\dot{\varphi} \cos \varphi. \end{aligned} \quad (3.14)$$

Пусть K — точка касания обода и диска. Скорости этих тел здесь совпадают. Для того чтобы определить угловую скорость диска 2, необходимо составить граф $K \xrightarrow[R]{\varphi} O$ и записать соответствующее уравнение в проекции на ось x : $v_{Ox} = v_{Kx} - R\omega_{1z} \sin \varphi$. Компонента скорости точки касания обода и диска равна $v_{Kx} = a\dot{\varphi} \sin \varphi(1 - \sin \varphi)$. Из графа $K \xrightarrow[r]{\varphi} A$ следует уравнение $v_{Ax} = v_{Kx} - r\omega_{2z} \sin \varphi$, из которого можно найти угловую скорость:

$$\omega_{2z} = \frac{v_{Kx} - v_{Ax}}{r \sin \varphi} = \frac{\dot{\varphi}(a(1 - \sin \varphi) - l)}{r}. \quad (3.15)$$

Кинетическая энергия плоского движения обода равна

$$T_1 = \frac{m_1 v_O^2}{2} + \frac{J_1 \omega_{1z}^2}{2},$$

где $J_1 = m_1 R^2$ — момент инерции обода. С учетом выражения для скорости центра получим $T_1 = m_1 a^2 \dot{\varphi}^2 \sin^2 \varphi$. Найдем кинетическую энергию плоского движения диска 2:

$$T_2 = \frac{m_2 v_A^2}{2} + \frac{J_2 \omega_{2z}^2}{2},$$

где $J_2 = m_2 r^2/2$ — момент инерции цилиндра. Подставим сюда выражения для скорости центра (3.14) и угловой скорости (3.15). Получим

$$T_2 = \frac{\dot{\varphi}^2}{4} m_2 (2l^2 \sin^2 \varphi + 2(R-r)^2 \cos^2 \varphi + (a(1 - \sin \varphi) - l)^2).$$

Кинетическая энергия системы имеет вид

$$T = T_1 + T_2 = \frac{\dot{\varphi}^2}{2} (C_1 + C_2 \sin^2 \varphi + C_3 \sin \varphi),$$

где C_1, C_2, C_3 — константы, зависящие от масс и размеров.

Обобщенная сила зависит от момента M и силы тяжести диска 2:

$$Q = \frac{1}{\dot{\varphi}} (M\omega_{2z} - m_2 g v_{Ay}) = M(a(1 - \sin \varphi) - l)/r + m_2 g (R-r) \cos \varphi.$$

Задача 39. На стержень AC двухзвенника OAC надета невесомая муфта B , шарнирно закрепленная на кривошипе OB длиной a . К кривошипу приложен момент M , к ползуну C , скользящему по горизонтальной поверхности, — сила F ; $OA = AC = a$ (рис. 152).

Масса стержня OA равна m_1 , масса ползуна C — m_2 . Составить уравнение движения системы. За обобщенную координату принять угол поворота кривошипа φ .

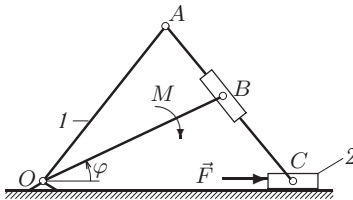


Рис. 152

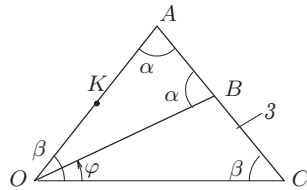


Рис. 153

Решение

Вычисление скоростей точек механизмов, содержащих муфты, в общем случае связано с теоремой сложения скоростей и представляет собой достаточно трудную задачу (например, задача 27, с. 74). Однако в данной задаче есть счастливое обстоятельство — наличие равнобедренного треугольника OBA (рис. 153). Это позволяет существенно упростить решение. Имеем $\angle OAB = \angle OBA = \alpha$. Обозначим $\angle COA = \beta$. Очевидны два уравнения:

$$2\alpha + \beta - \varphi = \pi, \quad \alpha + 2\beta = \pi.$$

Из этой системы получаем $\beta = (\pi - \varphi)/3$. Дифференцируя найденное соотношение углов, получаем угловую скорость стержня OA : $\omega_{1z} = -\dot{\varphi}/3$.

Для графа $O \xrightarrow{1/\beta} A \xrightarrow{-1/\beta} C$ запишем соответствующие уравнения для проекций скоростей:

$$v_{Cx} = v_{Ox} - a\omega_{1z} \sin \beta - a\omega_{3z} \sin(-\beta),$$

$$v_{Cy} = v_{Oy} + a\omega_{1z} \cos \beta + a\omega_{3z} \cos(-\beta).$$

Учитывая, что $v_{Ox} = v_{Oy} = 0$ и $v_{Cy} = 0$, из второго уравнения сначала получим угловую скорость $\omega_{3z} = -\omega_{1z}$, затем из первого уравнения имеем $v_{Cx} = -2a\omega_{1z} \sin \beta = (2/3)a\dot{\varphi} \sin \beta$.

Вычислим¹ кинетическую энергию вращательного движения стержня OA . Момент инерции J_1 стержня относительно оси вращения равен

¹В действительности в таких задачах кинетическая энергия и обобщенная сила не «вычисляются», т. е. им не ставится в соответствие некоторое число. В данном случае для решения задачи необходимо получить аналитические выражения всех величин через обобщенные координаты и скорости. Выражение «вычислим...» — дань традиции прикладных задач. В теоретической механике задачи, в основном, теоретические.

$m_1 a^2 / 3$. Отсюда

$$T_1 = J_1 \omega_{1z}^2 / 2 = m_1 a^2 \omega_{1z}^2 / 6 = m_1 a^2 \dot{\varphi}^2 / 54.$$

Кинетическая энергия ползуна:

$$T_2 = m_2 v_C^2 / 2 = (2/9) m_2 a^2 \dot{\varphi}^2 \sin^2 \beta.$$

Получаем вид кинетической энергии системы:

$$T = T_1 + T_2 = \frac{\dot{\varphi}^2}{18} \left(\frac{m_1}{3} + 4m_2 \sin^2 \left(\frac{\pi - \varphi}{3} \right) \right).$$

Для определения обобщенной силы потребуется значение вертикальной составляющей скорости точки приложения силы тяжести стержня AO .

Обозначим середину стержня K и составим к ней граф $O \xrightarrow{1/\beta} K$.

Найдем

$$v_{Ky} = v_{Oy} + (a/2) \omega_{1z} \cos \beta = (-a/6) \dot{\varphi} \cos \beta.$$

Получим выражение обобщенной силы:

$$Q = \frac{1}{\dot{\varphi}} (F v_{Cx} - m_1 g v_{Ky} - M \dot{\varphi}) = (2/3) F a \sin \beta + (a/6) m_1 g \cos \beta - M,$$

где $\beta = (\pi - \varphi)/3$.

Задача 40. На конце стержня AC , вращающегося на неподвижной оси B , шарнирно закреплена муфта A с массой m_1 и моментом инерции J_1 . На другом конце стержня AC закреплен ползун C , скользящий по нижней поверхности вертикального поршня. Муфта скользит по стержню KD , качающемуся на оси D . Масса поршня равна m_2 . К стержню KD приложен момент M (рис. 154). Дано: $AB = BD = a$, $BC = b$. Составить уравнение движения системы. За обобщенную координату принять угол поворота φ стержня AC .

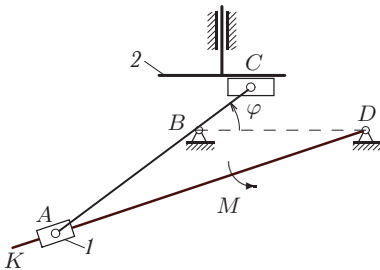


Рис. 154

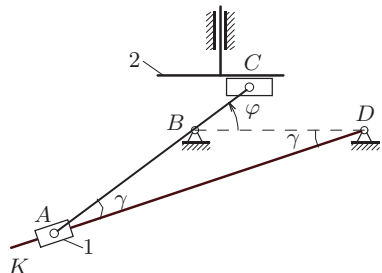


Рис. 155

Решение

Особенность решения этой задачи — способ определения угловой скорости ω_{1z} стержня KD , необходимой для определения вращательной части кинетической энергии муфты. Обычно все кинематические характеристики, включая угловые скорости, определяются из кинематических графов. В задачах, содержащих скользящие или вращающиеся муфты, используют формулу сложения скоростей, разделяя абсолютное движение на относительное и переносное. Здесь же можно поступить проще. Заметим, что по условию треугольник ABD равнобедренный. Углы при основании треугольника обозначим γ . Очевидно, что $\gamma = \varphi/2$ (рис. 155). Стержень KD образует с осью x угол $\gamma + \pi$. Дифференцируя, получаем $\omega_{1z} = \dot{\varphi}/2$. Найдем скорость ползуна и скорость поршня (вертикальную компоненту скорости ползуна). Составим граф $B \xrightarrow{b} C$. Получим необходимую компоненту скорости:

$$v_{Cy} = v_{By} + \dot{\varphi}b \cos \varphi.$$

Точка B — неподвижный шарнир, поэтому при $v_{By} = 0$ определяем скорость поршня $v_{Cy} = \dot{\varphi}b \cos \varphi$. Аналогично из графа $A \xrightarrow{a} B$ имеем

$$v_{Bx} = v_{Ax} - \dot{\varphi}a \sin \varphi,$$

$$v_{By} = v_{Ay} + \dot{\varphi}a \cos \varphi,$$

откуда находим компоненты скорости центра масс муфты $v_{Ax} = \dot{\varphi}a \sin \varphi$, $v_{Ay} = -\dot{\varphi}a \cos \varphi$, и $v_A^2 = (\dot{\varphi}a)^2$. Угловая скорость муфты совпадает с угловой скоростью стержня KD , на который она надета. Кинетическая энергия системы (муфта 1 и поршень 2) имеет вид

$$T = \frac{m_1 v_A^2}{2} + \frac{J_1 \omega_1^2}{2} + \frac{m_2 v_C^2}{2},$$

где $\omega_{1z} = \dot{\varphi}/2$, или, с учетом выражений

$$v_C^2 = (\dot{\varphi}b)^2 \cos^2 \varphi, \quad v_A^2 = (\dot{\varphi}a)^2,$$

получаем вид

$$T = \frac{\dot{\varphi}^2 (C_1 + C_2 \cos^2 \varphi)}{2}, \quad (3.16)$$

где $C_1 = m_1 a^2 + J_1/4$, $C_2 = m_2 b^2$. Обобщенная сила

$$\begin{aligned} Q &= ((-m_1 g)v_{Ay} + (-m_2 g)v_{Cy} + M\omega_1) / \dot{\varphi} = \\ &= (am_1 - bm_2)g \cos \varphi + M/2. \end{aligned}$$

Имея выражение (3.16) и обобщенную силу, уравнение движения получаем по формуле (3.7), с. 93.

3.3.2. Системы с двумя степенями свободы

Уравнения Лагранжа 2-го рода для систем с двумя степенями свободы имеют следующий вид:

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_1} \right) - \frac{\partial T}{\partial q_1} &= Q_1, \\ \frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_2} \right) - \frac{\partial T}{\partial q_2} &= Q_2. \end{aligned} \quad (3.17)$$

Здесь q_1, q_2 — обобщенные координаты, Q_1, Q_2 — соответствующие обобщенные силы.

Для консервативных систем можно ввести функцию Лагранжа (кинетический потенциал)

$$L = T - \Pi,$$

где Π — потенциальная энергия. Уравнения Лагранжа 2-го рода примут вид

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_1} \right) - \frac{\partial L}{\partial q_1} &= 0, \\ \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_2} \right) - \frac{\partial L}{\partial q_2} &= 0. \end{aligned} \quad (3.18)$$

Задача 41. Механическая система с идеальными стационарными связями имеет две степени свободы и состоит из груза A и однородных цилиндров B и C . Цилиндр C падает вертикально вниз и передает движение цилиндру B и грузу A , с которыми он связан нерастяжимой нитью. Нить разматывается с цилиндра C . Даны массы тел m_A, m_B, m_C (рис. 156). Найти ускорение груза A .

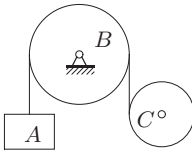


Рис. 156

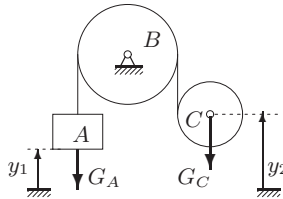


Рис. 157

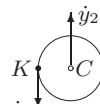


Рис. 158

Решение

Выбираем две независимые переменные $q_1 = y_1$ и $q_2 = y_2$, однозначно описывающие положение системы (рис. 157). Переменная y_1 указывает высоту груза по отношению к неподвижной системе отсчета, а y_2 — высоту оси цилиндра C .

Выражаем кинетическую энергию системы через обобщенные скорости \dot{y}_1 и \dot{y}_2 .

Кинетическую энергию всей системы представляем в виде суммы кинетических энергий груза и цилиндров: $T = T_A + T_B + T_C$.

Находим кинетическую энергию поступательного движения груза:

$$T_A = \frac{m_A \dot{y}_1^2}{2}.$$

Момент инерции цилиндра B равен $J_B = m_B R_B^2 / 2$. Кинетическая энергия вращающегося цилиндра:

$$T_B = \frac{J_B \omega_B^2}{2} = \frac{m_B R_B^2 (\dot{y}_1 / R_B)^2}{2 \cdot 2} = \frac{m_B \dot{y}_1^2}{4}.$$

Находим кинетическую энергию плоского движения цилиндра C :

$$T_C = \frac{m_C v_C^2}{2} + \frac{J_C \omega_C^2}{2}.$$

Скорость центра масс определена обобщенной координатой $v_{Cy} = \dot{y}_2$.

Угловую скорость цилиндра C найдем из графа $K \xrightarrow{0} C$ (рис. 158):

$$v_{Cy} = v_{Ky} + \omega_{Cz} R_C \cos 0,$$

или

$$\dot{y}_2 = -\dot{y}_1 + \omega_{Cz} R_C.$$

Откуда $\omega_{Cz} = (\dot{y}_1 + \dot{y}_2) / R_C$. Момент инерции цилиндра C равен $J_C = m_C R_C^2 / 2$. В итоге

$$T_C = \frac{m_C \dot{y}_2^2}{2} + \frac{m_C (\dot{y}_1 + \dot{y}_2)^2}{4}.$$

Кинетическая энергия всей системы:

$$T = (1/2) \dot{y}_1^2 (m_A + m_B/2 + m_C/2) + (1/2) \dot{y}_1 \dot{y}_2 m_C + (3/4) \dot{y}_2^2 m_C.$$

Обобщенную силу Q_1 вычисляем по формуле $Q_1 = \delta A_1 / \delta y_1$, где δA_1 — элементарная работа всех сил на перемещении δy_1 . Фиксируем высоту цилиндра C : $\delta y_2 = 0$. Груз A перемещается вверх на расстояние δy_1 , цилиндр B поворачивается на некоторый угол, а цилиндр C вращается вокруг неподвижной оси. На этом перемещении работу совершает только сила $G_A = m_A g$, следовательно, $\delta A_1 = -m_A g \delta y_1$ и $Q_1 = -m_A g$. Аналогично, фиксируя перемещение груза ($\delta y_1 = 0$), даем приращение высоте y_2 цилиндра C и вычисляем $Q_2 = -m_C g$. Знак минус в обоих

случаях возникает из-за того, что приращение δy и сила направлены в разные стороны¹.

Записываем уравнения Лагранжа (3.17) и вычисляем входящие в них производные:

$$\begin{aligned}\frac{\partial T}{\partial \dot{y}_1} &= \dot{y}_1(m_A + m_B/2 + m_C/2) + \dot{y}_2 m_C/2, \\ \frac{d}{dt} \left(\frac{\partial T}{\partial \dot{y}_1} \right) &= \ddot{y}_1(m_A + m_B/2 + m_C/2) + \ddot{y}_2 m_C/2, \\ \frac{\partial T}{\partial \dot{y}_2} &= \dot{y}_1 m_C/2 + (3/2)\dot{y}_2 m_C, \\ \frac{d}{dt} \left(\frac{\partial T}{\partial \dot{y}_2} \right) &= \dot{y}_1 m_C/2 + (3/2)\ddot{y}_2 m_C, \\ \frac{\partial T}{\partial y_1} &= 0, \quad \frac{\partial T}{\partial y_2} = 0.\end{aligned}$$

В результате уравнения Лагранжа принимают вид

$$\begin{aligned}(2m_A + m_B + m_C)\ddot{y}_1 + m_C\ddot{y}_2 &= -2m_A g, \\ m_C\ddot{y}_1 + 3m_C\ddot{y}_2 &= -2m_C g.\end{aligned}\tag{3.19}$$

Решаем систему уравнений (3.19) и находим ускорения:

$$\ddot{y}_1 = \frac{2g(m_C - 3m_A)}{6m_A + 3m_B + 2m_C}, \quad \ddot{y}_2 = -\frac{2g(m_A + m_B + m_C)}{6m_A + 3m_B + 2m_C}.$$

Как и в других задачах на составление уравнений Лагранжа 2-го рода, хорошей проверкой решения является решение при другом выборе обобщенных координат. В данном случае за одну координату можно взять угол поворота цилиндра B , за другую — угол поворота цилиндра C . Допустимы и другие комбинации, важно только, чтобы эти координаты были независимыми. Нельзя, например, брать угол поворота цилиндра B и высоту y_1 в качестве системы координат. Заметим также, что если учесть раскачивание груза и цилиндра C , то в решение добавятся еще две обобщенные координаты и задача существенно усложнится. Ускорения здесь уже не будут постоянными.

Maple-программа для решения этой задачи дана на с. 376.

Задача 42. По внутренней поверхности тонкого обода радиусом R катится однородный диск радиусом r . Обод катится по горизонтальной поверхности (рис. 159). Масса обода m_1 , диска — m_2 . Составить

¹В скалярном произведении, которое входит в выражение для работы, $\cos 180^\circ = -1$.

уравнение движения системы, найти частоту малых колебаний вокруг положения равновесия.

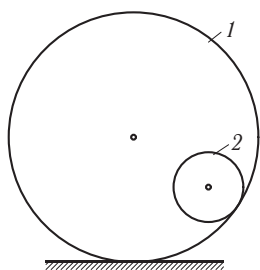


Рис. 159

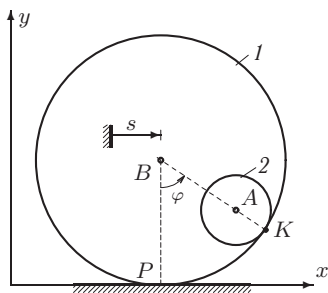


Рис. 160

Решение

Первая часть решения. Кинематика. Система имеет две степени свободы. Выберем обобщенные координаты — перемещение центра обода s и угол φ (рис. 160). Выразим угловые скорости обода и диска, а также скорость центра масс диска через обобщенные координаты. Начнем с графа $P \xrightarrow{\frac{R}{\pi/2}} B$ от мгновенного центра скоростей P обода до его геометрического центра. Первое уравнение этого графа имеет вид $v_{Bx} = v_{Px} - R\omega_{1z} \sin(\pi/2)$. Так как $v_{Bx} = \dot{s}$, а $v_{Px} = 0$, то отсюда получаем угловую скорость обода:

$$\omega_{1z} = -\dot{s}/R. \quad (3.20)$$

Все характерные точки задачи — точка контакта дисков, центры обода и диска непременно должны войти в кинематические графы. Составим граф $K \xrightarrow{\frac{R}{\varphi+\pi/2}} B$. Обе точки K и B лежат на обode, его угловая скорость ω_{1z} входит в уравнение графа: $v_{Bx} = v_{Kx} - R\omega_{1z} \sin(\varphi + \pi/2)$ или

$$\dot{s} = v_{Kx} - R\omega_{1z} \cos \varphi \quad (3.21)$$

Составим граф $A \xrightarrow{\frac{R-r}{\varphi+\pi/2}} B$. Теперь вершины графа A и B принадлежат некоторому условному (невесомому) стержню AB , составляющего угол φ с вертикалью и угол $\varphi + \pi/2$ с положительным направлением оси x и поэтому имеющему угловую скорость $\dot{\varphi}$. Исходя из этого получим

$$v_{Bx} = v_{Ax} - (R - r)\dot{\varphi} \sin(\varphi + \pi/2),$$

$$v_{By} = v_{Ay} + (R - r)\dot{\varphi} \cos(\varphi + \pi/2),$$

или

$$\dot{s} = v_{Ax} - (R - r)\dot{\varphi} \cos \varphi, \quad (3.22)$$

$$0 = v_{Ay} - (R - r)\dot{\varphi} \sin \varphi. \quad (3.23)$$

Эти уравнения определяют выражение скорости центра масс диска через обобщенные координаты. Осталось найти угловую скорость диска. Составим граф по диску, от точки K касания диска и обода до центра диска: $K \xrightarrow{\frac{r}{\varphi + \pi/2}} A$. Получим $v_{Ax} = v_{Kx} - r\omega_{2z} \sin(\varphi + \pi/2)$, или

$$v_{Ax} = v_{Kx} - r\omega_{2z} \cos \varphi. \quad (3.24)$$

Подставляем в это уравнение скорость v_{Kx} из (3.21) и скорость v_{Ax} из (3.22) и находим выражение для угловой скорости диска 2:

$$\omega_{2z} = ((r - R)\dot{\varphi} - \dot{s})/r. \quad (3.25)$$

Вторая часть решения. Кинетическая и потенциальная энергия. Находим кинетическую энергию обода, совершающего плоское движение

$$T_1 = \frac{m_1 v_B^2}{2} + \frac{J_1 \omega_1^2}{2}.$$

С учетом (3.20) и $J_1 = m_1 R^2$ получаем простое выражение

$$T_1 = m_1 \dot{s}^2. \quad (3.26)$$

Кинетическая энергия плоского движения диска:

$$T_2 = \frac{m_2 v_A^2}{2} + \frac{J_2 \omega_2^2}{2}, \quad (3.27)$$

где $J_2 = m_2 r^2/2$ — момент инерции однородного диска, а компоненты скорости точки A получены в соотношениях (3.22) и (3.23):

$$v_A^2 = v_{Ax}^2 + v_{Ay}^2 = \dot{s}^2 + 2\dot{s}\dot{\varphi}(R - r) \cos \varphi + (R - r)^2 \dot{\varphi}^2.$$

Суммарная энергия:

$$T = T_1 + T_2 = (3/4) m_2 (R - r)^2 \dot{\varphi}^2 + (1/2) m_2 (2 \cos \varphi + 1) (R - r) \dot{\varphi} \dot{s} + (1/4) (4 m_1 + 3 m_2) \dot{s}^2.$$

Потенциальная энергия системы равна

$$\Pi = m_2 g (R - (R - r) \cos \varphi). \quad (3.28)$$

Функция Лагранжа (кинетический потенциал) имеет вид

$$L = T - \Pi.$$

Запишем систему уравнений Лагранжа для консервативных систем:

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{s}} \right) - \frac{\partial L}{\partial s} &= 0, \\ \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\varphi}} \right) - \frac{\partial L}{\partial \varphi} &= 0. \end{aligned} \quad (3.29)$$

С учетом выражения для L после некоторых упрощений система дифференциальных уравнений движения примет вид

$$\begin{aligned} \ddot{s}(1 + 2 \cos \varphi) + 3\ddot{\varphi}(R - r) + 2g \sin \varphi &= 0, \\ \ddot{s}(4m_1 + 3m_2) + \ddot{\varphi}m_2(R - r)(1 + 2 \cos \varphi) - 2m_2(R - r)\dot{\varphi}^2 \sin \varphi &= 0. \end{aligned}$$

Линеаризуем уравнения. Пусть угол φ , перемещение s и их производные малы. Удерживаем в уравнениях линейные относительно φ , $\dot{\varphi}$, $\ddot{\varphi}$, s , \dot{s} , \ddot{s} слагаемые. Как обычно в таких случаях, полагаем $\sin \varphi \approx \varphi$, $\cos \varphi \approx 1$. Получаем

$$\begin{aligned} 3\ddot{s} + 3\ddot{\varphi}(R - r) + 2g\varphi &= 0, \\ \ddot{s}(4m_1 + 3m_2) + 3\ddot{\varphi}m_2(R - r) &= 0. \end{aligned} \quad (3.30)$$

Исключаем из уравнений \ddot{s} . В итоге переменная φ удовлетворяет уравнению $6m_1(R - r)\ddot{\varphi} + g(4m_1 + 3m_2)\varphi = 0$. Следовательно, частота малых колебаний системы имеет вид $k = \sqrt{g(4m_1 + 3m_2)/(6m_1(R - r))}$.

Maple-программа для вывода уравнений движения и получения частоты малых колебаний дана на с. 379.

3.4. Функция Рауса

Задача 43. Цилиндр радиуса r катится без проскальзывания по бруску. Брусок скользит по гладкой горизонтальной поверхности (рис. 161). Масса бруска 1 равна $4m$, масса точки 2, расположенной на ободе цилиндра, равна m , цилиндра 3 — $3m$. За обобщенные координаты принять смещение бруска x и угол поворота цилиндра φ . Найти функцию Рауса¹.

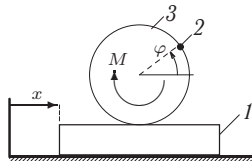


Рис. 161

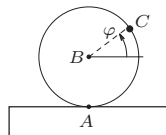


Рис. 162

¹Routh Edward John (1831–1907) — английский физик и математик.

Решение

Для начала решим задачу кинематики. Точку соприкосновения цилиндра и бруска обозначим A , ось цилиндра — B , точку на ободу — C (рис. 162). Найдем выражения скорости точки C и оси цилиндра через выбранные обобщенные скорости \dot{x} и $\dot{\varphi}$. Составим следующий кинематический граф:

$$A \xrightarrow{\frac{R}{\pi/2}} B \xrightarrow{\frac{R}{\varphi}} C.$$

Ему соответствуют два уравнения:

$$\begin{aligned} v_{Cx} &= v_{Ax} - r\dot{\varphi} \sin(\pi/2) - r\dot{\varphi} \sin \varphi, \\ v_{Cy} &= v_{Ay} + r\dot{\varphi} \cos(\pi/2) + r\dot{\varphi} \cos \varphi. \end{aligned}$$

С учетом $v_{Ax} = \dot{x}$, $v_{Ay} = 0$ получаем отсюда следующие выражения для проекций скоростей точки C :

$$\begin{aligned} v_{Cx} &= \dot{x} - r\dot{\varphi}(1 + \sin \varphi), \\ v_{Cy} &= r\dot{\varphi} \cos \varphi. \end{aligned}$$

Из графа $A \xrightarrow{\frac{R}{\pi/2}} B$ получаем компоненты скорости центра цилиндра:

$$v_{Bx} = \dot{x} - r\dot{\varphi} \sin(\pi/2) = \dot{x} - r\dot{\varphi}, \quad v_{By} = r\dot{\varphi} \cos(\pi/2) = 0.$$

Кинетическая энергия системы состоит из кинетической энергии поступательного движения бруска, кинетической энергии точки C и плоского движения цилиндра:

$$T = \frac{m_1 \dot{x}^2}{2} + \frac{m_2 (v_{Cx}^2 + v_{Cy}^2)}{2} + \frac{m_3 (v_{Bx}^2 + v_{By}^2)}{2} + \frac{J \dot{\varphi}^2}{2},$$

где $J = m_3 r^2 / 2$ — момент инерции однородного цилиндра. С учетом значений масс $m_1 = 4m$, $m_2 = m$, $m_3 = 3m$ и выражений для скоростей получим

$$T = m(4\dot{x}^2 - r\dot{x}\dot{\varphi}(4 + \sin \varphi) + r^2\dot{\varphi}^2(13/4 + \sin \varphi)).$$

Потенциальная энергия равна

$$\Pi = M\varphi + m_2 g r \sin \varphi.$$

Функция Лагранжа:

$$\begin{aligned} L = T - \Pi &= m(4\dot{x}^2 - r\dot{x}\dot{\varphi}(4 + \sin \varphi) + r^2\dot{\varphi}^2(13/4 + \sin \varphi)) - \\ &\quad - M\varphi - m_2 g r \sin \varphi. \end{aligned} \quad (3.31)$$

Обобщенная координата x не входит в выражение для L и является циклической. Найдем обобщенный импульс p_x , соответствующий циклической координате x . Имеем циклический интеграл

$$p_x = \frac{\partial L}{\partial \dot{x}} = m(8\dot{x} - r\dot{\varphi}(4 + \sin \varphi)).$$

В силу уравнения Лагранжа $p_x = \text{const}$. Выразим отсюда циклическую скорость:

$$\dot{x} = r\dot{\varphi}(4 + \sin \varphi)/8 + p_x/(8m). \quad (3.32)$$

С помощью преобразования Лежандра¹ запишем функцию Рауса, $R = p_x \dot{x} - L$. После подстановки сюда выражения для \dot{x} и некоторых преобразований получим²

$$R = -mr^2(35 + 8 \sin \varphi + \cos^2 \varphi)\dot{\varphi}^2/16 + p_x r \dot{\varphi}(4 + \sin \varphi)/8 + M\varphi + mgr \sin \varphi + p_x^2/(16m). \quad (3.33)$$

Функция Рауса удовлетворяет уравнению

$$\frac{d}{dt} \frac{\partial R}{\partial \dot{\varphi}} - \frac{\partial R}{\partial \varphi} = 0$$

или

$$r^2 m \dot{\varphi}^2 (\sin \varphi - 4) \cos \varphi - r^2 m \ddot{\varphi} (35 + 8 \sin \varphi + \cos^2 \varphi) - 8mgr \cos \varphi - 8M = 0. \quad (3.34)$$

Из этого уравнения можно найти $\varphi(t)$. Циклическая координата определяется из интеграла

$$x(t) = \int \frac{\partial R}{\partial p_x} dt = r(4\varphi - \cos \varphi)/8 + p_x t/(8m) + C, \quad (3.35)$$

где C — постоянная интегрирования. Константы C и p_x находят из начальных условий для $x(t)$. Заметим, что уравнения Лагранжа 2-го рода дают связанную систему уравнений:

$$mr^2(13 + 4 \sin \varphi)\ddot{\varphi} - 8mr\ddot{x} \sin \varphi + 2mr^2\dot{\varphi}^2 \cos \varphi + 2M + 2rmg \cos \varphi = 0, \\ r(4 + \sin \varphi)\ddot{\varphi} - 8\ddot{x} + r\dot{\varphi}^2 \cos \varphi = 0.$$

Maple-программа для определения функции Рауса, интегрирования уравнения (3.34) и анимации полученного решения дана на с. 383.

¹Adrien-Marie Legendre (1752–1833) — французский математик. Преобразованием Лежандра функции $X(x_1, x_2, \dots, x_n)$ называется функция $Y(y_1, y_2, \dots, y_n) = \sum_{i=1}^n x_i y_i - X$, где $y_i = \partial X / \partial x_i$ [38].

²В некоторых учебниках [13] функция Рауса берется с обратным знаком.

3.5. Уравнения Гамильтона

Задача 44. Сквозь муфту, закрепленную на вертикальном штоке, скользит горизонтальный стержень, соединенный шарниром с ободом цилиндра (рис. 163). Масса цилиндра m_1 , радиус — R . Цилиндр катится по горизонтальной плоскости без проскальзывания. Масса стержня m_2 . К цилиндру приложен момент M , к штоку — сила F . Составить уравнение движения системы в форме Гамильтона¹.

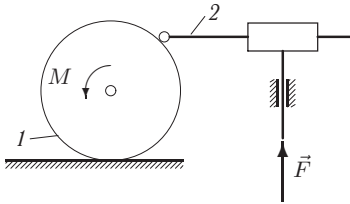


Рис. 163

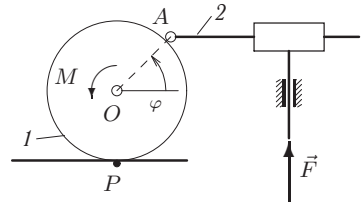


Рис. 164

Решение

В качестве обобщенной координаты принимаем угол φ поворота цилиндра. Кинетическая энергия однородного цилиндра, катящегося по поверхности без проскальзывания, согласно (3.2) равна $T_1 = (3/4)m_1 v_O^2$, где v_O — скорость центра цилиндра. Вычислим ее из графа от МЦС цилиндра P до его центра O : $P \xrightarrow{1/\pi/2} O$. Одно уравнение этого графа дает очевидный результат: $v_{Oy} = 0$, из другого уравнения (в проекции на ось x) следует $v_{Ox} = -R\dot{\varphi}$.

Стержень 2 движется поступательно: $T_2 = (1/2)m_2 v_A^2$. Скорость v_A найдем из графа $P \xrightarrow{1/\pi/2} O \xrightarrow{1/\varphi} A$:

$$\begin{aligned} v_{Ax} &= v_{Px} - R\dot{\varphi} \sin(\pi/2) - R\dot{\varphi} \sin \varphi, \\ v_{Ay} &= v_{Py} + R\dot{\varphi} \cos(\pi/2) + R\dot{\varphi} \cos \varphi. \end{aligned} \quad (3.36)$$

Отсюда получаем

$$v_A^2 = 2R^2 \dot{\varphi}^2 (1 + \sin \varphi). \quad (3.37)$$

Кинетическая энергия системы имеет вид

$$T = T_1 + T_2 = R^2 \dot{\varphi}^2 ((3/4)m_1 + m_2(1 + \sin \varphi)). \quad (3.38)$$

¹ William Rowan Hamilton (1805–1865) — английский математик.

Вычислим потенциальную энергию системы как работу, которую совершают внешние силы при перемещении в некоторое условное нулевое положение. За это положение возьмем $\varphi = 0$ и получим

$$\Pi = -M\varphi + (m_2g - F)R \sin \varphi. \quad (3.39)$$

Запишем функцию Лагранжа

$$L = T - \Pi = R^2\dot{\varphi}^2((3/4)m_1 + m_2(1 + \sin \varphi)) + M\varphi - (m_2g - F)R \sin \varphi. \quad (3.40)$$

Вычислим обобщенный импульс

$$p = \frac{\partial L}{\partial \dot{\varphi}} = 2R^2\dot{\varphi}((3/4)m_1 + m_2(1 + \sin \varphi)). \quad (3.41)$$

Отсюда

$$\dot{\varphi} = 2p/(R^2(3m_1 + 4m_2(1 + \sin \varphi))). \quad (3.42)$$

Функция Гамильтона для системы с одной степенью свободы имеет вид

$$H = p\dot{\varphi} - L. \quad (3.43)$$

Подставим сюда (3.42) и (3.40). После преобразований получим¹

$$H = p^2/(R^2(3m_1 + 4m_2(1 + \sin \varphi))) - M\varphi + (m_2g - F)R \sin \varphi. \quad (3.44)$$

Уравнения движения системы в форме Гамильтона (или канонические уравнения)

$$\dot{\varphi} = \frac{\partial H}{\partial p}, \quad \dot{p} = -\frac{\partial H}{\partial \varphi}$$

примут вид

$$\begin{aligned} \dot{\varphi} &= 2p/(R^2(3m_1 + 4m_2(1 + \sin \varphi))), \\ \dot{p} &= 4p^2m_2 \cos \varphi / (R^2(3m_1 + 4m_2(1 + \sin \varphi))^2 + \\ &\quad + M - (m_2g - F)R \cos \varphi. \end{aligned} \quad (3.45)$$

Maple–программа для вывода канонических уравнений этой системы дана на с. 386.

Гамильтонова форма уравнений движения используется в физике для релятивистских частиц во внешних полях [24].

3.6. Колебание узла фермы

Задача 45. Дана симметричная ферма с прямоугольной решеткой и восходящими раскосами, состоящая из $2n$ панелей одинаковой длины

¹Если кинетическую энергию представить в виде $T = (1/2)\dot{\varphi}^2 f(\varphi)$, то функция Гамильтона примет вид $H = p^2/(2f(\varphi)) + \Pi$.

$a = 3$ м (рис. 165). В центральный узел нижнего пояса помещен груз массой $m = 1000$ кг. Жесткость всех стержней одинакова и равна $EF = 5 \cdot 10^4$ кН; высота фермы $h = 4$ м. Найти собственные частоты колебания груза.

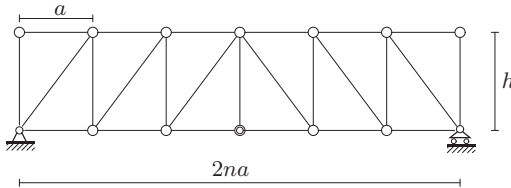


Рис. 165

Решение

Груз в узле плоской фермы имеет две степени свободы. Основные уравнения задачи следуют из уравнения Лагранжа 2-го рода (3.17), с. 108. В качестве обобщенных координат q_1, q_2 принимаем горизонтальные и вертикальные перемещения узла x_1 и x_2 . Предполагая, что упругие силы линейно зависят от перемещений, записываем уравнение Лагранжа 2-го рода в матричном виде:

$$A\ddot{\vec{x}} + C\vec{x} = 0, \quad (3.46)$$

где A — матрица инерции, C — матрица жесткости. Матрица, обратная C , — матрица податливости $B = C^{-1}$, коэффициенты которой (перемещения от единичных сил) вычисляем по формуле Максвелла–Мора¹ [58]:

$$b_{i,j} = b_{j,i} = \sum_{\mu=1}^n \frac{S_{\mu,i} S_{\mu,j} l_{\mu}}{E_{\mu} F_{\mu}}, \quad i, j = 1, 2, \quad (3.47)$$

где l_{μ} — длины стержней, E_{μ} и F_{μ} — модуль упругости и площадь поперечного сечения стержней, $S_{\mu,i}$ — безразмерное усилие в стержне с номером μ от действия единичной вертикальной ($i = 1$) или горизонтальной ($i = 2$) нагрузки на шарнир с массой. Очевидная симметрия $b_{i,j} = b_{j,i}$ выражает теорему взаимности Бетти².

Произведение $E_{\mu} F_{\mu}$ называют жесткостью, в данной задаче она считается одинаковой для всех стержней фермы $E_{\mu} = E$, $F_{\mu} = F$. Суммирование производится по всем n стержням фермы.

Коэффициенты $b_{i,j}$ имеют простой физический смысл: $b_{i,j}$ — это перемещение узла в направлении i под действием единичной силы, действующей в направлении j . Измеряются $b_{i,j}$ в м/Н.

¹Christian Otto Mohr (1835–1918) — немецкий механик.

²Betti Enrico (1823–1892) — итальянский математик.

Кинетическая энергия точки на плоскости имеет вид $T = m(\dot{x}_1^2 + \dot{x}_2^2)/2$, следовательно, матрица инерции является диагональной:

$$A = \begin{pmatrix} m & 0 \\ 0 & m \end{pmatrix}.$$

Умножаем (3.46) на B и делаем подстановку $\ddot{\vec{x}} = -\omega^2 \vec{x}$, что равносильно замене $x_1 = A_1 \sin(\omega t + \beta_0)$, $x_2 = A_2 \sin(\omega t + \beta_0)$, где A_1, A_2 — амплитуды, ω — частота, β_0 — начальная фаза колебаний. Получаем однородную систему $m\omega^2 B\vec{x} - \vec{x} = 0$, или

$$B\vec{x} = \lambda \vec{x},$$

где $\lambda = 1/(m\omega^2)$, имеющую ненулевое решение в том случае, если ее определитель равен нулю. Следовательно, задача свелась к поиску собственных значений λ матрицы B .

Используем принятые в строительной механике обозначения: $O_{k,i}$ — для усилий верхнего пояса, $D_{k,i}$ — для диагоналей (раскосов), $U_{k,i}$ — для нижнего пояса и $V_{k,i}$ — для вертикальных стержней (рис. 166). Индекс k указывает на номер панели, i — номер нагрузки. Напряженное состояние от действия вертикальной силы обозначим номером 1, горизонтальной — 2. На рисунке изображена ферма с $n = 3$ панелями в половине пролета и указаны обозначения для панели $k = 2$. Ферма освобождена от связей, реакции которых X_A, Y_A, Y_B . Приложены обе единичные силы к узлу, имеющему массу. Изображение нагрузок условное. В действительности они прикладываются поочередно, а не одновременно, и для каждой из них определяются реакции опор и усилия в стержнях.

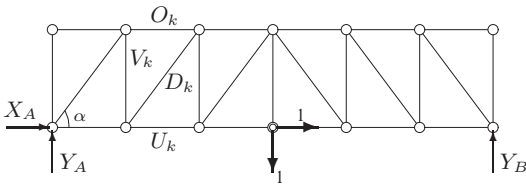


Рис. 166

Таким образом, имеем следующий план решения [27]:

1. Прикладываем к центральному узлу нижнего пояса фермы единичную (безразмерную) вертикальную силу. Определяем усилия в стержнях $U_{k,1}, O_{k,1}, D_{k,1}, V_{k,1}$, $k = 1, \dots, n$.

2. К этому же узлу прикладываем единичную горизонтальную силу. Определяем усилия в стержнях $U_{k,2}, O_{k,2}, D_{k,2}, V_{k,2}$.

3. Используя формулу Максвелла–Мора, вычисляем коэффициенты податливости $b_{i,j}$. Записываем их в симметричную матрицу:

$$B = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}. \quad (3.48)$$

4. Вычисляем собственные значения $\lambda_{1,2}$ матрицы B^1 .

5. Находим частоты собственных колебаний: $\omega_1 = 1/\sqrt{m\lambda_1}$, $\omega_2 = 1/\sqrt{m\lambda_2}$.

Итак, сначала прикладываем к среднему узлу фермы единичную вертикальную силу. Реакции опор: $Y_{A,1} = Y_{B,1} = 1/2$, $X_{A,1} = 0$.

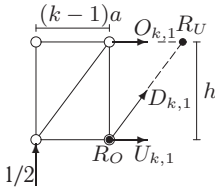


Рис. 167

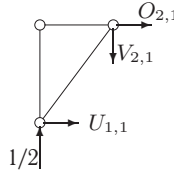


Рис. 168

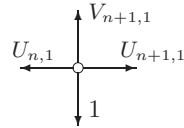


Рис. 169

Учитывая симметрию задачи, выпишем усилия в стержнях только для левой половины фермы. Определяем усилия методом Риттера, выполняя вертикальное сечение панели (рис. 167). Усилие $D_{k,1}$ находим из уравнения проекции на вертикальную ось, усилия $U_{k,1}$ и $O_{k,1}$ — из уравнений моментов относительно соответствующих точек Риттера (обозначены на рисунке как R_U и R_O):

$$\begin{aligned} U_{k,1} &= ka/(2h), & O_{k,1} &= -(k-1)a/(2h), \\ D_{k,1} &= -1/(2 \sin \alpha), & k &= 1, \dots, n. \end{aligned} \quad (3.49)$$

Здесь $\sin \alpha = h/l$, $l = \sqrt{a^2 + h^2}$ — длина раскоса.

Усилия в стойках находим методом Риттера, выполняя наклонное сечение двух соседних панелей (рис. 168) и записывая уравнения проекции $V_{k,1} = 1/2$, $k = 2, \dots, n$. На рисунке дано сечение панелей 1 и 2, очевидно, для других панелей уравнение проекций будет тем же. В крайних стойках усилие равно нулю: $V_{1,1} = 0$, $V_{2n+1,1} = 0$. Усилие в средней стойке находим методом вырезания узла: $V_{n+1,1} = 1$ (рис. 169).

Определяем усилия в стержнях от действия единичной горизонтальной силы. Здесь расчет значительно проще. Работает только левая половина нижнего пояса фермы. Усилия в остальных стержнях равны нулю. Имеем реакции опор: $Y_{A,2} = Y_{B,2} = 0$, $X_{A,2} = -1$. Усилия:

¹Собственные значения в системе **Maple** можно вычислить с помощью оператора **Eigenvalues(M)** из пакета **LinearAlgebra**, где **M** — матрица, заданная оператором **Matrix**, например так: **M:=Matrix([[1,2],[2,1]])**.

$U_{k,2} = 1, k = 1, \dots, n, O_{i,2} = D_{i,2} = V_{i,2} = 0, i = 1, \dots, 2n, V_{2n+1,2} = 0, U_{j,2} = 0, j = n + 1, \dots, 2n.$

По формуле Максвелла–Мора (3.47) определяем коэффициенты

$$b_{11} = \frac{1}{EF} \left(2 \sum_{k=1}^n (U_{k,1}^2 a + O_{k,1}^2 a + D_{k,1}^2 l + V_{k,1}^2 h) + V_{n+1,1}^2 h \right), \quad (3.50)$$

$$b_{22} = \frac{1}{EF} \sum_{k=1}^n U_{k,2}^2 a, \quad b_{12} = \frac{1}{EF} \sum_{k=1}^n U_{k,1} U_{k,2} a.$$

Учитывая, что $\sum_{k=1}^n k = n(n+1)/2, \sum_{k=1}^n k^2 = n(n+1)(2n+1)/6,$
 $\sum_{k=1}^n (k-1)^2 = n(n-1)(2n-1)/6,$ получим¹

$$b_{11} = \frac{1}{2EF} \left(\frac{a^3 n(2n^2 + 1)}{3h^2} + \frac{n(l^3 + h^3)}{h^2} + h \right), \quad (3.51)$$

$$b_{22} = \frac{na}{EF}, \quad b_{12} = \frac{n(n+1)a^2}{4EFh}.$$

Характеристическое уравнение матрицы (3.48) имеет вид $(b_{11} - \lambda)(b_{22} - \lambda) - b_{12}^2 = 0,$ или, с учетом (3.51),

$$\lambda^2 - \frac{\lambda(3h^3(n+1) + a^3 n(2n^2 + 1) + 3n(l^3 + 2h^2 a))}{6h^2 EF} + \frac{na(24h^3(n+1) + a^3 n(13n^2 - 6n + 5) + 24nl^3)}{48(hEF)^2} = 0. \quad (3.52)$$

Решаем квадратное уравнение, определяем его корни $\lambda_1, \lambda_2,$ по которым вычисляем частоты колебаний $\omega_1 = 1/\sqrt{m\lambda_1}$ и $\omega_2 = 1/\sqrt{m\lambda_2}.$ Подставляя данные задачи $a = 3$ м, $h = 4$ м, $n = 5, m = 1000$ кг, $EF = 5 \cdot 10^4$ кН, получаем $\omega_1 = 21,681$ рад/с, $\omega_2 = 64,866$ рад/с.

Замечание 1. Решение уравнения (3.52) имеет предельные значения при $n \rightarrow \infty.$ Предположим, что пролет фермы задан: $L = 2na.$ Переходя к пределу получим $\lambda_1 \rightarrow \infty, \lambda_2 \rightarrow L/(2EF).$

Замечание 2. Величина b_{11} представляет собой прогиб середины пролета фермы от действия единичной вертикальной силы, приложенной в этой же точке. Выбирая отношение высоты фермы к длине пролета, можно найти такое соотношение этих величин, при котором прогиб будет наименьшим. Это известная задача увеличения жесткости фермы. Введем безразмерный коэффициент $k = h/(2na).$ Очевидно, $k < 1.$ Для определения минимума b_{11} приравняем нулю производную

¹Суммирование в Maple: `factor(sum(k, k=1..n)).` Оператор `factor` служит для разложения выражения на множители.

$d(b_{11})/dk$. Уравнение для k примет вид

$$12k^3n^3 + 12k^3n^2 - 2n^2 - 1 - 3(1 - 2n^2k^2)\sqrt{1 + 4n^2k^2} = 0. \quad (3.53)$$

Пренебрегая 1 по сравнению n^2k^2 , что возможно для больших n , получаем простое уравнение $12k^3n + 6k^3 - 1 = 0$, из которого находим оптимальное соотношение

$$k = \sqrt[3]{1/(6 + 12n)}. \quad (3.54)$$

Для сравнения приближенного результата и решения (3.53) найдем минимум с помощью оператора `minimize(b11,k=0.01..1,location)`. Опция `location` включена для того чтобы найти не только минимум функции, но и получить соответствующее значение аргумента, т. е. k . Граница 0,01 поиска минимума сдвинута от нуля для исключения особенности, присутствующей здесь в нуле. Сравнение показывает, что погрешность решения (3.54) при $n = 4$ равна 6%, и с увеличением n она падает. При $n = 16$ получаем $k = 0,17157$, что отличается от решения, найденного оператором `minimize`, менее чем на 0,9%.

Замечание 3. В данной задаче решение получено как функция числа пролетов n . Для большинства подобных задач такую зависимость найти трудно, и обычно расчет проводят при каком-то фиксированном значении n . Однако с появлением системы **Maple** появилась возможность индуктивного вывода зависимости от n . Предположим, что при $n = 1$ получено аналитическое решение в виде формулы, один из коэффициентов которой равен j_1 . При $n = 2$ этот же коэффициент равен j_2 и т. д. Здесь j_n — целые числа. Получить искомую зависимость можно с помощью оператора `rgf_findrecur` из пакета `genfunc`.

Поясним это на примере. В результате шестикратного (с естественно повышающейся сложностью) решения некоторой задачи найдена последовательность чисел 3, 9, 19, 33, 51, 73. Какое будет следующее число? Для больших n непосредственные вычисления затруднительны. Прибегаем к методу индукции. Сначала получаем рекуррентное уравнение

$\> S := \text{rgf_findrecur}(3, [3, 9, 19, 33, 51, 73], t, n);$

$$S := t(n) = 3t(n-1) - 3t(n-2) + t(n-3)$$

В качестве аргумента функции `rgf_findrecur` берутся пары чисел исследуемой последовательности. Первый аргумент — число этих пар. Два последних аргумента `t` и `n` — обозначение для функции и ее аргумента. Если в результате получается выражение, содержащее дроби, число пар необходимо увеличить. Порядок рекуррентного уравнения при этом возрастет.

Далее решаем уравнение, обозначенное как S , с использованием начальных значений (из самой же последовательности)

```
> simplify(rsolve({S,t(1)=3,t(2)=9,t(3)=19},t(n)));
2n^2 + 1
```

Зависимость найдена. В качестве примера мы взяли один из коэффициентов в выражении (3.51).

3.7. Уравнения движения с множителями связей

Задача 46. Невесомые стержни AO и BC шарнирно соединены в точке A . Стержень AO вращается на неподвижном шарнире O , стержень BC прикреплен к вертикальному ползуну B . В точке C сосредоточена масса m . Механизм¹ расположен в вертикальной плоскости (рис. 170). Дано: $AO = a$, $AC = b$, $AB = c$. Найти частоту малых колебаний системы около положения равновесия.

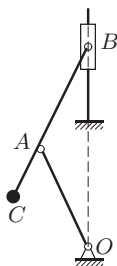


Рис. 170

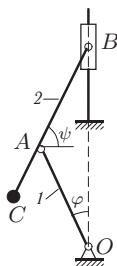


Рис. 171

Решение

Механизм имеет одну степень свободы. В качестве обобщенной координаты выберем угол φ . Стержни AO и BC обозначим номерами 1 и 2. Предложим два способа решения задачи.

Способ 1.

Кинетическая энергия механизма сосредоточена в точке C :

$$T = \frac{mv_C^2}{2}.$$

¹Вариант паллографа *О. Шлика* — прибора для записи вибраций корпуса корабля. В оригинальном паллографе вместо ползуна стоит качающаяся муфта, в которой скользит стержень BC . Прибор состоит из двух независимых частей. Здесь изучается часть, ответственная за горизонтальные колебания. См. *Николаи Е. Л.* Теоретическая механика. Ч. 2. Динамика. — М.: Государственное издательство физико-математической литературы, 1958.

Для определения скорости составим граф $O \xrightarrow[\varphi+\pi/2]{a} A \xrightarrow[\psi+\pi]{b} C$. Ему соответствуют два уравнения:

$$\begin{aligned} v_{Cx} &= v_{Ox} - a\omega_{1z} \sin(\varphi + \pi/2) - b\omega_{2z} \sin(\psi + \pi), \\ v_{Cy} &= v_{Oy} + a\omega_{1z} \cos(\varphi + \pi/2) + b\omega_{2z} \cos(\psi + \pi). \end{aligned}$$

С учетом $v_{Ox} = v_{Oy} = 0$, $\omega_{1z} = \dot{\varphi}$, получаем отсюда следующие выражения для проекций скоростей точки C :

$$\begin{aligned} v_{Cx} &= -a\dot{\varphi} \cos \varphi + b\omega_{2z} \sin \psi, \\ v_{Cy} &= -a\dot{\varphi} \sin \varphi - b\omega_{2z} \cos \psi. \end{aligned} \quad (3.55)$$

Угловую скорость ω_{2z} , входящую в эти выражения, выразим через $\dot{\varphi}$ с помощью графа $O \xrightarrow[\varphi+\pi/2]{a} A \xrightarrow[\psi]{c} B$. Получим

$$v_{Bx} = -a\dot{\varphi} \cos \varphi - c\omega_{2z} \sin \psi = 0.$$

С учетом геометрического соотношения $\psi = \arccos((a/c) \sin \varphi)$ находим

$$\omega_{2z} = -\frac{\dot{\varphi} a \cos \varphi}{\sqrt{c^2 - a^2 \sin^2 \varphi}}. \quad (3.56)$$

Для проверки этого значения положим $a = c$. В этом случае треугольник OAB становится равнобедренным и, очевидно, стержни AO и BC имеют равные по модулю, но противоположно направленные угловые скорости, $\omega_{2z} = -\dot{\varphi}$. Это следует также из (3.56).

После преобразований получаем кинетическую энергию

$$T = \frac{ma^2\dot{\varphi}^2}{2c^2} \left((b+c)^2 \cos^2 \varphi + \sin^2 \varphi \frac{(ab \cos \varphi - c\sqrt{c^2 - a^2 \sin^2 \varphi})^2}{c^2 - a^2 \sin^2 \varphi} \right).$$

Обобщенная сила имеет вид

$$Q = \frac{-mgv_{Cy}}{\dot{\varphi}} = mga \sin \varphi \left(1 - \frac{ab \cos \varphi}{c\sqrt{c^2 - a^2 \sin^2 \varphi}} \right).$$

Рассмотрим малые отклонения механизма от положения равновесия. Для того чтобы уравнение Лагранжа было линейным по φ , $\dot{\varphi}$ и $\ddot{\varphi}$, необходимо линеаризовать обобщенную силу Q , оставив только слагаемые первой степени по указанным переменным, а в выражении для кинетической энергии оставить и квадраты этих величин, так как в уравнение входят производные T по φ и $\dot{\varphi}$. Имеем

$$T = \frac{ma^2\dot{\varphi}^2(b+c)^2}{2c^2}, \quad Q = mga\varphi \frac{c^2 - ab}{c^2}.$$

Уравнение Лагранжа примет вид

$$\ddot{\varphi}a(b+c)^2 + \varphi g(ab-c^2) = 0. \quad (3.57)$$

Это уравнение описывает малые колебания системы с частотой

$$k = \frac{\sqrt{g(b-c^2/a)}}{b+c}. \quad (3.58)$$

Очевидно, решение существует при $c < \sqrt{ab}$. Меняя размеры a , b , c , можно в широких пределах менять собственную частоту колебаний. При $c^2 = ab$ частота обращается в ноль. Интересно заметить, что если стержни OA и BC имеют одинаковую длину, т. е. $a = b + c$, то найденное критическое соотношение $c/a = (a-c)/c$ совпадает с соотношением золотого сечения.

Maple-программа этого решения дана на с. 387.

Способ 2. Множители Лагранжа.

В первом способе решения неявно присутствует вторая обобщенная координата — ψ . Она зависит от выбранной обобщенной координаты φ и легко выражается через нее. Более того, можно ввести еще одну координату y (рис. 172) и наложить на три координаты две связи:

$$\begin{aligned} f_1 &= a \cos \varphi + c \sin \psi - y = 0, \\ f_2 &= a \sin \varphi - c \cos \psi = 0. \end{aligned} \quad (3.59)$$

С тремя обобщенными координатами механизм имеет три степени свободы: одна степень свободы (поворот вокруг опоры) у стержня AO и две — у стержня BC (поворот и вертикальное перемещение). Теперь надо решить задачу с тремя обобщенными координатами и двумя уравнениями связи, для которых справедливы уравнения [38]

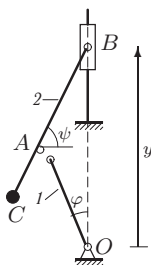


Рис. 172

$$\frac{d}{dt} \frac{\partial T}{\partial \dot{q}_i} - \frac{\partial T}{\partial q_i} = Q_i + \sum_{k=1}^2 \lambda_k \frac{\partial f_k}{\partial q_i}, \quad i = 1, 2, 3. \quad (3.60)$$

В данной задаче получается пять уравнений (3.59–3.60) для трех обобщенных координат $q_1 = \varphi$, $q_2 = \psi$, $q_3 = y$ и двух неопределенных множителей λ_1 и λ_2 .

Вычислим кинетическую энергию через выбранные обобщенные скорости. Составим граф $C \xrightarrow{b+c} B$. Ему соответствуют два уравнения:

$$v_{Bx} = v_{Cx} - (b+c)\omega_{2z} \sin \psi, \quad v_{By} = v_{Cy} + (b+c)\omega_{2z} \cos \psi.$$

С учетом $v_{Bx} = 0$, $v_{By} = \dot{y}$, $\omega_{2z} = \dot{\psi}$ получаем отсюда следующие выражения для проекций скоростей точки C :

$$v_{Cx} = (b+c)\dot{\psi} \sin \psi, \quad v_{Cy} = \dot{y} - (b+c)\dot{\psi} \cos \psi.$$

Получаем кинетическую энергию

$$T = mv_C^2/2 = m(\dot{\psi}^2(b+c)^2 + \dot{y}^2 - 2(b+c)\dot{\psi}\dot{y} \cos \psi)/2.$$

Мощность сил, действующих на систему, $N = -mgv_{Cy} = -mg(\dot{y} - (b+c)\dot{\psi} \cos \psi)$. Обобщенные силы имеют вид

$$Q_1 = \frac{\partial N}{\partial \dot{\varphi}} = 0, \quad Q_2 = \frac{\partial N}{\partial \dot{\psi}} = mg(b+c) \cos \psi, \quad Q_3 = \frac{\partial N}{\partial \dot{y}} = -mg.$$

Запишем уравнения (3.60):

$$\lambda_1 a \sin \varphi - \lambda_2 a \cos \varphi = 0,$$

$$m(b+c) \left(\ddot{\psi}(b+c) - \ddot{y} \cos \psi + \dot{y}\dot{\psi} \sin \psi - g \cos \psi \right) - \lambda_1 c \cos \psi - \lambda_2 c \sin \psi = 0, \quad (3.61)$$

$$m\ddot{y} - m(b+c)\ddot{\psi} \cos \psi + m\dot{\psi}^2(b+c) \sin \psi + mg + \lambda_1 = 0.$$

Полагая малыми углы φ , $\alpha = \pi/2 - \psi$, их скорости $\dot{\varphi}$, $\dot{\alpha}$, ускорения $\ddot{\varphi}$, $\ddot{\alpha}$ и скорость \dot{y} , линеаризуем систему (3.61) и уравнения связи (3.59)

$$\lambda_1 a \varphi - \lambda_2 a = 0,$$

$$\ddot{\alpha} m(b+c)^2 + \alpha m g(b+c) + \alpha \lambda_1 c + \lambda_2 c = 0, \quad (3.62)$$

$$m g + \lambda_1 = 0, \quad a + c - y = 0, \quad a \varphi - c \alpha = 0.$$

Исключаем из системы (3.62) неопределенные множители, α и y , получаем уравнение колебаний (3.57). Оба способа дают один и тот же результат. Во втором способе не потребовалось использовать обратные тригонометрические функции и система уравнений получилась проще. В первом способе уравнение движения настолько громоздкое, что приводить его целиком не имеет смысла — ни проанализировать, ни попытаться его решить нет возможности. Заметим, что во втором способе решения допустим граф $O \xrightarrow[\varphi+\pi/2]{a} A \xrightarrow[\psi+\pi]{b} C$ по «разорванной цепи». Система уравнений получится другая, в ней вместо \ddot{y} будет ускорение $\ddot{\varphi}$. Результат (уравнение малых колебаний) при этом не изменится.

Maple-программа этого решения дана на с. 389.

СОПРОТИВЛЕНИЕ МАТЕРИАЛОВ

4.1. Моменты инерции сечения

Геометрические характеристики сечения, к которым относятся площадь, координаты центра тяжести и моменты инерции, встречаются в ряде задач механики. Существуют различные способы их вычисления (см. с. 391), в том числе и те, в которых существенным образом задействованы возможности системы символьной математики. Наиболее распространенный — метод разбиения фигуры на простейшие части, для которых моменты инерции известны. Применим этот метод для сечения, состоящего из стандартных прокатных профилей.

Моменты инерции плоской фигуры относительно произвольных осей x и y определяются интегралами по площади фигуры:

$$J_x = \iint_F y^2 dF, \quad J_y = \iint_F x^2 dF.$$

Центробежный и полярный моменты инерции равны соответственно

$$J_{xy} = \iint_F xy dF, \quad J_p = \iint_F (x^2 + y^2) dF = J_x + J_y. \quad (4.1)$$

Для большинства распространенных в технике форм сечений имеются таблицы моментов инерции (табл. 7–11, с. 485–489)¹.

Задача 47. Найти моменты инерции и ориентацию главных осей сечения, составленного из швеллера 18а, ГОСТ 8240-89, и уголка № 6.3/4, ГОСТ 8509-86 (рис. 173).

¹ В Maple, начиная с версии 13, двойной интеграл можно вычислять так: `int(x*y, y=0..x, x=0..1)`, а раньше: `int(int(x*y, y=0..x), x=0..1)`.



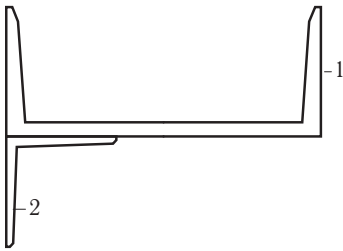


Рис. 173

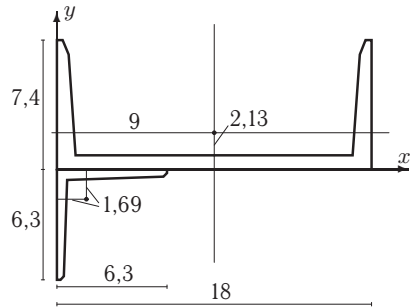


Рис. 174

Решение

1. Вводим произвольную систему координат. Определяем координаты центров тяжести фигур, составляющих сечение (рис. 174). Согласно табл. 10, с. 488 (строка выделена полужирным шрифтом) имеем:

Фигура 1 (швеллер № 18а), $x_1 = 9$ см, $y_1 = 2,13$ см. Здесь $y_1 = z_0$, $x_1 = h/2$, где h — стандартное обозначение для высоты швеллера, z_0 — расстояние до центра тяжести согласно таблице 10, с. 488.

Фигура 2 (уголок № 6.3/4), согласно таблице 11, с. 489, имеем $x_2 = 1,69$ см, $y_2 = -1,69$ см.

2. Вычисляем площадь всей фигуры. Площадь сечения швеллера $22,2$ см², уголка — $4,96$ см²: $F = \sum_i F_i = 22,2 + 4,96 = 27,16$ см².

3. Определяем координаты центра тяжести составного сечения:

$$x_c = \frac{\sum_i F_i x_i}{F} = \frac{22,2 \cdot 9 + 4,96 \cdot 1,69}{27,16} = 7,67 \text{ см},$$

$$y_c = \frac{\sum_i F_i y_i}{F} = \frac{22,2 \cdot 2,13 - 4,96 \cdot 1,69}{27,16} = 1,43 \text{ см}.$$
(4.2)

4. Определяем осевые J_{xc} , J_{yc} моменты инерции сечения относительно центральных осей. Моменты инерции швеллера берем из таблицы 10: $J_{xc_1} = 105$ см⁴, $J_{yc_1} = 1190$ см⁴. Заметим, здесь при работе с таблицей надо согласовать оси. Оси в условии задачи повернуты относительно таблицы на 90° . Для симметричной фигуры $J_{xyc_1} = 0$. С уголком проще, согласно таблице 11: $J_{xc_2} = J_{yc_2} = 18,9$ см⁴. Итак,

вычисляем:

$$\begin{aligned}
 J_{xc} &= \sum_{i=1}^2 (J_{xc_i} + (y_c - y_i)^2 F_i) = 105 + 22,2 \cdot (2,13 - 1,43)^2 + \\
 &\quad + 18,9 + 4,96 \cdot (1,69 + 1,43)^2 = 183,06 \text{ см}^4, \\
 J_{yc} &= \sum_{i=1}^2 (J_{yc_i} + (x_c - x_i)^2 F_i) = 1190 + 22,2 \cdot (9 - 7,67)^2 + \\
 &\quad + 18,9 + 4,96 \cdot (1,69 - 7,67)^2 = 1425,54 \text{ см}^4.
 \end{aligned} \tag{4.3}$$

В табл. 11 на с. 489 указан только *модуль* центробежного момента инерции уголка. Знак определяется его поворотом относительно

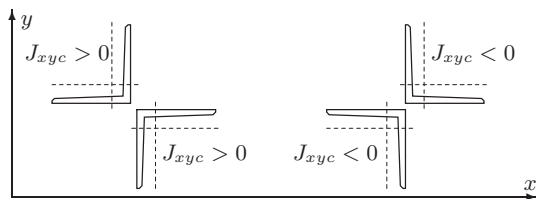


Рис. 175

выбранных осей. Центробежный момент инерции уголка положительный, если его биссектриса лежит в 2-й и 4-й четвертях, и отрицательный, если биссектриса лежит в 1-й и 3-й четвертях (рис. 175). Таким образом, $J_{xyc_2} = 11 \text{ см}^4$. Получим

$$\begin{aligned}
 J_{xyc} &= \sum_{i=1}^2 (J_{xyc_i} + (x_c - x_i)(y_c - y_i)F_i) = 22,2 \cdot (9 - 7,67)(2,13 - 1,43) + \\
 &\quad + 11 + 4,96 \cdot (1,69 - 7,67)(-1,69 - 1,43) = 124,21 \text{ см}^4.
 \end{aligned}$$

5. Находим главные моменты инерции:

$$J_{\max, \min} = \frac{J_{yc} + J_{xc}}{2} \pm \sqrt{\frac{(J_{yc} - J_{xc})^2}{4} + J_{xyc}^2} = 804,3 \pm 633,54, \tag{4.4}$$

$$J_{\max} = 1437,84 \text{ см}^4, \quad J_{\min} = 170,76 \text{ см}^4.$$

6. Вычисляем главные радиусы инерции: $i_{\max} = \sqrt{J_{\max}/F} = \sqrt{1437,84/27,16} = 1,39 \text{ см}$, $i_{\min} = \sqrt{J_{\min}/F} = \sqrt{170,76/27,16} = 0,48 \text{ см}$.

7. Находим направление главных осей:

$$\operatorname{tg} \alpha_{\max} = \frac{J_{xyc}}{J_{yc} - J_{\max}} = \frac{124,21}{1425,54 - 1437,84} = -10,10, \tag{4.5}$$

$$\alpha_{\max} = \operatorname{arctg}(-10,10) = -1,47 = -84^\circ.$$

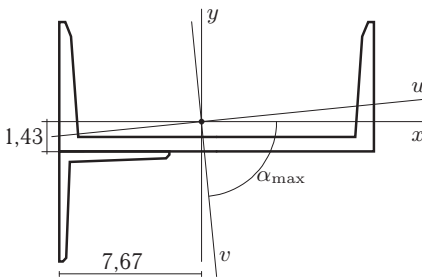


Рис. 176

Направление главных осей можно вычислить также по формуле

$$\operatorname{tg} 2\alpha_{\max} = \frac{2J_{xyc}}{J_{yc} - J_{xc}}. \quad (4.6)$$

Угол $\alpha_{\max} < 0$ откладываем по часовой стрелке от положительного направления оси x . Относительно оси v момент инерции фигуры максимальный.

8. Выполняем проверку. Находим тригонометрические функции:

$$\begin{aligned} \sin^2 \alpha_{\max} &= \operatorname{tg}^2 \alpha_{\max} / (1 + \operatorname{tg}^2 \alpha_{\max}) = 0,99, \\ \cos^2 \alpha_{\max} &= 1 / (1 + \operatorname{tg}^2 \alpha_{\max}) = 0,01, \\ \sin 2\alpha_{\max} &= 2 \operatorname{tg} \alpha_{\max} / (1 + \operatorname{tg}^2 \alpha_{\max}) = -0,20. \end{aligned}$$

Проверяем соотношение:

$$\begin{aligned} J_{xc} \cos^2 \alpha_{\max} + J_{yc} \sin^2 \alpha_{\max} - J_{xyc} \sin 2\alpha_{\max} &= J_{\max}, \\ J_{xc} \sin^2 \alpha_{\max} + J_{yc} \cos^2 \alpha_{\max} + J_{xyc} \sin 2\alpha_{\max} &= J_{\min}. \end{aligned} \quad (4.7)$$

Имеем

$$\begin{aligned} J_{\max} &= 183,06 \cdot 0,01 + 1425,54 \cdot 0,99 + 124,21 \cdot 0,20 = 1437,84 \text{ см}^4, \\ J_{\min} &= 183,06 \cdot 0,99 + 1425,54 \cdot 0,01 - 124,21 \cdot 0,20 = 170,76 \text{ см}^4. \end{aligned}$$

Такие же значения получены в п. 5. Проверка выполнена.

Замечание 1. В качестве проверки можно сначала вычислить моменты инерции всей фигуры относительно заданных осей:

$$\begin{aligned} J_x &= \sum_i (J_{x_i} + y_i^2 F_i), \quad J_y = \sum_i (J_{y_i} + x_i^2 F_i), \\ J_{xy} &= \sum_i (J_{xy_i} + x_i y_i F_i), \end{aligned}$$

а затем уже пересчитать результаты относительно центральных осей. Центральные моменты инерции имеют вид

$$J_{xc} = J_x - y_c^2 F, \quad J_{yc} = J_y - x_c^2 F, \quad J_{xyc} = J_{xy} - x_c y_c F. \quad (4.8)$$

Замечание 2. В тех случаях, когда для уголков в сортаменте не указан центробежный момент инерции J_{xy} , его легко вычислить по формуле (4.4), из которой в случае *равнополочных* уголков $J_{xc} = J_{yc}$ следует $J_{xy} = J_{\max} - J_{xc}$.

Задача 48. Найти максимальный и минимальный моменты инерции плоской фигуры (рис. 177) и угол наклона главной оси инерции к оси x . Размеры даны в сантиметрах.

Решение

1. Вводим систему координат. Разбиваем фигуру на пять частей: один квадрат размером 7×7 см (фигура № 1) и четыре вырезанных из него фигуры — два треугольника (№ 2, № 3), прямоугольник № 4, и четверть круга № 5 радиусом $R = 2$ см (рис. 178)¹. Определяем координаты центров тяжести фигур:

$$\begin{aligned}x_1 &= 7/2 = 3,5 \text{ см}, & y_1 &= 7/2 = 3,5 \text{ см}, \\x_2 &= 4/3 = 1,33 \text{ см}, & y_2 &= 2/3 = 0,67 \text{ см}, \\x_3 &= 5 + 4/3 = 6,33 \text{ см}, & y_3 &= (2/3) \cdot 4 = 2,67 \text{ см}, \\x_4 &= 5/2 = 2,5 \text{ см}, & y_4 &= 6 + 0,5 = 6,5 \text{ см}, \\x_5 &= 4R/(3\pi) = 0,85 \text{ см}, & y_5 &= 4 + 4R/(3\pi) = 4,85 \text{ см}.\end{aligned}$$

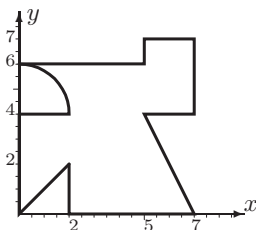


Рис. 177

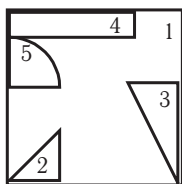


Рис. 178

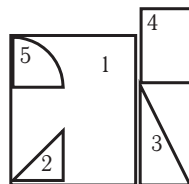


Рис. 179

2. Вычисляем площадь фигуры. Вычисляем площади составляющих, приписывая знак минус вырезанным частям: $F_1 = 7 \cdot 7 = 49 \text{ см}^2$, $F_2 = -(1/2) \cdot 2 \cdot 2 = -2 \text{ см}^2$, $F_3 = -(1/2) \cdot 4 \cdot 2 = -4 \text{ см}^2$, $F_4 = -5 \cdot 1 = -5 \text{ см}^2$, $F_5 = -\pi R^2/4 = -3,14 \text{ см}^2$.

В сумме имеем $F = \sum_i F_i = 49 - 2 - 4 - 5 - 3,14 = 34,86 \text{ см}^2$.

3. Определяем координаты центра тяжести фигуры:

$$x_c = \frac{49 \cdot 3,5 - 2 \cdot 1,33 - 4 \cdot 6,33 - 5 \cdot 2,5 - 3,14 \cdot 0,85}{34,86} = 3,68 \text{ см},$$

$$y_c = \frac{49 \cdot 3,5 - 2 \cdot 0,67 - 4 \cdot 2,67 - 5 \cdot 6,5 - 3,14 \cdot 4,85}{34,86} = 3,2 \text{ см}.$$

¹Возможно и другое разбиение на части (рис. 179). Здесь из прямоугольника № 1 вырезаны четверть круга № 5 и треугольник № 2. К полученной фигуре добавлен треугольник № 3 и прямоугольник № 4.

4. Определяем осевые J_{xc} , J_{yc} и центробежный J_{xyc} моменты инерции сечения относительно центральных осей.

Сначала записываем моменты инерции составляющих фигур относительно их центральных осей. Моменты инерции прямоугольника № 1 (квадрат $h = b = 7$ см) имеют вид (табл. 7, с. 485)

$$J_{xc1} = J_{yc1} = bh^3/12 = 70 \cdot 70^3/12 = 200,08 \text{ см}^4, \quad J_{xy1} = 0.$$

Осевые моменты инерции относительно собственных центральных осей прямоугольного равнобедренного треугольника № 2 ($h = b = 2$ см) вычисляем по формулам из таблицы 8 (с. 486), не забывая приписать знак минус, как вырезанной фигуре:

$$J_{xc2} = -bh^3/36 = -2 \cdot 2^3/36 = -0,44 \text{ см}^4, \quad J_{yc2} = -hb^3/36 = -0,44 \text{ см}^4.$$

Знак центробежного момента инерции прямоугольного треугольника зависит от его ориентации (рис. 180). Если внешняя нормаль к гипотенузе направлена в 1-ю или 3-ю четверть декартовых координат

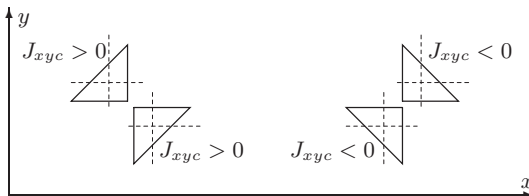


Рис. 180

(в правой системе координат), то центробежный момент инерции меньше нуля, как в табл. 8, с. 486: $J_{xy} = -b^2h^2/72$. В случае, если нормаль направлена во 2-ю или 4-ю четверть — знак плюс: $J_{xy} = b^2h^2/72$. В нашем случае у треугольника № 2 нормаль направлена во 2-ю четверть, но так как он вырезан, меняем знак момента инерции и получаем $J_{xy2} = -b^2h^2/72 = -0,22 \text{ см}^4$.

Рассуждая аналогично, получаем моменты инерции вырезанного прямоугольного треугольника № 3 ($h = 4$ см, $b = 2$ см):

$$J_{xc3} = -bh^3/36 = -2 \cdot 4^3/36 = -3,56 \text{ см}^4,$$

$$J_{yc3} = -hb^3/36 = -4 \cdot 2^3/36 = -0,89 \text{ см}^4,$$

$$J_{xy3} = b^2h^2/72 = 0,89 \text{ см}^4.$$

Моменты инерции прямоугольника № 4 ($h = 1$ см, $b = 5$ см) равны:

$$J_{xc4} = bh^3/12 = 5 \cdot 1^3/12 = 0,42 \text{ см}^4,$$

$$J_{yc4} = hb^3/12 = 1 \cdot 5^3/12 = 10,42 \text{ см}^4, \quad J_{xy4} = 0.$$

Сектор (четверть круга) № 5 радиусом $R = 2$ см имеет следующие моменты инерции относительно собственных центральных осей (табл. 8, с. 486):

$$J_{xc5} = J_{yc5} = -R^4 \left(\frac{\pi}{16} - \frac{4}{9\pi} \right) = -2^4 \cdot 0,0548 = -0,88 \text{ см}^4,$$

$$J_{xy5} = -R^4 \left(\frac{1}{8} - \frac{4}{9\pi} \right) = 2^4 \cdot 0,0165 \text{ см}^4 = 0,26 \text{ см}^4.$$

Моменты инерции всей фигуры относительно центральных осей получаем по формулам (4.3):

$$\begin{aligned} J_{xc} = & 200,08 + 49(3,5 - 3,2)^2 - 0,44 - 2(0,67 - 3,2)^2 - \\ & - 3,56 - 4(2,67 - 3,2)^2 - 0,42 - 5(6,5 - 3,2)^2 - 0,88 - \\ & - 3,14(4,85 - 3,2)^2 = 122,23 \text{ см}^4, \end{aligned} \quad (4.9)$$

$$\begin{aligned} J_{yc} = & 200,08 + 49(3,5 - 3,68)^2 - 0,44 - 2(1,33 - 3,68)^2 - \\ & - 0,89 - 4(6,33 - 3,68)^2 - 10,42 - 5(2,5 - 3,68)^2 - \\ & - 0,88 - 3,14(0,85 - 3,68)^2 = 117,72 \text{ см}^4, \end{aligned} \quad (4.10)$$

$$\begin{aligned} J_{xyc} = & 49(3,2 - 3,5)(3,68 - 3,5) - 0,22 - \\ & - 2(3,2 - 0,67)(3,68 - 1,33) + \\ & + 0,89 - 4(3,2 - 2,67)(3,68 - 6,33) - \\ & - 5(3,2 - 6,5)(3,68 - 2,5) + 0,27 - \\ & - 3,14(3,2 - 4,85)(3,68 - 0,85) = 26,2 \text{ см}^4. \end{aligned} \quad (4.11)$$

5. Находим главные моменты инерции:

$$\begin{aligned} J_{\max} = & \frac{117,72 + 122,23}{2} + \sqrt{\frac{(117,72 - 122,23)^2}{4} + 26,2^2} = \\ = & 119,98 + 26,3 = 146,27 \text{ см}^4, \end{aligned} \quad (4.12)$$

$$J_{\min} = 119,98 - 26,3 = 93,69 \text{ см}^4.$$

6. Вычисляем главные радиусы инерции:

$$i_{\max} = \sqrt{J_{\max}/F} = \sqrt{146,27/34,86} = 2,05 \text{ см},$$

$$i_{\min} = \sqrt{J_{\min}/F} = \sqrt{93,69/34,86} = 1,64 \text{ см}.$$

7. Находим направление главных осей $\text{tg } \alpha_{\max} = J_{xyc}/(J_{yc} - J_{\max}) = 26,2/(117,72 - 146,27) = -0,92$. Ось v (рис. 181), относительно которой момент инерции фигуры максимальный, направлен под углом $\alpha_{\max} = -42,54^\circ$ к оси x . Для проверки угол можно вычислить по формуле (4.6).

8. Выполняем проверку. Находим тригонометрические функции $\cos \alpha_{\max} = 0,74$, $\sin \alpha_{\max} = -0,68$, $\sin 2\alpha_{\max} = -0,99$. Проверяем соотношение (4.7). Имеем

$$J_{\max} = 122,23 \cdot 0,74^2 + 117,72 \cdot 0,68^2 + 26,2 \cdot 0,99 = 146,27 \text{ см}^4,$$

$$J_{\min} = 122,23 \cdot 0,68^2 + 117,72 \cdot 0,74^2 - 26,2 \cdot 0,99 = 93,69 \text{ см}^4.$$

Эти же значения получены в п. 5. Проверка выполнена.

Замечание. Эллипс инерции обладает свойством, позволяющим

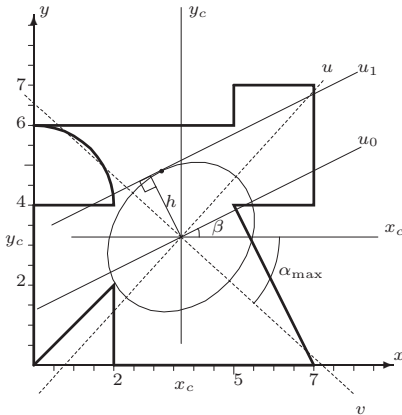


Рис. 181

получать моменты фигуры инерции относительно центральных осей, наклоненных под произвольным углом. Для того чтобы найти момент инерции фигуры относительно центральной оси u_0 , наклоненной под произвольным углом β к оси x , достаточно провести касательную u_1 к эллипсу параллельно оси u_0 и найти расстояние h между u_0 и u_1 (рис. 181). Это расстояние является радиусом инерции фигуры относительно u_0 . Таким образом, $J_{u_0} = h^2 F$.

Другое аналитическое выражение для этого момента инерции имеет вид $J_{u_0} = J_{x_c} \cos^2 \beta - J_{x_c} \sin 2\beta + J_{y_c} \sin^2 \beta$. Формула для угла наклона главных осей (4.6) легко получается из условия минимума этого выражения.

Решение этой задачи в Maple по формуле Грина¹ см. на с. 395.

4.2. Кручение вала

Закон Гука $\tau = G\gamma$ в простейших задачах кручения вала связывает касательные напряжения τ и деформации γ , где $G = E/(2(1 + \nu))$, E — модуль упругости (модуль Юнга), G — модуль сдвига, ν — коэффициент Пуассона². Пусть к валу приложен крутящий момент M . Относительный (или погонный) угол закручивания вычисляем по формуле $\theta = M/(GJ_k)$, где J_k — момент инерции поперечного сечения при кручении. Для круглого сечения диаметром d он равен полярному:

¹George Green (1793–1841) — английский математик, физик.

²Simeon Denis Poisson (1781–1840) — французский физик, математик.

$J_k = J_p = \pi d^4 / 32$. Максимальное касательное напряжение возникает на боковой поверхности вала и равно $\tau_{\max} = M/W_k$, где W_k — момент сопротивления при кручении. Для вала круглого сечения $W_k = 2J_k/d = \pi d^3 / 16$.

Задача 49. К стальному валу кусочно-постоянного круглого сечения приложены моменты $M_1 = 30$ кНм, $M_2 = 10$ кНм, $M_3 = 20$ кНм. Третий (правый) участок вала имеет кольцевое сечение. Построить

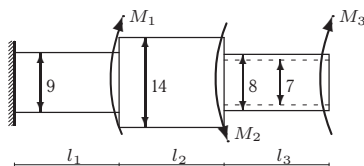


Рис. 182

эпюры углов закручивания и касательных напряжений на поверхности вала. Модуль сдвига материала $G = 80$ ГПа. Внешние диаметры вала и внутренний диаметр даны в сантиметрах. Даны длины: $l_1 = 1,1$ м, $l_2 = 1,2$ м, $l_3 = 0,8$ м.

Решение

1. Методом сечений, из условия равновесия внешних моментов и реакций опор находим моменты на участках 1, 2, 3. Для того чтобы не вычислять реакцию заделки на левом конце вала, рассматриваем равновесие правой отсеченной части (рис. 183).

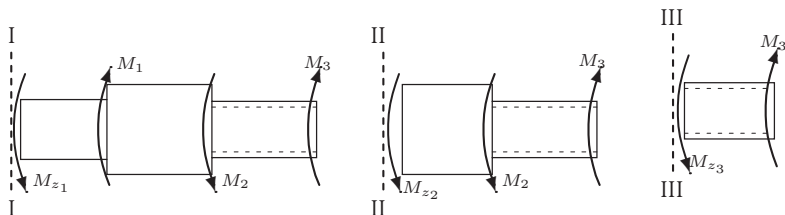


Рис. 183

Получаем $M_{z_1} = M_3 - M_2 + M_1 = 20 - 10 + 30 = 40$ кНм, $M_{z_2} = M_3 - M_2 = 20 - 10 = 10$ кНм, $M_{z_3} = M_3 = 20$ кНм.

2. Вычисляем полярные моменты инерции (4.1) сечений вала J_{pk} .

$$J_{p_1} = 0,09^4 \pi / 32 = 0,644 \cdot 10^{-5} \text{ м}^4,$$

$$J_{p_2} = 0,14^4 \pi / 32 = 3,771 \cdot 10^{-5} \text{ м}^4,$$

$$J_{p_3} = (0,08^4 - 0,07^4) \pi / 32 = 0,166 \cdot 10^{-5} \text{ м}^4.$$

3. На каждом участке определяем относительный угол закручивания $\theta_k = M_k / (GJ_{pk})$ и угол закручивания правого сечения относитель-

но левого $\tilde{\varphi}_1 = \theta_k l_k$. Учитывая значение модуля сдвига $G = 80 \text{ ГПа} = 8 \times 10^{10} \text{ Па}$, получаем (в радианах):

$$\tilde{\varphi}_1 = 40000 \cdot 1,1 / (8 \cdot 10^{10} \cdot 0,644 \cdot 10^{-5}) = 0,0640,$$

$$\tilde{\varphi}_2 = 10000 \cdot 1,2 / (8 \cdot 10^{10} \cdot 3,771 \cdot 10^{-5}) = 0,0039,$$

$$\tilde{\varphi}_3 = 20000 \cdot 0,8 / (8 \cdot 10^{10} \cdot 0,166 \cdot 10^{-5}) = 0,1202.$$

4. Вычисляем абсолютные углы закручивания:

$$\varphi_1 = \tilde{\varphi}_1 = 0,0640,$$

$$\varphi_2 = \tilde{\varphi}_1 + \tilde{\varphi}_2 = 0,0680,$$

$$\varphi_3 = \tilde{\varphi}_1 + \tilde{\varphi}_2 + \tilde{\varphi}_3 = 0,1882.$$

Строим эпюру моментов, соединяя значения полученных ординат отрезками прямых (рис. 184).

5. Вычисляем касательные напряжения на участках по формуле

$$\tau_k = M_{kz} / W_{pk}, \quad (4.13)$$

где $W_{pk} = 2J_{pk}/d_k$ — полярный момент сопротивления или момент сопротивления при кручении. Получаем (МПа)

$$\tau_1 = 40000 \cdot 0,09 / (2 \cdot 0,644 \cdot 10^{-5} \cdot 10^6) = 279,45,$$

$$\tau_2 = 10000 \cdot 0,14 / (2 \cdot 3,771 \cdot 10^{-5} \cdot 10^6) = 18,56,$$

$$\tau_3 = 20000 \cdot 0,08 / (2 \cdot 0,166 \cdot 10^{-5} \cdot 10^6) = 480,75.$$

Строим эпюру максимальных касательных напряжений (рис. 185).

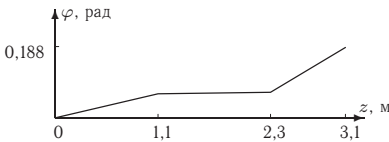


Рис. 184

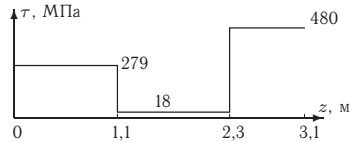


Рис. 185

Замечание 1. В условиях некоторых задач на эту тему задается мощность N , передаваемая валу, и его угловая скорость ω . По этим значениям для установившегося процесса ($M = \text{const}$) вычисляем момент: $M = N/\omega$. В общем случае мощность есть скорость изменения работы $N = dA/dt$, где $A = \int M d\varphi$.

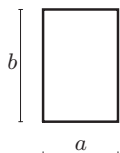


Рис. 186

Замечание 2. Для прямоугольного сечения (рис. 186, $b > a$) формулы для расчета относительного угла закручивания и касательного напряжения имеют вид

$$\theta = \frac{M}{GJ_k}, \quad J_k = \beta ba^3, \quad \tau_{\max} = \frac{M}{W_k}, \quad (4.14)$$

где $W_k = \alpha ba^2$ — момент сопротивления при кручении. Коэффициенты α , β зависят от соотношения сторон сечения и получаются в результате решения задачи упругости. Для узких прямоугольников, $b \gg a$, с большой степенью точности можно принять $\alpha = \beta = 1/3$. Соответствующая программа **Maple** приведена на с. 412. Результаты решения записаны в таблицу 6 на с. 413.

Максимальное касательное напряжение наблюдается в середине большей стороны прямоугольника. Напряжения в середине короткой стороны имеют вид $\tau = \gamma \tau_{\max}$.

Maple–программа расчета на кручение вала с круглым сечением приведена на с. 407.

Задача 50. Найти моменты в заделках опор вала кусочно-постоянного круглого сечения (рис. 187). К валу приложены моменты $M_1 = 10$ кНм, $M_2 = 18$ кНм, $M_3 = 15$ кНм. Построить эпюры углов

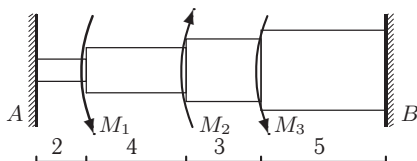


Рис. 187

закручивания. Модуль сдвига материала $G = 80$ ГПа. Длины участков даны в метрах. Диаметры участков: $d_1 = 0,09$ м, $d_2 = 0,18$ м, $d_3 = 0,25$ м, $d_4 = 0,32$ м.

Решение

Задача статически неопределимая. Для двух неизвестных реакций в опорах имеется только одно статическое уравнение равновесия: сумма всех моментов, приложенных к валу, включая реакции опор, равна нулю. Поэтому для решения требуется привлечь еще какое-то условие. Этим условием является отсутствие угла поворота в заделке.

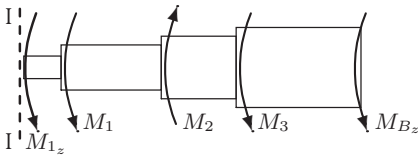


Рис. 188

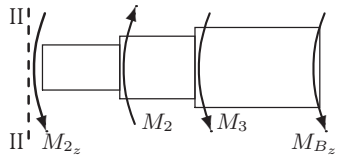


Рис. 189

1. Методом сечений, из условия равновесия внешних моментов и реакций опор находим моменты на участках 1, 2, 3, 4. Рассматриваем равновесие правой отсеченной части (рис. 188–191.)

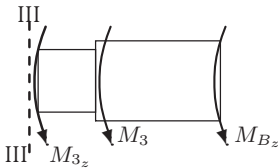


Рис. 190

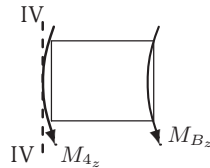


Рис. 191

Получаем

$$\begin{aligned}
 M_{z_1} &= -M_1 + M_2 - M_3 - M_{B_z} = -7 - M_{B_z}, \\
 M_{z_2} &= M_2 - M_3 - M_{B_z} = 3 - M_{B_z}, \\
 M_{z_3} &= -M_3 - M_{B_z} = -15 - M_{B_z}, \\
 M_{z_4} &= -M_{B_z}.
 \end{aligned}
 \tag{4.15}$$

2. Вычисляем полярные моменты инерции сечений вала J_p :

$$\begin{aligned}
 J_{p_1} &= \pi 0,09^4 / 32 = 0,644 \cdot 10^{-5} \text{ м}^4, \\
 J_{p_2} &= \pi 0,18^4 / 32 = 10,306 \cdot 10^{-5} \text{ м}^4, \\
 J_{p_3} &= \pi 0,25^4 / 32 = 38,349 \cdot 10^{-5} \text{ м}^4, \\
 J_{p_4} &= \pi 0,32^4 / 32 = 102,943 \cdot 10^{-5} \text{ м}^4.
 \end{aligned}$$

3. На каждом участке определяем угол закручивания правого сечения относительно левого:

$$\begin{aligned}
 \tilde{\varphi}_1 &= -(7 + M_{B_z}) \cdot l_1 / (GJ_{p_1}), \\
 \tilde{\varphi}_2 &= (3 - M_{B_z}) \cdot l_2 / (GJ_{p_2}), \\
 \tilde{\varphi}_3 &= -(15 + M_{B_z}) \cdot l_3 / (GJ_{p_3}), \\
 \tilde{\varphi}_4 &= -M_{B_z} \cdot l_4 / (GJ_{p_4}).
 \end{aligned}
 \tag{4.16}$$

4. Из уравнения $\varphi_1 = \tilde{\varphi}_1 + \tilde{\varphi}_2 + \tilde{\varphi}_3 + \tilde{\varphi}_4 = 0$ определяем неизвестную реакцию в опоре B . Сокращая модуль сдвига G , получаем

$$-(7 + M_{B_z})l_1/J_{p_1} + (3 - M_{B_z})l_2/J_{p_2} - (15 + M_{B_z})l_3/J_{p_3} - M_{B_z}l_4/J_{p_4} = 0.$$

Подставляем значения длин и моментов инерции:

$$-21,744 - 3,619M_{B_z} = 0.$$

Получаем момент в правой заделке: $M_{B_z} = -6,006$ кНм. Статическая неопределимость раскрыта.

5. Вычисляем абсолютные углы закручивания:

$$\begin{aligned}\varphi_4 &= \tilde{\varphi}_4, \\ \varphi_3 &= \tilde{\varphi}_3 + \tilde{\varphi}_4, \\ \varphi_2 &= \tilde{\varphi}_2 + \tilde{\varphi}_3 + \tilde{\varphi}_4, \\ \varphi_1 &= \tilde{\varphi}_1 + \tilde{\varphi}_2 + \tilde{\varphi}_3 + \tilde{\varphi}_4.\end{aligned}\tag{4.17}$$

6. С учетом найденной реакции по формулам (4.15) вычисляем моменты на участках (кНм):

$$\begin{aligned}M_{z_1} &= -7 + 6,007 = -0,993, \\ M_{z_2} &= 3 + 6,007 = 9,007, \\ M_{z_3} &= -15 + 6,007 = -8,993, \\ M_{z_4} &= 6,007.\end{aligned}$$

7. Определяем абсолютные углы закручивания (в радианах), подставляя найденную реакцию в выражения (4.16) и (4.17). Сначала вычисляем относительные углы поворота сечений:

$$\begin{aligned}\tilde{\varphi}_1 &= -0,993 \cdot 2 / (8 \cdot 10^{10} \cdot 0,644 \cdot 10^{-5}) = -3,845 \cdot 10^{-3}, \\ \tilde{\varphi}_2 &= 9,007 \cdot 4 / (8 \cdot 10^{10} \cdot 10,306 \cdot 10^{-5}) = 4,370 \cdot 10^{-3}, \\ \tilde{\varphi}_3 &= -8,993 \cdot 3 / (8 \cdot 10^{10} \cdot 38,349 \cdot 10^{-5}) = -0,879 \cdot 10^{-3}, \\ \tilde{\varphi}_4 &= 6,007 \cdot 5 / (8 \cdot 10^{10} \cdot 102,943 \cdot 10^{-5}) = 0,365 \cdot 10^{-3},\end{aligned}$$

затем абсолютные:

$$\begin{aligned}\varphi_4 &= 0,365 \cdot 10^{-3}, \\ \varphi_3 &= (-0,879 + 0,365) \cdot 10^{-3} = -0,515 \cdot 10^{-3}, \\ \varphi_2 &= (4,370 - 0,879 + 0,365) \cdot 10^{-3} = 3,855 \cdot 10^{-3}, \\ \varphi_1 &= (-3,845 + 4,370 - 0,879 + 0,365) \cdot 10^{-3} = -0,1 \cdot 10^{-7}.\end{aligned}$$

По значениям φ_4 , φ_3 и φ_2 строим эпюру углов поворота (рис. 192). Вычисление φ_1 является простой проверкой расчетов. По условию угол поворота сечения в левой заделке должен быть равным нулю. Кроме того, $\varphi_5 = 0$.

8. Вычисляем касательные напряжения на участках по формуле (4.13):

$$\tau_1 = -993 \cdot 0,09 / (0,644 \cdot 10^{-5} \cdot 2 \cdot 10^6) = -6,939 \text{ МПа},$$

$$\tau_2 = 9007 \cdot 0,18 / (10,306 \cdot 10^{-5} \cdot 2 \cdot 10^6) = 7,865 \text{ МПа},$$

$$\tau_3 = -8993 \cdot 0,25 / (38,349 \cdot 10^{-5} \cdot 2 \cdot 10^6) = -2,931 \text{ МПа},$$

$$\tau_4 = -6007 \cdot 0,32 / (102,943 \cdot 10^{-5} \cdot 2 \cdot 10^6) = 0,934 \text{ МПа}.$$

Строим эпюру максимальных касательных напряжений (рис. 193). В точках приложения сосредоточенных моментов эпюра моментов имеет разрыв (скачок), а эпюра углов поворота — излом.

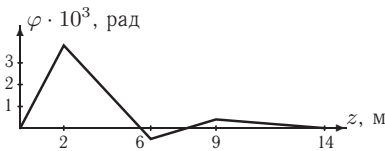


Рис. 192

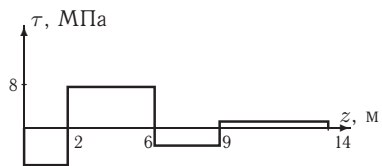


Рис. 193

Maple-программа решения статически неопределимой задачи на кручение приведена на с. 408.

4.3. Секториальные характеристики сечения

В задачах кручения различают свободное и стесненное кручение стержней. В первом случае концы стержня не закреплены, и к ним прикладывают моменты, во втором случае один из концов или оба закреплены. Ограничения на осевые перемещения точек сечения вызывают дополнительные эффекты, учесть которые можно, привлекая такие геометрические характеристики, как секториальные моменты инерции.

При свободном кручении стержней сечения остаются плоскими, поворачиваясь одно относительно другого. При стесненном кручении тонкостенных стержней сечения искривляются — возникает деформация, переменная по длине стержня. Это приводит к дополнительным касательным и нормальным напряжениям в сечении. В решении задачи о стесненном кручении стержней, принадлежащее В. З. Власову¹, помимо уже изученных моментов инерций (с. 127), входят новые геометрические характеристики сечения, определение которых составляет содержание этого раздела.

¹Василий Захарович Власов (1906–1958) — выдающийся советский ученый, чл.-корр. АН СССР, автор технической теории тонкостенных оболочек.

Основной новой характеристикой сечения является понятие секториальной площади точки кривой.

Секториальная площадь точки кривой C относительно точки B с полюсом A определяется как криволинейный интеграл:

$$\omega_{B,C}^A = - \left[\int_B^C \vec{r} \times d\vec{r} \right]_z = \int_B^C (y - y_A) dx - (x - x_A) dy,$$

и численно равна удвоенной площади, ометаемой радиусом \vec{AP} , где P — точка на кривой, при движении точки P от B к C , взятой со знаком плюс, если поворот вектора происходит по часовой стрелке, и с минусом — в противном случае. Если кривая является отрезком прямой

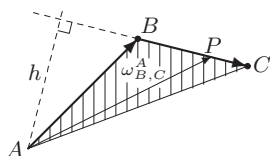


Рис. 194

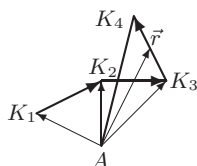


Рис. 195

BC (рис. 194), то секториальная площадь $\omega_{B,C}^A$ точки кривой C относительно точки B с полюсом A есть проекция вектора $\vec{\Omega} = \vec{AB} \times \vec{BC}$ на ось z со знаком минус: $\omega_{B,C}^A = -\Omega_z$. Если кривая представляет собой ломаную K_1, \dots, K_N , то площадь определяется суммой $\omega_{N,1}^A = \sum_i^{N-1} \omega_{i,i+1}^A$, где на каждом отрезке определяется секториальная площадь конца отрезка K_{i+1} относительно его начала K_i : $\omega_{i,i+1}^A$. В частности, для отрезка $K_1K_2K_3K_4$ (рис. 195) имеем $\omega_{K_4,K_1}^A = 2F_{\Delta AK_1K_2} + 2F_{\Delta AK_2K_3} - 2F_{\Delta AK_3K_4}$. Удвоенная площадь треугольника AK_3K_4 взята с минусом, так как поворот вектора \vec{r} при движении по отрезку от K_3 к точке K_4 происходит против часовой стрелки.

План решения

1. Определяем площадь сечения как сумму площадей составляющих прямоугольников¹

$$F = \sum_{i=1}^n l_i \delta_i. \quad (4.18)$$

Введены обозначения: l_i — длина i -го участка, δ_i — его толщина.

¹Скруглениями на стыках прямоугольников, существующими в реальных конструкциях, пренебрегаем.

2. Определяем координаты центра тяжести сечения. Вводим произвольную систему координат. Если сечение имеет ось симметрии, то начало координат лучше выбирать на ней. Строим эпюры координат точек сечений. Координаты центра тяжести определяем по формулам

$$x_c = \frac{1}{F} \int_F y dF, \quad y_c = \frac{1}{F} \int_F x dF, \quad (4.19)$$

где интегрирование выполняем по правилу Верещагина¹. Согласно правилу Верещагина, интеграл от произведения эпюр $m(x)$ и $M(x)$ на участке от l_1 до l_2 равен произведению двух чисел:

$$\int_{l_1}^{l_2} m(x)M(x)dx = \Omega h,$$

одно из которых Ω равно площади криволинейной эпюры, второе h — ордината линейной, взятая под центром тяжести криволинейной. Если обе эпюры линейные, то за криволинейную можно взять любую из них.

3. Определяем центральные моменты инерции сечения. Сначала моменты инерции относительно выбранных осей определяем по формулам

$$J_x = \int_F y^2 dF, \quad J_y = \int_F x^2 dF, \quad (4.20)$$

где интегрирование также выполняем по правилу Верещагина. Находим центральные моменты инерции сечения по формулам (4.8), с. 130.

4. Строим эпюру секториальных площадей ω_B сечения при произвольном полюсе B и произвольной начальной точке².

¹Безусловно, самый очевидный путь решения задачи — непосредственное интегрирование, тем более что интеграл табличный. Однако в процессе решения иногда приходится выполнять это простое действие так много раз, что напрашивается желание как-то ускорить эту многократно повторяющуюся операцию. Правило Верещагина — один из путей упрощения решения (без потери точности). Другой путь — формула Симпсона. Она имеет вид $\int_{l_1}^{l_2} m(x)M(x)dx = (l/6)(aA + 4cC + bB)$, где a, b — ординаты на концах линейной эпюры, A, B — ординаты на концах криволинейной эпюры, c и C — ординаты эпюр, вычисленные в середине участка длиной l .

²Начальная точка принадлежит срединной кривой сечения, а полюс может располагаться произвольно. Для симметричного профиля полюс и начальную точку можно расположить на оси симметрии и даже совместить.

5. Оси координат совмещаем с главными центральными осями сечения¹. Вычисляем секториально-линейные моменты:

$$S_{x\omega_B} = \int_F y\omega dF, \quad S_{y\omega_B} = \int_F x\omega dF. \quad (4.21)$$

Интегрирование выполняем по правилу Верещагина.

6. Определяем координаты центра изгиба A :

$$\alpha_x = \frac{-S_{x\omega_A}J_{y_c} + S_{y\omega_A}J_{xyc}}{J_{x_c}J_{y_c} - J_{xyc}^2}, \quad \alpha_y = \frac{S_{y\omega_A}J_{x_c} - S_{x\omega_A}J_{xyc}}{J_{x_c}J_{y_c} - J_{xyc}^2}.$$

Для симметричных сечений $J_{xyc} = 0$ и формулы получаются проще

$$\alpha_x = -S_{x\omega_B}/J_{x_c}, \quad \alpha_y = S_{y\omega_B}/J_{y_c}.$$

7. Строим эпюру секториальных площадей ω_A сечения при полюсе в центре изгиба и произвольной начальной точке.

8. Вычисляем секториально-статический момент площади

$$S_{\omega_A} = \int_F \omega dF.$$

Определяем константу D , задающую главные секториальные площади сечения,

$$D = S_{\omega_A}/F.$$

9. Строим эпюру главных секториальных площадей

$$\omega = \omega_A - D.$$

10. Вычисляем секториальный момент инерции (бимомент инерции) сечения

$$J_{\omega} = \int_F \omega^2 dF.$$

Для проверки можно вычислить секториально-статический момент площади профиля S_{ω} . Если полюс находится в центре изгиба, то он должен быть равен нулю.

Задача 51. Найти геометрические характеристики симметричного сечения тонкостенного стержня открытого профиля (рис. 196).

¹Для симметричного сечения эти оси проходят через центр тяжести и направлены по осям (оси) симметрии.

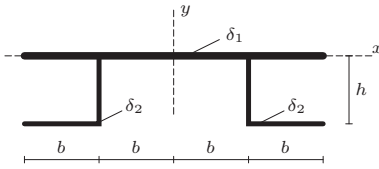


Рис. 196

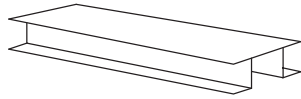


Рис. 197

Размеры относятся к средним линиям сечения и указаны в сантиметрах. Стержень состоит из пластины толщиной $\delta_1 = 1,5$ см и двух уголков $b \times h$ толщиной $\delta_2 = 1$ см (рис. 197). Дано: $b = 23$ см, $h = = 21$ см.

Решение

1. Определяем площадь сечения как сумму площадей пяти составляющих прямоугольников: $F = \sum_{i=1}^5 l_i \delta_i = 4b\delta_1 + 2(b+h)\delta_2 = 226 \text{ см}^2$.

2. Определяем координаты центра тяжести сечения. Вводим систему координат с началом в точке B (рис. 196). Строим эпюры координат точек сечений (рис. 198, 199).

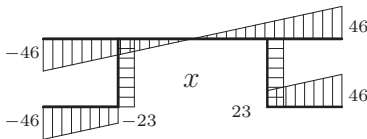


Рис. 198

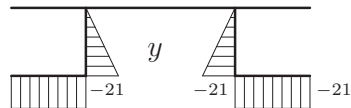


Рис. 199

Координаты центра тяжести определяем по формулам (4.19). Интегрирование заменяем суммированием. Замечая, что на каждом i -м участке $dF_i = \delta_i ds$, где δ_i — толщина сечения на участке, ds — дифференциал дуги, получаем

$$y_c = \frac{1}{F} \sum_i \int_{l_i} y dF = \frac{1}{F} \sum_i \delta_i \int_{l_i} y ds = \frac{\sum_i \delta_i Y_i}{F},$$

где Y_i — площадь эпюры y на i -м участке. Сечение симметрично относительно оси y , отсюда $x_c = 0$. Пользуясь эпюрой y на рис. 199, вычисляем: $y_c = -2\delta_2((1/2) \cdot 21 \cdot 21 + 21 \cdot 23)/226 = -6,23$ см.

3. Определяем главные моменты инерции сечения. Моменты инерции относительно выбранных осей определяем по формулам (4.20). Интегрирование выполняем по правилу Верещагина, умножая площадь одной эпюры на ординату другой под центром тяжести первой. Пользуемся эпюрой y на рис. 199, содержащей два участка (с учетом

симметрии). Умножаем эпюру саму на себя и находим

$$J_x = 2\delta_2 \left(\frac{1}{2} \cdot 21 \cdot 21 \cdot \frac{2}{3} \cdot 21 + 21 \cdot 23 \cdot 21 \right) = 26460 \text{ см}^4.$$

Ордината выделена полужирным шрифтом, $\delta_2 = 1$ см. Заметим, что при таком способе вычисления моментов инерции не учитываются моменты инерции прямоугольных частей сечения относительно собственных центральных продольных осей¹. Аналогично, пользуясь эпюрой x на рис. 198, при $\delta_1 = 1,5$ см получаем

$$\begin{aligned} J_y &= 2 \left(\delta_1 \frac{1}{2} \cdot 46 \cdot 46 \cdot \frac{2}{3} \cdot 46 + \delta_2 \left(23 \cdot 21 \cdot 23 + \right. \right. \\ &\quad \left. \left. + \frac{1}{2} \cdot 23 \cdot 23 \left(\frac{2}{3} \cdot 23 + \frac{1}{3} \cdot 46 \right) + \frac{1}{2} \cdot 46 \cdot 23 \left(\frac{2}{3} \cdot 46 + \frac{1}{3} \cdot 23 \right) \right) \right) = \\ &= 176333,33 \text{ см}^4. \end{aligned}$$

Находим главные моменты инерции сечения

$$J_{x_c} = J_x - y_c^2 F = 26460 - 6,23^2 \cdot 226 = 17700,49 \text{ см}^4,$$

$$J_{y_c} = J_y - x_c^2 F = J_y = 176333,33 \text{ см}^4.$$

4. Строим эпюру секториальных площадей ω_B сечения. Полюс B и начальную точку располагаем на оси симметрии — в начале координат (рис. 200). Очевидно, $\omega_{B,C}^B = 0$, $\omega_{B,C'}^B = 0$, $\omega_{B,K}^B = 0$, $\omega_{B,K'}^B = 0$ ².

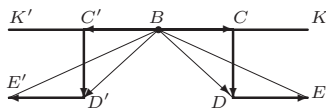


Рис. 200

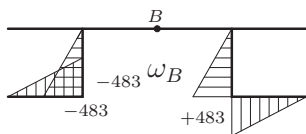


Рис. 201

¹Точное выражение для момента инерции сечения, состоящего из прямоугольников, имеет вид $J_x = 4b\delta_1^3/12 + 2b\delta_2^3/12 + 2\delta_2 h^3/3 + 2b\delta_2 h^2 = 26489,71 \text{ см}^4$. В приближенном выражении первые два слагаемые, содержащие δ^3 , не учтены. Погрешность составляет 0,1%.

²В общем, если три точки K_1 , K_2 и K_3 лежат на одной прямой, то секториальная площадь $\omega_{K_2, K_3}^{K_1}$ точки кривой K_3 относительно точки K_2 с полюсом K_1 равна нулю.

Для правой половины сечения получаем

$$\begin{aligned}\omega_{C,D}^B &= bh = 483 \text{ см}^2, \quad \omega_{E,D}^B = -bh = -483 \text{ см}^2, \\ \omega_{B,E}^B &= \omega_{B,C}^B + \omega_{C,D}^B + \omega_{E,D}^B = 0, \\ \omega_{B,D}^B &= \omega_{B,C}^B + \omega_{C,D}^B = 483 \text{ см}^2.\end{aligned}$$

Аналогично для левой части

$$\begin{aligned}\omega_{C',D'}^B &= -bh = -483 \text{ см}^2, \quad \omega_{E',D'}^B = bh = 483 \text{ см}^2, \\ \omega_{B',E'}^B &= \omega_{B,C'}^B + \omega_{C',D'}^B + \omega_{E',D'}^B = 0, \\ \omega_{B',D'}^B &= \omega_{B,C'}^B + \omega_{C',D'}^B = -483 \text{ см}^2.\end{aligned}$$

На рис. 201 изображена полученная эпюра.

5. Оси координат совмещаем с главными центральными осями сечения (рис. 202). Строим эпюру координат y в новой (центральной) системе координат (рис. 203). Все ординаты эпюры на рис. 199 получат приращение 6,23, а эпюра x не изменится, т. к. $x_c = 0$.

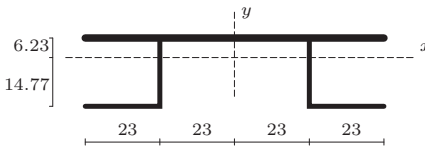


Рис. 202

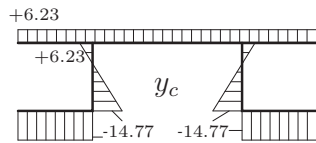


Рис. 203

Вычисляем секториально-линейный статический момент $S_{y\omega_B}$, перемножая эпюру координат x (рис. 198) и эпюру секториальных площадей ω_B (рис. 201). С учетом симметрии функции $x\omega_B$, по правилу Верещагина получаем

$$S_{y\omega_B} = 2\delta_2 \left(\frac{483}{2} \cdot 21 \cdot 23 + \frac{483}{2} \cdot 23 \cdot \left(\frac{2}{3} \cdot 23 + \frac{1}{3} \cdot 46 \right) \right) = 573965 \text{ см}^5.$$

Так как эпюра ω антисимметричная, а эпюра y симметричная (рис. 203), то $S_{x\omega_B} = \int_F y\omega dF = 0$. Эпюра координат y в новой (центральной) системе координат в данном случае не потребовалась.

6. Определяем координаты центра изгиба A : $\alpha_x = 0$, $\alpha_y = S_{y\omega_B} / J_{yc} = 573965 / 176333,33 = 3,26 \text{ см}$.

7. Строим эпюру секториальных площадей ω_A сечения при полюсе в центре изгиба A и начальной точке B (рис. 204).

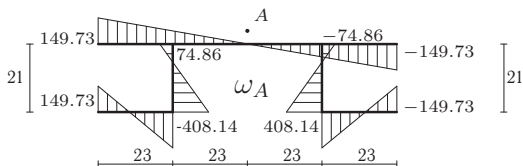


Рис. 204

8. Определяем константу D , задающую главные секториальные площади. В данном случае эпюра ω_A имеет площадь, равную нулю, так как сечение симметрично, а точка отсчета секториальных площадей (начальная точка) была выбрана на оси симметрии. Итак, $S_{\omega_A} = \int_F \omega dF = 0$ и $D = S_{\omega_A} / F = 0$.

9. Эпюра главных секториальных площадей сечения $\omega = \omega_A - D$ при $D = 0$ совпадает с эпюрой ω_A (рис. 204).

10. Вычисляем секториальный момент инерции сечения J_ω . Умножаем эпюру ω (рис. 204) саму на себя. Интеграл берем по правилу Верещагина. Учитываем симметрию эпюры (общий множитель 2), умножаем площадь эпюры на ее ординату под центром тяжести. Ординату выделяем шрифтом:

$$\begin{aligned}
 J_\omega = \int_F \omega^2 dF = & 2\delta_1 \left(\frac{149,73}{2} \cdot 46 \cdot \frac{2}{3} \cdot 149,73 \right) + \\
 & + 2\delta_2 \left(\frac{74,86}{2} \cdot 21 \cdot \left(\frac{2}{3} \cdot 74,86 - \frac{1}{3} \cdot 408,14 \right) \right) + \\
 & + \frac{408,14}{2} \cdot 21 \cdot \left(\frac{2}{3} \cdot 408,14 - \frac{1}{3} \cdot 74,86 \right) + \\
 & + \frac{408,14}{2} \cdot 23 \cdot \left(\frac{2}{3} \cdot 408,14 - \frac{1}{3} \cdot 149,73 \right) + \\
 & + \frac{149,73}{2} \cdot 23 \cdot \left(\frac{2}{3} \cdot 149,73 - \frac{1}{3} \cdot 408,14 \right) = 4974887,924 \text{ см}^6.
 \end{aligned}$$

Замечание 1. Если в п. 4 полюс и начальную точку поместить в точку C' , то получится другая эпюра секториальных площадей (рис. 205),

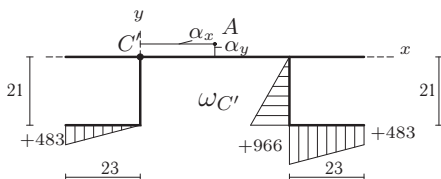


Рис. 205

однако решение не изменится, так как в п. 5 найдем

$$S_{x\omega_{C'}} = \int_F y\omega dF = \delta_2 \left(\frac{966}{2} \cdot 21 \cdot \left(-\frac{2}{3} \cdot 14,77 + \frac{1}{3} \cdot 6,23 \right) - \right. \\ \left. - \frac{966 + 483}{2} \cdot 23 \cdot 14,77 - \frac{1}{2} \cdot 483 \cdot 23 \cdot 14,77 \right) = -407111,30 \text{ см}^4.$$

Секториально-линейный статический момент относительно оси y не изменится: $S_{y\omega_{C'}} = S_{y\omega_B} = 573965 \text{ см}^5$. В п. 6 определим координаты центра изгиба в системе координат с центром в C' : $\alpha_x = -S_{x\omega_{C'}}/J_{x_c} = 407111,30/17700,49 = 23 \text{ см}$, $\alpha_y = S_{y\omega_B}/J_{y_c} = 573965/176333,33 = 3,26 \text{ см}$. Таким образом, центр изгиба опять окажется в точке A .

Замечание 2. Константа D , задающая главные секториальные площади в п. 8, в рассмотренном примере оказалась равной нулю, так как эпюра ω_A была симметричной и имела нулевую площадь. Но это бывает не всегда, и все зависит от выбора начальной точки. Возьмем для сравнения с уже полученным результатом начальную точку не на оси симметрии, а, например, в точке C' . Оставляя полюс в той же точке — в центре изгиба A , построим эпюру секториальных площадей (рис. 206).

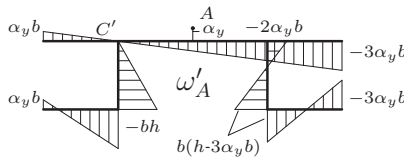


Рис. 206

Эпюра получилась несимметричной, и ее площадь отлична от нуля:

$$S_{\omega'_A} = \int_F \omega dF = -2\alpha_y(2b^2\delta_1 + b\delta_2(b+h)) = -16919,5 \text{ см}^4.$$

Найдем константу D :

$$D = \frac{S_{\omega'_A}}{F} = -\frac{2\alpha_y(2b^2\delta_1 + b\delta_2(b+h))}{2(2b\delta_1 + \delta_2(b+h))} = -b\alpha_y = -74,86 \text{ см}^2.$$

Если теперь вычесть полученную константу из ординат эпюры ω'_A на рис. 206, то получится прежняя эпюра главных секториальных площадей (рис. 204).

Задача 52. Найти геометрические характеристики сечения тонкостенного стержня открытого профиля (рис. 207). Размеры относятся к

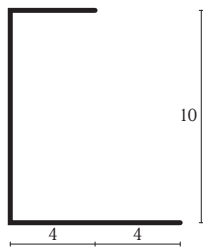


Рис. 207

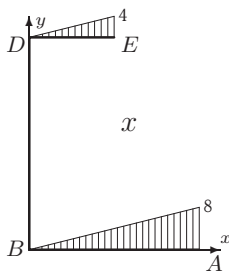


Рис. 208

средним линиям сечения и указаны в сантиметрах. Толщина стенки δ равна $0,2 \text{ см}^1$.

Решение

1. Определяем площадь сечения как сумму площадей трех прямоугольников (рис. 207):

$$F = \sum_{i=1}^3 l_i \delta = (8 + 10 + 4) \cdot 0,2 = 4,4 \text{ см}^2.$$

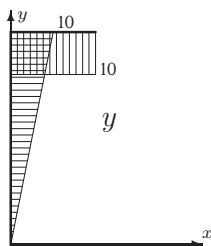


Рис. 209

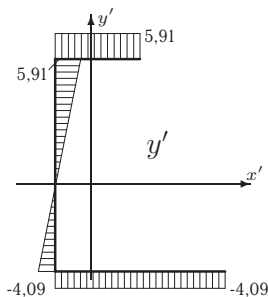


Рис. 210

2. Определяем координаты центра тяжести сечения. Вводим прямоугольную систему координат с началом в произвольной точке, например, угловой точке B (рис. 208). Строим эпюры координат точек сечений (рис. 208, 209).

¹Бояришинов С. В. Основы строительной механики машин. — М.: Машиностроение, 1973 — С. 31.

Координаты центра тяжести определяем по формулам

$$x_C = \sum_{i=1}^3 l_i x_i \delta / F = 1,82 \text{ см}, \quad y_C = \sum_{i=1}^3 l_i y_i \delta / F = 4,09 \text{ см}.$$

Этот же результат можно получить, заменив интегрирование

$$x_C = \int_{l_i} x dF / F, \quad y_C = \int_{l_i} y dF / F$$

умножением эпюр (рис. 208, 209) на единичную эпюру по правилу Верещагина:

$$x_C = \frac{\delta}{F} \left(\frac{1}{2} \cdot 4 \cdot 4 \cdot 1 + \frac{1}{2} \cdot 8 \cdot 8 \cdot 1 \right) = \frac{0,2 \cdot 40}{4,4} = 1,82 \text{ см},$$

$$y_C = \frac{\delta}{F} \left(\frac{1}{2} \cdot 10 \cdot 10 \cdot 1 + 10 \cdot 4 \cdot 1 \right) = \frac{0,2 \cdot 90}{4,4} = 4,09 \text{ см}.$$

3. Определяем главные моменты инерции сечения. Моменты инерции относительно выбранных осей xy определяем по формулам (4.20).

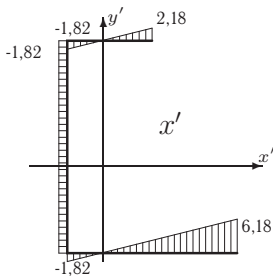


Рис. 211

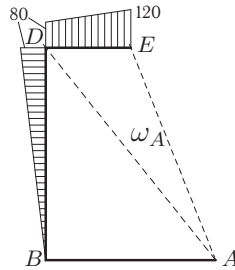


Рис. 212

Интегрирование выполняем по правилу Верещагина, умножая площадь одной эпюры на ординату другой под центром тяжести первой. В данном случае эпюры "возводим в квадрат", умножая самих на себя. Пользуясь эпюрой y на рис. 209, содержащей два участка, найдем

$$J_x = \delta \left(4 \cdot 10 \cdot 10 + \frac{1}{2} \cdot 10 \cdot 10 \cdot \frac{2}{3} \cdot 10 \right) = 146,66 \text{ см}^4.$$

Ордината выделена полужирным шрифтом, $\delta = 0,2$ см. Аналогично, пользуясь эпюрой x на рис. 208, получаем

$$J_y = \delta \left(\frac{1}{2} \cdot 4 \cdot 4 \cdot \frac{2}{3} \cdot 4 + \frac{1}{2} \cdot 8 \cdot 8 \cdot \frac{2}{3} \cdot 8 \right) = 38,4 \text{ см}^4.$$

Сечение несимметричное, поэтому центробежный момент инерции отличен от нуля. Перемножим эпюры координат на рис. 208 и 209:

$$J_{xy} = \delta \left(\frac{1}{2} \cdot 4 \cdot 4 \cdot 10 \right) = 16 \text{ см}^4.$$

Находим центральные моменты инерции сечения

$$J_{x_c} = J_x - y_c^2 F = 146,66 - 4,09^2 \cdot 4,4 = 73,03 \text{ см}^4,$$

$$J_{y_c} = J_y - x_c^2 F = 38,4 - 1,82^2 \cdot 4,4 = 23,85 \text{ см}^4,$$

$$J_{x_y_c} = J_{xy} - x_c y_c F = 16 - 4,09 \cdot 1,82 \cdot 4,4 = -16,73 \text{ см}^4.$$

Заметим, что этот же результат можно получить, перемножая (или возводя в квадрат) соответствующие эпюры координат x' , y' с началом в центре тяжести (рис. 211, 210).

4. Строим эпюру секториальных площадей ω_A сечения. Полюс и начальную точку располагаем в произвольной точке, например точке A (рис. 212). Очевидно, $\omega_{A,B}^A = 0$, $\omega_{A,D}^A = 10 \cdot 8 = 80 \text{ см}^2$, $\omega_{A,E}^A = \omega_{A,D}^A + \omega_{D,E}^A = 80 + 10 \cdot 4 = 120 \text{ см}^2$.

5. Оси координат совмещаем с центральными осями сечения. Строим эпюры координат x' , y' в новой (центральной) системе координат (рис. 211, 210). Все ординаты эпюры на рис. 208 получают приращение $-1,82$, и эпюра координат x примет вид (рис. 211). Аналогично, вычитая из всех ординат на эпюры y значение $y_c = 4,09$, получаем эпюру y' (рис. 210).

Вычисляем секториально-линейные статические моменты $S_{y'\omega_A}$, $S_{x'\omega_A}$, перемножая, соответственно, эпюры координат x' и y' (рис. 208, 209) и эпюру секториальных площадей ω_A (рис. 212). По правилу Верещагина в соответствии с формулами (4.21) получаем

$$\begin{aligned} S_{y'\omega_A} &= \int_F x' \omega_A dF = \delta \left(4 \cdot \frac{2,18}{2} \cdot \left(\frac{2}{3} \cdot 120 + \frac{1}{3} \cdot 80 \right) - \right. \\ &\quad \left. - 4 \cdot \frac{1,82}{2} \cdot \left(\frac{2}{3} \cdot 80 + \frac{1}{3} \cdot 120 \right) - 1,82 \cdot 10 \cdot \frac{80}{2} \right) = -120,533 \text{ см}^5, \\ S_{x'\omega_A} &= \int_F y' \omega_A dF = \delta \left(5,91 \cdot 4 \cdot \frac{120 + 80}{2} + 5,91 \cdot \frac{10}{2} \cdot \frac{2}{3} \cdot 80 - \right. \\ &\quad \left. - 4,91 \cdot \frac{10}{2} \cdot \frac{1}{3} \cdot 80 \right) = 678,933 \text{ см}^5. \end{aligned}$$



6. Определяем координаты центра изгиба P :

$$x_P = \alpha_x^A + x_A = \frac{-S_{x'\omega_A} J_{y_c} + S_{y'\omega_A} J_{x y_c}}{J_{x_c} J_{y_c} - J_{x y_c}^2} + x_A = \frac{-14180,848}{1462,303} + 8 = -1,698 \text{ см},$$

$$y_P = \alpha_y^A + y_A = \frac{S_{y'\omega_A} J_{x_c} - S_{x'\omega_A} J_{x y_c}}{J_{x_c} J_{y_c} - J_{x y_c}^2} + y_A = \frac{2572,929}{1462,303} = 1,759 \text{ см}.$$

Для проверки координат можно выбрать другой полюс при построении эпюр секториальных площадей, поместив его, например,

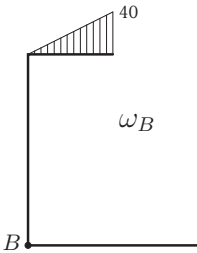


Рис. 213

в начало координат, точку B . Сюда же поместим и начальную точку. Полученная эпюра на рис. 213 называется теперь ω_B (по названию полюса) и оказывается проще — в ней больше нулевых участков, что упрощает процедуру умножения по правилу Верещагина. Вычисляем секториально-линейные статические моменты:

$$S_{y'\omega_B} = \int_F x' \omega_B dF = \delta \cdot \frac{1}{2} \cdot 4 \cdot 40 \left(\frac{2}{3} \cdot 2,18 - \frac{1}{3} \cdot 1,82 \right) = 13,576 \text{ см}^5,$$

$$S_{x'\omega_B} = \int_F y' \omega_B dF = \delta \cdot \frac{1}{2} \cdot 4 \cdot 40 \cdot 5,91 = 94,545 \text{ см}^5.$$

Проверяем координаты центра изгиба P :

$$x_P = \alpha_x^B + x_B = \frac{-S_{x'\omega_B} J_{y_c} + S_{y'\omega_B} J_{x y_c}}{J_{x_c} J_{y_c} - J_{x y_c}^2} = \frac{-2482,424}{1462,303} = -1,698 \text{ см},$$

$$y_P = \alpha_y^B + y_B = \frac{S_{y'\omega_B} J_{x_c} - S_{x'\omega_B} J_{x y_c}}{J_{x_c} J_{y_c} - J_{x y_c}^2} = \frac{2572,929}{1462,303} = 1,759 \text{ см}.$$

Значения получаются теми же. Можно переходить к заключительной части задачи — определению секториального момента инерции.

7. Строим эпюру секториальных площадей ω_P сечения при полюсе в центре изгиба P и начальной точке A (рис. 214). Поворот радиус-вектора от направления PA к PB происходит по часовой стрелке,

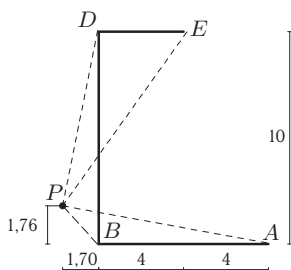


Рис. 214

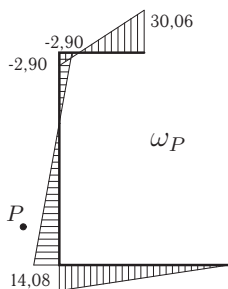


Рис. 215

поэтому секториальная координата точки B по отношению к точке A положительная и равна удвоенной площади треугольника PAB : $\omega_{A,B}^P = 1,759 \cdot 8 = 14,076$. От точки B к D радиус-вектор поворачивается против часовой стрелки, ометая отрицательную площадь. Поэтому $\omega_{A,D}^P = 14,076 - 1,697 \cdot 10 = 2,900$. К точке E поворот радиус-вектора с полюсом в P происходит по часовой стрелке, в результате $\omega_{A,E}^P = 2,900 + (10 - 1,759) \cdot 4 = 30,062$. На рисунке 215 изображена полученная эпюра.

8. Определяем константу D , задающую главные секториальные площади. Находим площадь построенной эпюры ω_P :

$$S_{\omega_P} = \int_F \omega_P dF = 0,2 \left(\frac{30,062 - 2,9}{2} \cdot 4 + \frac{14,076 - 2,9}{2} \cdot 10 + \frac{14,076}{2} \cdot 8 \right) = 33,302 \text{ см}^4.$$

Получаем $D = S_{\omega_P} / F = 33,302 / 4,4 = 7,568 \text{ см}^2$.

9. Вычитаем из ординат эпюры ω_P (рис. 215) константу 7,568. Эпюра главных секториальных площадей сечения $\omega = \omega_P - D$ получает вид (рис. 216). Для проверки вычислим площадь построенной эпюры, т. е. секториально-статический момент. Имеем $S_\omega = (22,49 - 10,47) / 2 \cdot 4 + (6,51 - -10,47) / 2 \cdot 10 + (6,51 - 7,57) / 2 \cdot 8 = 24,04 - 19,80 - 4,24 = 0$. Проверка выполнена.

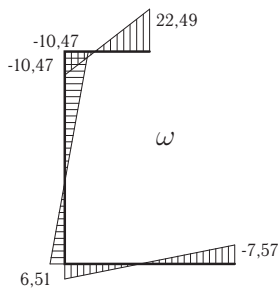


Рис. 216

10. Вычисляем секториальный момент инерции сечения J_ω . Умножаем эпюру ω (рис. 216) саму на себя. Интеграл берем по правилу Верещагина. Умножаем площадь эпюры на ее ординату под центром тяжести. Ординату выделяем шрифтом:

$$\begin{aligned}
 J_\omega = \int_F \omega^2 dF = & \delta \left(\frac{22,49}{2} \cdot 4 \cdot \left(\frac{2}{3} \cdot 22,49 - \frac{1}{3} \cdot 10,47 \right) - \right. \\
 & - \frac{10,47}{2} \cdot 4 \cdot \left(\frac{1}{3} \cdot 22,49 - \frac{2}{3} \cdot 10,47 \right) - \\
 & - \frac{10,47}{2} \cdot 10 \cdot \left(\frac{1}{3} \cdot 6,508 - \frac{2}{3} \cdot 10,47 \right) + \\
 & + \frac{6,508}{2} \cdot 10 \cdot \left(\frac{2}{3} \cdot 6,508 - \frac{1}{3} \cdot 10,47 \right) + \\
 & + \frac{6,508}{2} \cdot 8 \cdot \left(\frac{2}{3} \cdot 6,508 - \frac{1}{3} \cdot 7,569 \right) - \\
 & \left. - \frac{7,569}{2} \cdot 8 \cdot \left(\frac{1}{3} \cdot 6,508 - \frac{2}{3} \cdot 7,569 \right) \right) = \\
 = & 0,2(517,42 - 10,82 + 251,84 + 27,61 + 47,27 + 87,09) = 184,096 \text{ см}^6.
 \end{aligned}$$

Maple-программа определения секториального момента инерции приведена на с. 416.

4.4. Кручение тонкостенных стержней открытого профиля

Задача 53. К тонкостенному стержню открытого профиля приложен крутящий момент $M = 1$ Нм (рис. 217). Найти относительный угол закручивания θ и максимальное касательное напряжение. Модуль упругости материала $E = 0,7 \cdot 10^5$ МПа, коэффициент Пуассона $\nu = 0,3$. Размеры даны в миллиметрах.

Решение

Разбиваем сечение на три длинных прямоугольника $s_1 = 42$ мм, $\delta_1 = 1$ мм, $s_2 = 15$ мм, $\delta_2 = 2$ мм, $s_3 = 22$ мм, $\delta_3 = 1$ мм (рис. 218). По формуле (4.14) на с. 137 вычисляем геометрическую характеристику

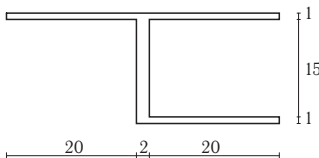


Рис. 217

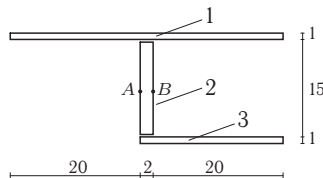


Рис. 218

жесткости сечения (момент инерции при кручении), выбирая, согласно таблице 6 на с. 413, значение поправочного коэффициента $\beta = 1/3$:

$$J_k = (1/3) \sum_i^3 l_i \delta_i^3 = (1/3)(42 \cdot 1^3 + 15 \cdot 2^3 + 22 \cdot 1^3) = 61,33 \text{ мм}^4.$$

Вычисляем относительный угол закручивания стержня. Модуль сдвига материала¹ $G = E/(2(1 + \nu)) = 0,7 \cdot 10^{11}/(2(1 + 0,3)) = 0,269 \times 10^{11}$ Па. Имеем

$$\theta = M/(GJ_k) = 1/(0,269 \cdot 10^{11} \cdot 61,33 \cdot 10^{-12}) = 0,605 \text{ рад.}$$

Вычисляем максимальное касательное напряжение в сечении. Максимальное касательное напряжение зависит от толщины δ_i участка $\tau_{\max}^i = \delta_i M/J_k$. Наибольшее значение будет там, где толщина максимальна. В данном случае берем $\delta = 2$ мм:

$$\tau_{\max} = \delta M/J_k = 1 \cdot 0,002/(61,33 \cdot 10^{-12}) = 0,326 \cdot 10^8 \text{ Па} = 32,6 \text{ МПа.}$$

Максимальное касательное напряжение наблюдается в точках A и B , рис. 218.

Замечание 1. Для *замкнутого* тонкостенного профиля постоянной толщины δ формулы для расчета относительного угла закручивания и касательного напряжения имеют вид

$$\theta = Ms/(4GF_*^2\delta), \quad \tau_{\max} = M/(2F_*\delta),$$

где F_* — площадь сечения, ограниченная средней линией замкнутого контура, s — длина контура.

Замечание 2. Открытый тонкостенный профиль плохо сопротивляется кручению. В этом можно убедиться, если сравнить полярный момент инерции круглого вала той же площади, что и в рассмотренном примере. Вычисляем площадь сечения на рис. 217, состоящего из прямоугольников: $F = 1 \cdot 42 + 15 \cdot 2 + 22 \cdot 1 = 94 \text{ мм}^2$. Из равенства площадей круглого сечения и тонкостенного $\pi d^2/4 = 94$ получаем диаметр вала сравнения $d = 10,94$ мм. Для этого вала полярный момент инерции равен $\pi \cdot (10,94)^4/32 = 1406,29 \text{ мм}^4$, что в 23 раза больше, чем аналогичная величина для тонкостенного профиля. Еще больший эффект получается от применения замкнутого тонкостенного профиля.

Задача 54. Тонкостенный стержень открытого профиля длиной $l = l_1 + l_2$ жестко заделан на одном конце. На стержень действуют продольные силы P_1 , P_2 и крутящий момент M (рис. 219). Найти угол поворота сечения стержня при $z = l_3 = 400$ мм, координата z

¹Алюминий или алюминиевый сплав.

отсчитывается от заделки. Даны размеры: $b = 30$ мм, $h = 20$ мм, $l_1 = 500$ мм, $l_2 = 400$ мм, нагрузки $P_1 = 100$ Н, $P_2 = 200$ Н, $M_0 = 3$ Нм, модуль упругости $E = 2 \cdot 10^5$ МПа, коэффициент Пуассона $\nu = 0,3$. Толщина δ стенки равна 1 мм.

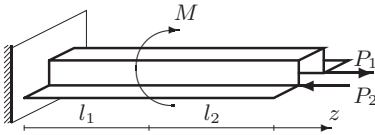


Рис. 219

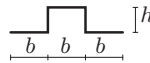


Рис. 220

Решение

1. Строим эпюру главных секториальных площадей сечения ω (мм²).

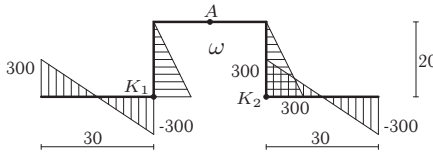


Рис. 221

Центр изгиба располагается в точке A .

2. Определяем секториальный момент инерции сечения J_ω . Умножаем эпюру ω саму на себя. Интеграл берем по правилу Верещагина. Учитываем симметрию сечения:

$$J_\omega = \int_F \omega^2 dF = 2\delta \left(\left(\frac{300}{2} \cdot 20 \cdot \frac{2}{3} \cdot 300 \right) + 2 \cdot \frac{300}{2} \cdot 30 \cdot \left(\frac{2}{3} \cdot 300 - \frac{1}{3} \cdot 300 \right) \right) = 3 \cdot 10^6 \text{ мм}^6 = 3 \text{ см}^6.$$

3. Вычисляем геометрическую характеристику жесткости сечения. Толщина на всех участках одинаковая: $\delta = 1$ мм. Получаем

$$J_k = (1/3) \sum_{k=1}^5 s_k \delta_k^3 = (1/3) \cdot 1 \cdot (30 + 20 + 30 + 20 + 30) = 43,33 \text{ мм}^4.$$

4. Находим изгибно-крутильную характеристику стержня $k = \sqrt{GJ_k/(EJ_\omega)}$. Так как модуль сдвига имеет вид $G = E/(2(1 + \nu))$,

то

$$k = \sqrt{\frac{J_k}{2(1+\nu)J_\omega}} = \sqrt{\frac{43,33 \cdot 10^{-12}}{2(1+0,3)3 \cdot 10^{-12}}} = \sqrt{5,555} = 2,357 \text{ м}^{-1}.$$

5. Выделяем на стержне два участка $0 \leq z \leq l_1$, $l_1 \leq z \leq l$, разделенных точкой приложения внешнего момента M . На первом участке удельный угол закручивания обозначим θ_1 , на втором — θ_2 .

6. Записываем дифференциальные уравнения для удельного угла закручивания θ на участках:

$$\begin{aligned} \theta_1'' - k^2\theta_1 &= -k^2M_z/(GJ_k), \\ \theta_2'' - k^2\theta_2 &= 0, \end{aligned} \quad (4.22)$$

где штрихом обозначена производная по осевой координате z ; $M_z = M_0$. Момент положительный, так как он направлен в ту же сторону, куда отсчитываются положительные секториальные площади, если смотреть с конца оси z , т. е. по часовой стрелке. На втором участке внешних моментов справа от участка нет, уравнение получается однородным. Вычислим правую часть первого уравнения:

$$k^2M_z/(GJ_k) = 5,555 \cdot 3/(0,769 \cdot 10^{11} \cdot 0,433 \cdot 10^{-10}) = 5 \text{ м}^{-3}.$$

7. Определяем бимомент, возникающий от действия продольных сил на конце стержня:

$$B = \omega_1 P_{1z} + \omega_2 P_{2z} = 300 \cdot 100 + (-300) \cdot (-200) = 90000 \text{ Нмм}^2 = 0,09 \text{ Нм}^2.$$

8. Интегрируем¹ дифференциальные уравнения (4.22):

$$\begin{aligned} \theta_1(z) &= C_1 \text{sh}(kz) + C_2 \text{ch}(kz) + M_0/(GJ_k), \\ \theta_2(z) &= C_3 \text{sh}(kz) + C_4 \text{ch}(kz). \end{aligned} \quad (4.23)$$

Константы интегрирования определяем из условий на концах стержня:

$$\theta_2'(l) = -B/(EJ_\omega), \quad \theta_1(0) = 0, \quad (4.24)$$

и условий сопряжения при $z = l_1$:

$$\theta_1'(l_1) = \theta_2'(l_1), \quad \theta_1(l_1) = \theta_2(l_1). \quad (4.25)$$

Для определения констант потребуются производные:

$$\begin{aligned} \theta_1'(z) &= kC_1 \text{ch}(kz) + kC_2 \text{sh}(kz), \\ \theta_2'(z) &= kC_3 \text{ch}(kz) + kC_4 \text{sh}(kz). \end{aligned} \quad (4.26)$$

¹Решение можно искать также в виде $\theta_1(z) = C_1 e^{kz} + C_2 e^{-kz} + M_0/(GJ_k)$, $\theta_2(z) = C_3 e^{kz} + C_4 e^{-kz}$.

Подставляем (4.24) и (4.25) в (4.23) и (4.26), получаем систему уравнений для констант C_1, C_2, C_3, C_4 :

$$\begin{aligned} 9,97C_3 + 9,69C_4 + 0,15 &= 0, \\ C_2 + 0,90 &= 0, \\ 4,19C_1 + 3,47C_2 - 4,19C_3 - 3,47C_4 &= 0, \\ 1,47C_1 + 1,78C_2 - 1,47C_3 - 1,78C_4 + 0,9 &= 0. \end{aligned} \quad (4.27)$$

Решаем систему (4.27). Находим константы интегрирования $C_1 = 0,63, C_3 = -0,69, C_4 = 0,7, C_2 = -0,9$.

9. Вычисляем угол поворота в сечении $z = l_3$:

$$\begin{aligned} \varphi &= \int_0^{l_3} \theta dz = (1/k)(C_1 \operatorname{ch}(kz) + C_2 \operatorname{sh}(kz)) + zM_0/(GJ_k) \Big|_0^{l_3} = \\ &= (1/k)(C_1(\operatorname{ch}(kl_3) - 1) + C_2 \operatorname{sh}(kl_3)) + l_3M_0/(GJ_k) = 0,07166 \text{ рад.} \end{aligned}$$

Maple-программа интегрирования дифференциального уравнения кручения тонкостенного стержня приведена на с. 420.

4.5. Напряжения и деформации

Задача 55. К треугольной пластине (рис. 223) с заданными размерами $AB = 10$ см, $BC = 7$ см приложены напряжения в МПа. Из условия равновесия пластины найти недостающие напряжения, проверить ее прочность, узнать деформации пластины и смещения ее вершин. Коэффициент Пуассона $\nu = 0,3$, нормативное сопротивление $R_n = 100$ МПа, модуль упругости $E = 2$ ГПа, коэффициент запаса прочности $[k_{пч}] = 1,5$.

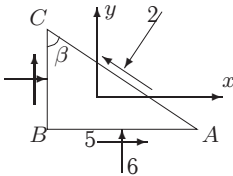


Рис. 222

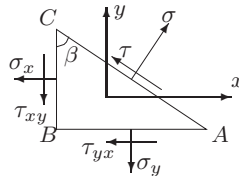


Рис. 223

Решение

1. В соответствии с правилом знаков на рис. 223 получаем величины напряжений на площадках $\sigma_y = -6$ МПа, $\tau_{xy} = -5$ МПа, $\sigma = -2$ МПа. По закону парности $\tau_{xy} = \tau_{yx}$. Неизвестными являются

касательное напряжение τ на наклонной площадке и нормальное напряжение σ_x . Векторы этих напряжений изображаем в положительных направлениях.

2. Вычисляем незаданные напряжения. Находим площади граней $F_{BC} = 7 \cdot t$, $F_{AB} = 10 \cdot t$, $F_{AC} = \sqrt{7^2 + 10^2} t = 12,21t$, где t — толщина пластины. Записываем уравнения равновесия пластины в проекциях на оси с учетом закона парности $\tau_{xy} = \tau_{yx}$:

$$\sum X_k = -\sigma_x \cdot F_{BC} - \tau_{yx} \cdot F_{AB} + \sigma \cdot \cos \beta \cdot F_{AC} - \tau \cdot \sin \beta \cdot F_{AC} = 0,$$

$$\sum Y_k = -\sigma_y \cdot F_{AB} - \tau_{xy} \cdot F_{BC} + \sigma \cdot \sin \beta \cdot F_{AC} + \tau \cdot \cos \beta \cdot F_{AC} = 0.$$

Так как $BC = AC \cos \beta$, $AB = AC \sin \beta$, $\sin \beta = 10/\sqrt{7^2 + 10^2} = 10/12,21 = 0,82$, $\cos \beta = 7/12,21 = 0,57$, система уравнений принимает вид

$$\begin{aligned} \sum X_k &= -\sigma_x \cdot 0,57 + 5 \cdot 0,82 - 2 \cdot 0,57 - \tau \cdot 0,82 = 0, \\ \sum Y_k &= 6 \cdot 0,82 + 5 \cdot 0,57 - 2 \cdot 0,82 + \tau \cdot 0,57 = 0. \end{aligned} \quad (4.28)$$

Решаем систему уравнений (4.28). Получаем $\tau = -10,71$ МПа, $\sigma_x = 20,45$ МПа.

3. Определяем главные напряжения:

$$\sigma_{\max} = (\sigma_x + \sigma_y)/2 + \sqrt{(\sigma_x - \sigma_y)^2/4 + \tau_{xy}^2} = 21,36 \text{ МПа},$$

$$\sigma_{\min} = (\sigma_x + \sigma_y)/2 - \sqrt{(\sigma_x - \sigma_y)^2/4 + \tau_{xy}^2} = -6,91 \text{ МПа}.$$

4. Находим максимальное и минимальное касательное напряжение:

$$\tau_{\max} = \frac{1}{2}(\sigma_{\max} - \sigma_{\min}) = \frac{1}{2}(21,36 + 6,91) = 14,14 \text{ МПа},$$

$$\tau_{\min} = -\tau_{\max} = -14,14 \text{ МПа}.$$

5. Вычисляем угол наклона главных осей:

$$\operatorname{tg}(\alpha_{\max}) = \frac{\sigma_{\max} - \sigma_x}{\tau_{xy}} = \frac{21,36 - 20,45}{-5} = -0,18,$$

получаем $\alpha_{\max} = -10,36^\circ$.

6. Определяем главные напряжения. Так как $\sigma_1 > \sigma_2 > \sigma_3$, то $\sigma_1 = 21,36$ МПа, $\sigma_2 = 0$, $\sigma_3 = -6,91$ МПа.

7. Рассчитываем деформации пластинки. Вычисляем по закону Гука относительные главные деформации, или деформации ε_{x1} , ε_{y1} в осях,

повернутых на угол α_{\max} . С учетом значения модуля Юнга $E = 2 \times 10^3$ МПа и коэффициента Пуассона $\nu = 0,3$ получаем

$$\varepsilon_{x1} = \varepsilon_{\max} = \frac{\sigma_{\max} - \nu\sigma_{\min}}{E} = \frac{21,36 + 6,91 \cdot 0,3}{2 \cdot 10^3} = 117,18 \cdot 10^{-4},$$

$$\varepsilon_{y1} = \varepsilon_{\min} = \frac{\sigma_{\min} - \nu\sigma_{\max}}{E} = \frac{-6,91 - 21,36 \cdot 0,3}{2 \cdot 10^3} = -66,61 \cdot 10^{-4},$$

$$\varepsilon_z = -\nu \frac{\sigma_{\max} + \sigma_{\min}}{E} = -0,3 \frac{21,36 - 6,91}{2 \cdot 10^3} = -21,67 \cdot 10^{-4}.$$

Удельное изменение объема: $\Delta V/V = \varepsilon_{x1} + \varepsilon_{y1} + \varepsilon_z = 28,9 \cdot 10^{-4}$.

8. Определяем искажение формы пластинки, вызванное напряженным состоянием. Находим перемещение вершин треугольника в осях, связанных с его центром тяжести. Вычисляем координаты вершин в центральных осях xu и в повернутых на угол α_{\max} осях x_1y_1 по формулам

$$\begin{aligned} x_1 &= x \cos \alpha_{\max} + y \sin \alpha_{\max}, \\ y_1 &= y \cos \alpha_{\max} - x \sin \alpha_{\max}. \end{aligned} \quad (4.29)$$

С учетом того что $x_A = x_B + 10$ и $x_C = x_B$, а координата центра тяжести по условию равна нулю, из уравнения $x_0 = 0 = (x_A + x_B + x_C)/3$ получаем $x_B = -3,33$ см, аналогично из уравнений $y_C = y_B + 7$, $y_A = y_B$, $y_0 = 0 = (y_A + y_B + y_C)/3$ находим $y_A = -2,33$ см. Результаты вычислений заносим в таблицу

	x	y	x_1	y_1
A	6,67	-2,33	6,98	-1,10
B	-3,33	-2,33	-2,86	-2,90
C	-3,33	4,67	-4,12	4,00

В формулах (4.29) используем значения тригонометрических функций $\cos \alpha_{\max} = 0,98$, $\sin \alpha_{\max} = -0,18$.

9. Определяем перемещения вершин, вызванные деформацией:

$$\Delta x_{1A} = x_{1A} \varepsilon_{x1} = 6,98 \cdot 117,18 \cdot 10^{-4} = 81,76 \cdot 10^{-3} \text{ см},$$

$$\Delta y_{1A} = y_{1A} \varepsilon_{y1} = 1,1 \cdot 66,61 \cdot 10^{-4} = 7,3 \cdot 10^{-3} \text{ см},$$

$$\Delta x_{1B} = x_{1B} \varepsilon_{x1} = -2,86 \cdot 117,18 \cdot 10^{-4} = -33,5 \cdot 10^{-3} \text{ см},$$

$$\Delta y_{1B} = y_{1B} \varepsilon_{y1} = 2,9 \cdot 66,61 \cdot 10^{-4} = 19,28 \cdot 10^{-3} \text{ см},$$

$$\Delta x_{1C} = x_{1C} \varepsilon_{x1} = -4,12 \cdot 117,18 \cdot 10^{-4} = -48,25 \cdot 10^{-3} \text{ см},$$

$$\Delta y_{1C} = y_{1C} \varepsilon_{y1} = -4 \cdot 66,61 \cdot 10^{-4} = -26,59 \cdot 10^{-3} \text{ см}.$$



Изображаем схему деформации пластинки (рис. 224), существенно увеличивая для наглядности деформации.

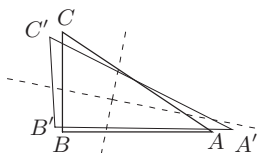


Рис. 224

10. Оцениваем прочность материала. Расчетное сопротивление $R = 100$ МПа. Вычисляем приведенные напряжения σ_i и проверяем условие прочности $\sigma_i < R$. Определяем коэффициент запаса прочности $k_{пч}$ и по критерию $k_{пч} > [k_{пч}]$ оцениваем возможность разрушения или течения материала в предположении, что:

а) Материал хрупкий,

I-я гипотеза прочности: $\sigma_i = \sigma_1 = 21,36$ МПа $< R = 100$ МПа. Коэффициент запаса прочности $k_{пч(I)} = 100/21,36 = 4,68 > [k_{пч}] = 1,5$. Вывод: прочность обеспечена.

II-я гипотеза прочности: $\sigma_i = \sigma_1 - \nu(\sigma_2 + \sigma_3) = 23,44$ МПа $< R = 100$ МПа. Коэффициент запаса прочности $k_{пч(II)} = 100/23,44 = 4,27 > [k_{пч}] = 1,5$. Вывод: прочность обеспечена. Разрушение не произойдет.

б) Материал пластичный,

III-я гипотеза прочности: $\sigma_i = \sigma_1 - \sigma_3 = 28,28$ МПа $< R = 100$ МПа. Коэффициент запаса прочности $k_{пч(III)} = 100/28,28 = 3,54 > [k_{пч}] = 1,5$. Вывод: прочность обеспечена.

IV-я гипотеза прочности (энергетическая):

$$\begin{aligned} \sigma_i &= \sqrt{0,5((\sigma_1 - \sigma_2)^2 + (\sigma_1 - \sigma_3)^2 + (\sigma_2 - \sigma_3)^2)} = \\ &= \sqrt{0,5(456,36 + 47,8 + 799,55)} = 25,53 \text{ МПа} < R = 100 \text{ МПа}. \end{aligned}$$

Коэффициент запаса прочности $k_{пч(IV)} = 100/25,53 = 3,92 > [k_{пч}] = 1,5$. Вывод: прочность обеспечена¹.

Maple-программа для решения этой задачи дана на с. 399.

Задача 56. На элементарный объем тела действуют заданные напряжения (рис. 225). Найти главные нормальные и касательные напряжения, относительные главные деформации, относительное изменение объема, октаэдрическое напряжение. Дан модуль упругости материала² $E = 1 \cdot 10^5$ МПа и коэффициент Пуассона $\nu = 0,4$.

¹ Существует и пятая гипотеза прочности [55]. Согласно этой гипотезе $\sigma_i = \sigma_1 - k\sigma_3$, где коэффициент k вычисляется для одноосного напряженного состояния (испытание стержня) и равен отношению предельного напряжения при растяжении к предельному напряжению при сжатии: $k = \sigma_t/\sigma_c$.

² Медь или медный сплав.

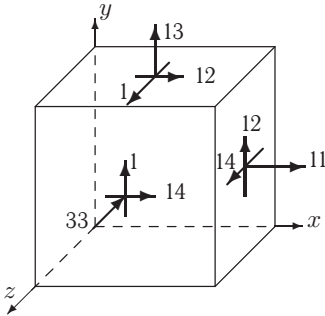


Рис. 225

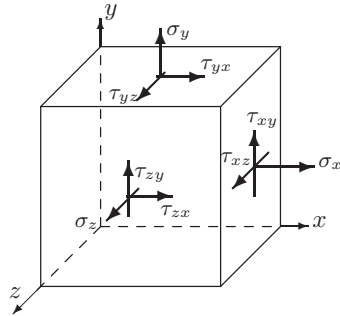


Рис. 226

Решение

Для того чтобы определить главные напряжения σ_1 , σ_2 , σ_3 , необходимые для оценки прочности материала, требуются компоненты тензора напряжений. Если модули и направления напряжений заданы, то остается определить их знаки в тензоре напряжений, руководствуясь принятым правилом. Нормальные напряжения σ_x , σ_y , σ_z положительные, если они растягивающие, т. е. направлены по внешней нормали к элементу объема. Первый индекс i касательного напряжения $\tau_{i,j}$ есть орт оси, нормальный к площадке, на которой это напряжение действует. Второй индекс j указывает орт оси, вдоль которой направлено напряжение. Если внешняя нормаль к площадке направлена по направлению орта i , то положительным касательным напряжением будет то, которое направлено в сторону орта j .

1. Правило знаков (рис. 226) дает следующие компоненты тензора напряжений: $\sigma_x = 11$ МПа, $\sigma_y = 13$ МПа, $\sigma_z = -33$ МПа, $\tau_{xy} = 12$ МПа, $\tau_{xz} = 14$ МПа, $\tau_{yz} = 1$ МПа.

2. Вычисляем инварианты тензора напряжения:

$$J_1 = -(\sigma_x + \sigma_y + \sigma_z) = -11 - 13 + 33 = 9 \text{ МПа},$$

$$J_2 = \sigma_x \sigma_y + \sigma_y \sigma_z + \sigma_x \sigma_z - \tau_{xy}^2 - \tau_{zx}^2 - \tau_{yz}^2 = -990 \text{ МПа}^2, \quad (4.30)$$

$$J_3 = \sigma_x \tau_{yz}^2 + \sigma_y \tau_{xz}^2 + \sigma_z \tau_{xy}^2 - \sigma_x \sigma_y \sigma_z - 2\tau_{yz} \tau_{yx} \tau_{xz} = 2190 \text{ МПа}^3.$$

3. Решаем кубическое уравнение $\sigma^3 + J_1 \sigma^2 + J_2 \sigma + J_3 = 0$. Используем итерационный метод Ньютона¹. В скобках указываем номер итерации ($k = 0, 1, \dots, n$):

¹ В системе **Maple** для решения этого уравнения достаточно одного оператора: `fsolve(sigma**3+ 9.*sigma**2 - 990.*sigma + 2190.)`.

$$\sigma_{(k+1)} = \sigma_{(k)} - \frac{\sigma_{(k)}^3 + J_1\sigma_{(k)}^2 + J_2\sigma_{(k)} + J_3}{3\sigma_{(k)}^2 + 2J_1\sigma_{(k)} + J_2}.$$

Даем начальное приближение $\sigma_{(0)} = (\sigma_x + \sigma_y + \sigma_z)/3 = -3$ МПа.

$$\sigma_{(1)} = -3 - \frac{(-3)^3 + 9(-3)^2 + 990 \cdot 3 + 2190}{3(-3)^2 - 18 \cdot 3 - 990} = -3 + 5,1268 = 2,127 \text{ МПа,}$$

$$\sigma_{(2)} = 2,127 - \frac{(2,127)^3 + 9(2,127)^2 - 990 \cdot 2,127 + 2190}{3(2,127)^2 + 18 \cdot 2,127 - 990} = 2,271 \text{ МПа,}$$

$$\sigma_{(3)} = 2,271 - \frac{(2,271)^3 + 9(2,271)^2 - 990 \cdot 2,271 + 2190}{3(2,271)^2 + 18 \cdot 2,271 - 990} = 2,271 \text{ МПа.}$$

Две последовательные итерации дали одно и то же значение (в пределах заданной точности, в данном случае с точностью до 0,001 МПа). Следовательно, первый корень найден: $\sigma = 2,27$ МПа. Делим кубическое уравнение $\sigma^3 + 9\sigma^2 - 990\sigma + 2190 = 0$ на $\sigma - 2,27$:

$$\begin{array}{r|l} \sigma^3 + 9\sigma^2 - 990\sigma + 2190 & \sigma - 2,27 \\ \underline{\sigma^3 - 2,27\sigma^2} & \sigma^2 + 11,27\sigma - 964,4 \\ 11,27\sigma^2 - 990\sigma + 2190 & \\ \underline{11,27\sigma^2 - 25,6\sigma} & \\ -964,4\sigma + 2190 & \\ \underline{-964,4\sigma + 2190} & \end{array}$$

Если первый корень найден верно, то деление произойдет без остатка. Получаем квадратное уравнение $\sigma^2 + 11,27\sigma - 964,4 = 0$, решаем его и находим еще два значения: $\sigma_1^* = 25,93$ МПа, $\sigma_2^* = -37,20$ МПа, или (в порядке убывания) $\sigma_1 = 25,93$ МПа, $\sigma_2 = 2,27$ МПа, $\sigma_3 = -37,20$ МПа.

4. Выполняем проверку. Находим инварианты:

$$\begin{aligned} J_1 &= -(\sigma_1 + \sigma_2 + \sigma_3) = -25,93 - 2,27 + 37,20 = 9 \text{ МПа,} \\ J_2 &= \sigma_1\sigma_2 + \sigma_2\sigma_3 + \sigma_1\sigma_3 = \\ &= 25,93 \cdot 2,27 - 2,27 \cdot 37,2 - 25,93 \cdot 37,2 = -990 \text{ МПа,} \\ J_3 &= -\sigma_1\sigma_2\sigma_3 = -2190 \text{ МПа.} \end{aligned} \quad (4.31)$$

Результат совпадает с величинами, вычисленными ранее (4.30).

5. Находим направляющие косинусы главных осей. Записываем систему уравнений

$$\begin{aligned} (\sigma_y - \sigma_1)m_1/l_1 + \tau_{yz}n_1/l_1 + \tau_{xy} &= 0, \\ \tau_{xy}m_1/l_1 + \tau_{xz}n_1/l_1 + (\sigma_x - \sigma_1) &= 0. \end{aligned} \quad (4.32)$$

Обозначаем $x_1 = m_1/l_1$, $x_2 = n_1/l_1$. Система (4.32) принимает вид

$$\begin{aligned} -12,97x_1 + 1 \cdot x_2 + 12 &= 0, \\ 12x_1 + 14x_2 - 14,97 &= 0. \end{aligned} \quad (4.33)$$

Решаем систему (4.33), находим $x_1 = 0,95$, $x_2 = 0,25$. Затем вычисляем один направляющий косинус: $l_1 = 1/\sqrt{1 + x_1^2 + x_2^2} = 1/\sqrt{1 + (0,95)^2 + (0,25)^2} = 0,71$, и выражаем через него другие: $m_1 = x_1 l_1 = 0,95 \cdot 0,71 = 0,68$, $n_1 = x_2 l_1 = 0,25 \cdot 0,71 = 0,18$.

6. Проверяем решение:

$$m_1^2 + n_1^2 + l_1^2 = (0,68)^2 + (0,18)^2 + (0,71)^2 = 0,46 + 0,03 + 0,51 = 1.$$

7. По закону Гука находим относительные главные деформации:

$$\begin{aligned} \varepsilon_1 &= \frac{\sigma_1 - \nu(\sigma_2 + \sigma_3)}{E} = \frac{25,93 + 0,4 \cdot 34,93}{1,0} \cdot 10^{-5} = 3,99 \cdot 10^{-4}, \\ \varepsilon_2 &= \frac{\sigma_2 - \nu(\sigma_3 + \sigma_1)}{E} = \frac{2,27 + 0,4 \cdot 11,27}{1,0} \cdot 10^{-5} = 0,68 \cdot 10^{-4}, \\ \varepsilon_3 &= \frac{\sigma_3 - \nu(\sigma_1 + \sigma_2)}{E} = \frac{-37,2 - 0,4 \cdot 28,2}{1,0} \cdot 10^{-5} = -4,85 \cdot 10^{-4}. \end{aligned} \quad (4.34)$$

8. Вычисляем относительное изменение объема:

$$(V_1 - V_0)/V_0 = \varepsilon_1 + \varepsilon_2 + \varepsilon_3 = (3,99 + 0,68 - 4,85) \cdot 10^{-4} = -0,18 \cdot 10^{-4}.$$

9. Находим главные касательные напряжения:

$$\begin{aligned} \tau_1 &= (\sigma_2 - \sigma_3)/2 = (2,27 + 37,20)/2 = 19,73 \text{ МПа}, \\ \tau_2 &= -(\sigma_3 - \sigma_1)/2 = -(-37,20 - 25,93)/2 = 31,56 \text{ МПа}, \\ \tau_3 &= (\sigma_1 - \sigma_2)/2 = (25,93 - 2,27)/2 = 11,83 \text{ МПа}. \end{aligned} \quad (4.35)$$

10. Находим октаэдрическое напряжение:

$$\tau_{oct} = \frac{2}{3} \sqrt{\tau_1^2 + \tau_2^2 + \tau_3^2} = \frac{2}{3} \sqrt{19,73^2 + 31,56^2 + 11,83^2} = 26,04 \text{ МПа}.$$

Maple-программа для решения этой задачи дана на с. 400.

4.6. Прогиб фермы

Задача 57. Ферма состоит из n одинаковых стержней, расположенных по окружности радиуса r , и радиальных стержней, соединенных с центром (рис. 227). Найти прогиб середины пролета фермы под действием силы P .



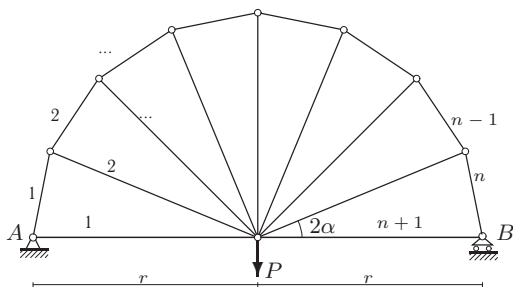


Рис. 227

Решение

Очевидно, вертикальные реакции опор фермы равны $P/2$, горизонтальная реакция неподвижной опоры равна нулю. На рисунке 227 ферма изображена при $n = 8$.

Рассмотрим правый опорный узел B , на который действует вертикальная реакция $Y_B = P/2$ и усилия двух стержней (рис. 228).

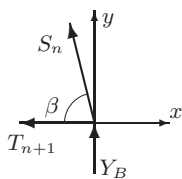


Рис. 228

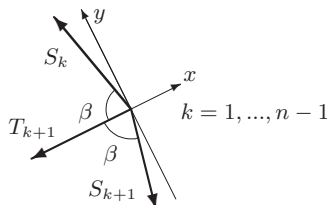


Рис. 229

Усилия в окружных стержнях обозначим S_k , $k = 1, \dots, n$, радиальных — T_k , $k = 1, \dots, n+1$. Из условия равновесия узла B в проекциях на оси координат получаем два уравнения

$$\begin{aligned} \sum X_k &= -T_{n+1} - S_n \cos \beta = 0, \\ \sum Y_k &= Y_B + S_n \sin \beta = 0, \end{aligned}$$

где $\beta = \pi/2 - \alpha$, $\alpha = \pi/(2n)$, $Y_B = P/2$. Находим решение системы: $S_n = -Y_B / \sin \beta = -P/(2 \cos \alpha)$, $T_{n+1} = -S_n \cos \beta = (P/2) \operatorname{tg} \alpha$. Из симметрии задачи $T_1 = T_{n+1}$, $T_k = T_{n+2-k}$, $k = 1, \dots, n+1$.

Рассмотрим равновесие произвольного k -го узла на окружности фермы. Наклоним систему координат. Ось x направим по радиальному стержню (рис. 229):

$$\begin{aligned} \sum X_k &= -T_{k+1} - S_{k+1} \cos \beta - S_k \cos \beta = 0, \\ \sum Y_k &= S_k \sin \beta - S_{k+1} \sin \beta = 0. \end{aligned}$$

Из второго уравнения получаем, что усилия во всех окружных стержнях одинаковы: $S_k = -P/(2 \cos \alpha)$, $k = 1, \dots, n$, т. е. эти стержни сжаты. Из первого уравнения находим усилия в радиальных стержнях $T_{k+1} = -2S_{k+1} \cos \beta = P \operatorname{tg} \alpha$, $k = 1, \dots, n$.

По формуле Максвелла–Мора (3.47) найдем вертикальное перемещение точки приложения силы P . Учитывая, что длины радиальных стержней r , а окружных — $l = 2r \sin \alpha$, получим

$$\begin{aligned} \delta &= (1/EF) \left(\sum_{k=2}^n T_k^2 r + T_1^2 r + T_{n+1}^2 r + \sum_{k=1}^n S_k^2 l \right) = \\ &= (Pr/EF) ((n-1/2) \operatorname{tg}^2 \alpha + (n/2) \sin \alpha / \cos^2 \alpha). \end{aligned}$$

Интересно отметить, что это выражение имеет предел при $n \rightarrow \infty$ $\lim_{n \rightarrow \infty} \delta = Pr\pi/(4EF)$. В системе **Maple** этот предел вычисляется так: `limit(delta, n=infinity)`.

4.7. Статически неопределимые стержневые системы

4.7.1. Шарнирно закрепленный брус

Задача 58. Шарнирно закрепленный на неподвижной опоре абсолютно твердый брус AB удерживается в равновесии двумя упругими стержнями BD длиной l_1 и CE длиной l_2 (рис. 230).

На брус действует сила $P = 89$ Н. Даны размеры $AK = KC = 2$ м, $CB = 1$ м, и соотношение длин стержней $l_1 = 2l_2$. Жесткости стержней одинаковы. Определить усилия в стержнях.

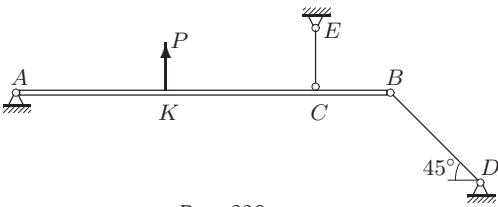


Рис. 230

Решение

С точки зрения теоретической механики такую систему рассчитать нельзя — в задаче содержится четыре неизвестных, для которых можно записать только три уравнения статики. Но если учесть деформации опорных стержней, то условие совместности их деформаций дает еще одно уравнение, позволяющее решить задачу.

Способ 1. Уравнения совместности деформаций

1. Находим геометрическое соотношение между абсолютными деформациями стержней Δl_1 и Δl_2 . Рассматриваем малую деформацию системы. Предполагаем, что брус повернулся на малый угол против часовой стрелки (рис. 231). Выбор направления несущественен. Ход

расчета и его результат не изменятся, если повернуть брус в обратном направлении. Предполагаем, что точки стержней при повороте перемещаются по малым дугам, которые заменяем отрезками, перпендикулярными к первоначальному (недеформированному) положению стержней. Шарнир B переместится вертикально вверх и займет положение B' . Стержень BD удлинится на $\Delta l_1 = BB'$ и повернется на некоторый малый угол. Перемещение шарнира B и удлинение стержня связаны.

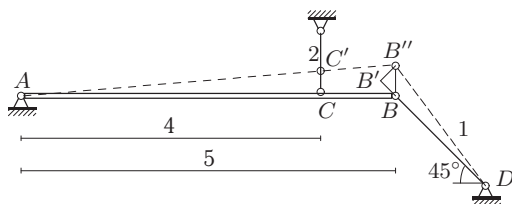


Рис. 231

Перемещение точки B в положение B'' представим как результат удлинения стержня 1 в положение B' с последующим поворотом стержня на малый угол в положение B'' . Из прямоугольного треугольника $BB'B''$, где $B'D \perp B'B''$, имеем $BB' = BB'' \cos 45^\circ$.

Шарнир C также переместится вверх и займет положение C' . Стержень CE укоротится: $\Delta l_2 = -CC'$. Знак минус ставим в случае укорочения стержня. Из подобия треугольников ABB'' и ACC' имеем $CC'/AC = BB''/AB$. С учетом выражений для BB'' и CC' имеем искомое геометрическое соотношение

$$-\Delta l_2/4 = \Delta l_1/(5 \cos 45^\circ). \quad (4.36)$$

2. Используем закон Гука $\Delta l_1 = l_1 S_1/(EF)$, $\Delta l_2 = l_2 S_2/(EF)$, где EF — жесткость стержней. С учетом (4.36) и соотношения $l_1 = 2l_2$ после сокращения на жесткость EF получаем дополнительное уравнение для усилий

$$8\sqrt{2}S_1 + 5S_2 = 0. \quad (4.37)$$

3. Записываем уравнение моментов всех сил, действующих на брус, включая неизвестные реакции стержней, относительно неподвижного шарнира. Силы S_1 и S_2 , приложенные в действительности к стержням 1 и 2 в точках D и E соответственно, при составлении уравнения равновесия можно для удобства перенести вдоль линий их действия к точкам B и C (рис. 232):

$$\sum M_A = P \cdot AK + S_2 \cdot AC - S_1 \cdot AB \sin 45^\circ = 0. \quad (4.38)$$

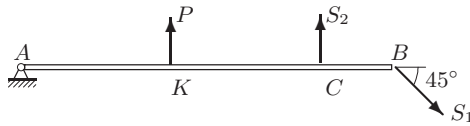


Рис. 232

4. Решаем систему уравнений (4.37), (4.38). Находим усилия в стержнях $S_1 = 14,14$ Н, $S_2 = -32$ Н. Стержень 2 оказывается сжат, стержень 1 — растянут.

Способ 2. Метод сил

Основой метода является формула Максвелла–Мора для вычисления перемещений в упругих системах (3.47, с. 118). В том случае, когда деформация является следствием удлинения или укорочения стержней, вызванных продольными силами (моментом и поперечной силой пренебрегаем), формула для перемещения по направлению вектора \vec{e}_i от действия единичной силы вдоль \vec{e}_j имеет вид

$$\delta_{ij} = \sum_k^n \frac{s_{k,i} s_{k,j} l_k}{EF}, \quad (4.39)$$

где $s_{k,i}$ и $s_{k,j}$ — усилия в стержне k от единичных сил, действующих по направлению \vec{e}_i и \vec{e}_j соответственно, l_k — длины стержней. Суммирование производится по всем стержням. Перемещения по направлению вектора \vec{e}_i от действия внешней нагрузки вычисляем по той же формуле Максвелла–Мора

$$\Delta_{ip} = \sum_k^n \frac{s_{k,i} s_{k,p} l_k}{EF}, \quad (4.40)$$

где $s_{k,p}$ — усилие в стержне k от действия внешней нагрузки.

По методу сил действие лишних связей заменяем неизвестными силами. Полученная статически определимая система называется основной¹. Значения неизвестных подбираем из условия выполнения кинематических ограничений связей. Соответствующие перемещения (угловые или линейные) должны быть равны нулю. Эти уравнения для

¹ Иногда вспомогательную статически определимую систему с отброшенными лишними связями называют основной, а вспомогательную статически определимую систему, в которой лишние связи заменены неизвестными силами, — эквивалентной.

n лишних связей имеют вид

$$\sum_{j=1}^n \delta_{ij} S_j + \Delta_{ip} = 0, \quad i = 1, \dots, n-1. \quad (4.41)$$

Решение

1. Выбираем основную систему. Освобождаем от связи стержень 1 в месте его крепления к шарниру D . Усилие S_1 в этом стержне назначаем неизвестным метода сил. Основная система представляет собой статически определимую систему, состоящую из шарнирно закрепленного бруса с одним опорным стержнем 2, на который действуют внешняя сила P и неизвестное усилие S_1 (рис. 233).

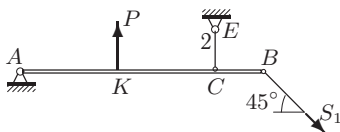


Рис. 233

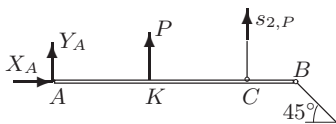


Рис. 234

2. Находим усилие $s_{2,P}$ в опорном стержне 2 основной системы от действия внешней нагрузки P . Записываем уравнение моментов сил, приложенных к брусу, относительно неподвижного шарнира A (рис. 234). Стержень 1 остается ненагруженным (его отсоединили от опоры D при получении основной системы). Приравниваем сумму моментов нулю: $s_{2,P}AC + P \cdot AK = 0$. Получаем усилие $s_{2,P} = -P/2 = -44,5$ кН.

3. Находим усилие $s_{2,1}$ в опорном стержне 2 основной системы от единичной (безразмерной) нагрузки, приложенной к брусу вместо реакции лишней связи S_1 (рис. 235).

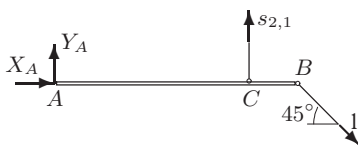


Рис. 235

Из уравнения моментов относительно опоры A (рис. 235) следует

$$-1 \cdot AB \sin 45^\circ + s_{2,1}AC = 0.$$

Решаем это уравнение: $s_{2,1} = 5\sqrt{2}/8 = 0,884$.

4. По формулам Максвелла–Мора (4.39), (4.40) определяем коэффициенты канонических уравнений δ_{11} и Δ_{1P} . С учетом $l_2 = 0,5l_1$ и

$s_{1,1} = 1, s_{1,P} = 0$ получаем

$$\delta_{11} = \sum_{k=1}^2 \frac{s_{k,1}^2 l_k}{EF} = \frac{s_{1,1}^2 \cdot l_1 + s_{2,1}^2 \cdot l_2}{EF} = \frac{1,39 l_1}{EF},$$

$$\Delta_{1P} = \sum_{k=1}^2 \frac{s_{k,1} s_{k,P} l_k}{EF} = \frac{s_{2,1} s_{2,P} \cdot l_2}{EF} = -\frac{0,884 \cdot 44,5 \cdot 0,5 l_1}{EF} = -\frac{19,67 l_1}{EF}.$$

5. Составляем и решаем каноническое уравнение метода сил. Для один раз статически неопределимой системы имеем одно уравнение: $\delta_{11} S_1 + \Delta_{1P} = 0$. С учетом найденных значений коэффициентов получаем $1,39 S_1 - 19,67 = 0$. Жесткость EF и длина l_1 в это уравнение не вошли, решаем его. Находим реакцию отброшенной связи: $S_1 = 14,14$ кН. Стержень растянут.

6. Из уравнения (4.38) моментов относительно неподвижного шарнира всех сил основной системы, включая найденные реакции отброшенных связей, определяем усилие в опорном стержне 2. Получаем $S_2 = -32$ кН. Стержень сжат. Метод деформаций и метод сил дают одинаковый результат.

Задача 59. Дана стержневая конструкция, состоящая из жесткого недеформируемого бруса OC , закрепленного одним концом на неподвижном шарнире O , и трех упругих стержней одинаковой жесткости,

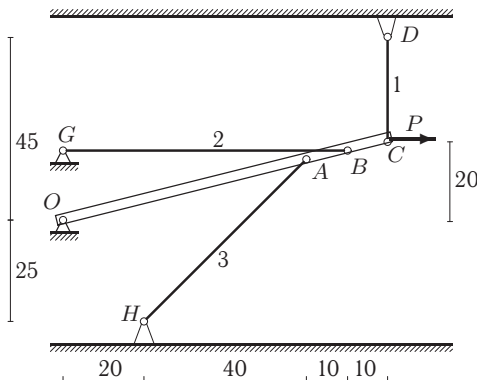


Рис. 236

прикрепленных к брусу шарнирно (рис. 236). Стержень CD вертикальный, BG — горизонтальный. Вес стержней и деформации опор не учитывать. К свободному концу бруса приложена горизонтальная сила

$P = 16$ кН. Размер бруса дан в проекциях (в сантиметрах). Найти усилия в стержнях.

Решение

Способ 1. Уравнения совместности деформаций.

Система статически неопределимая. Для удержания бруса в равновесии, очевидно, достаточно неподвижного шарнира и только одного опорного стержня или трех стержней (если они, конечно, не пересекаются в одной точке) без опоры. Следовательно, система содержит две дополнительные неизвестные величины. Методами теоретической механики, где тела предполагаются абсолютно твердыми, эту задачу решить нельзя. Если предположить, что стержни деформируются, то для определения дополнительных неизвестных помимо трех уравнений равновесия можно составить два дополнительных уравнения. Этими уравнениями являются уравнения совместности деформаций. Совместность деформаций обеспечивает жесткость бруса. Стержни CD , BG и AH обозначим номерами 1, 2, 3 соответственно.

1. Находим два геометрических соотношения между абсолютными деформациями стержней Δl_1 , Δl_2 , Δl_3 .

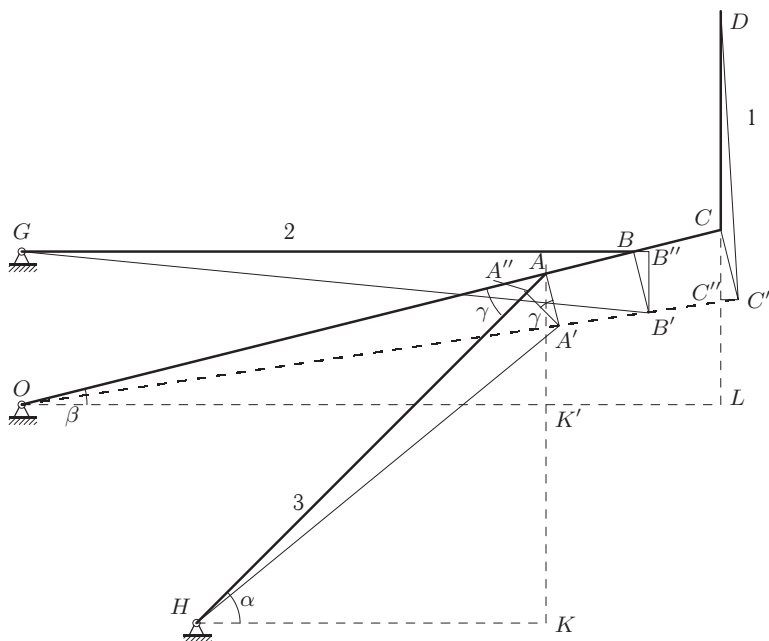


Рис. 237

Рассмотрим малое отклонение OC' бруса от недеформированного состояния OC — поворот вокруг опоры по часовой стрелке¹ (рис. 237). Шарниры на брус 1, 2, 3 обозначим буквами A, B, C . После малой деформации эти шарниры сместятся вместе с бруском и примут положения A', B', C' . В силу малости деформаций дуги, по которым переместятся точки, заменим перпендикулярами к первоначальному положению бруса. Таким образом, отрезки AA', BB', CC' перпендикулярны OC . Новые положения опорных стержней: DC', GB', HA' . Стержни 1 и 2 удлинятся, стержень 3 укоротится.

Опуская перпендикуляры из новых положений концов стержней на их первоначальные направления, получим удлинения $\Delta l_1 = CC''$ (рис. 238), $\Delta l_2 = BB''$ (рис. 239), $\Delta l_3 = -AA''$ (рис. 237).

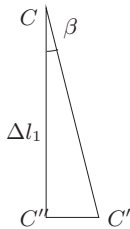


Рис. 238

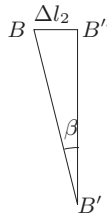


Рис. 239

Минус в последнем случае объясняется сжатием стержня 3 (удлинение меньше нуля). Для перемещений концов стержней справедлива очевидная пропорция

$$\frac{AA'}{60} = \frac{BB'}{70} = \frac{CC'}{80}. \quad (4.42)$$

Кроме того, из прямоугольных треугольников имеем

$$AA' = -\Delta l_3 / \sin \gamma, \quad BB' = \Delta l_2 / \sin \beta, \quad CC' = \Delta l_1 / \cos \beta,$$

где

$$\begin{aligned} \sin \beta &= CL/OC = 20/\sqrt{20^2 + 80^2}, \quad \operatorname{tg} \beta = 1/4, \\ \sin \alpha &= AK/O_3A = 40/\sqrt{40^2 + 40^2}, \quad \gamma = \alpha - \beta. \end{aligned} \quad (4.43)$$

Пропорция (4.42) примет вид

$$-\frac{\Delta l_3}{60 \sin \gamma} = \frac{\Delta l_2}{70 \sin \beta} = \frac{\Delta l_1}{80 \cos \beta}. \quad (4.44)$$

2. Используем закон Гука $\Delta l_k = S_k l_k / (EF)$, где S_k — усилие в k -м стержне, а l_k — его длина. Жесткость стержней EF одинакова.

¹Выбор направления поворота бруса произволен.

С учетом найденных геометрических соотношений получаем два дополнительных уравнения для усилий S_1 , S_2 , S_3 .

Записываем закон Гука:

$$\Delta l_1 = \frac{S_1 l_1}{EF}, \quad \Delta l_2 = \frac{S_2 l_2}{EF}, \quad \Delta l_3 = \frac{S_3 l_3}{EF}. \quad (4.45)$$

Длины стержней определяем из рис. 237: $l_1 = 25$ см, $l_2 = 70$ см, $l_3 = HA = \sqrt{HK^2 + AK^2} = \sqrt{HK^2 + (KK' + OK' \operatorname{tg} \beta)^2} = \sqrt{40^2 + (25 + 60 \cdot (1/4))^2} = 40\sqrt{2} = 56,57$ см.

Подставляем (4.45) в (4.44). Получаем

$$-\frac{S_3 l_3}{60 \sin \gamma} = \frac{S_2 l_2}{70 \sin \beta} = \frac{S_1 l_1}{80 \cos \beta}. \quad (4.46)$$

Выражаем два усилия через третье:

$$S_2 = \frac{70 S_1 l_1 \sin \beta}{80 l_2 \cos \beta} = 0,228 S_1, \quad S_3 = -\frac{60 S_1 l_1 \sin \gamma}{80 l_3 \cos \beta} = -0,064 S_1. \quad (4.47)$$

3. Записываем уравнение статического равновесия всех сил, действующих на брус, включая неизвестные реакции стержней. Для того чтобы не включать в решение неизвестные реакции неподвижного шарнира O (рис. 240), составляем относительно этого шарнира уравнение моментов:

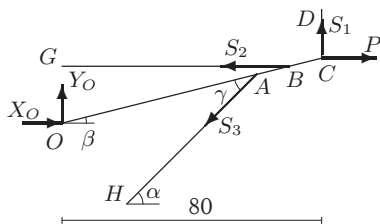


Рис. 240

$$S_1 \cdot 80 + S_2 \cdot OB \sin \beta - S_3 \cdot OA \sin \gamma - P \cdot OC \sin \beta = 0, \quad (4.48)$$

где

$$OA = 60 / \cos \beta, \quad OB = 70 / \cos \beta, \quad OC = 80 / \cos \beta. \quad (4.49)$$

4. Решаем систему трех уравнений, одно из которых — уравнение статики (4.48), а два других — соотношения (4.47), найденные из условия совместности деформаций. Находим усилия в стержнях. Подставляем в (4.48) соотношения (4.47). После упрощений получаем

$$86,96 S_1 - 20 P = 0,$$

откуда находим усилие $S_1 = 0,23 P = 3,68$ кН, и с помощью (4.47) два других: $S_2 = 0,018 P = 0,29$ кН, $S_3 = -0,04 P = -0,65$ кН.

Способ 2. Метод сил

1. Выбираем основную систему. Освобождаем от связей стержни 1 и 2 в местах их крепления к шарнирам 6 и 4. Усилия в этих стержнях

назначаем неизвестными метода сил. Основная система представляет собой статически определимую систему, состоящую из шарнирно закрепленного тела (наклонного бруса) с одним опорным стержнем 3, на которую действуют внешние силы и неизвестные усилия S_1, S_2 .

2. Находим усилие в опорном стержне 3 основной системы от действия внешней нагрузки P . Записываем уравнение моментов сил, приложенных к брусу, относительно неподвижного шарнира O (рис. 241). Стержни 1 и 2 остаются ненагруженными. При построении основной системы их освободили от связей в точках крепления. Приравняем сумму моментов нулю: $-s_{3,P}OA \sin \gamma - P \cdot OC \sin \beta = 0$. Это уравнение также следует из (4.48) при $S_1 = S_2 = 0$. Подставляем сюда геометрические данные (4.43), (4.49) и $P = 16$ кН и получаем $s_{3,P} = -10,06$ кН.

3. Находим усилие в опорном стержне 3 основной системы от единичной нагрузки вдоль реакции лишней связи S_1 .

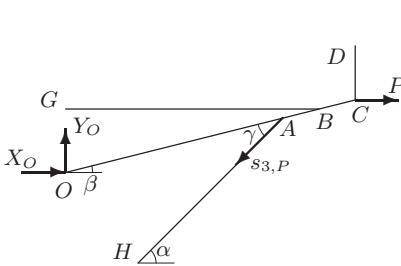


Рис. 241

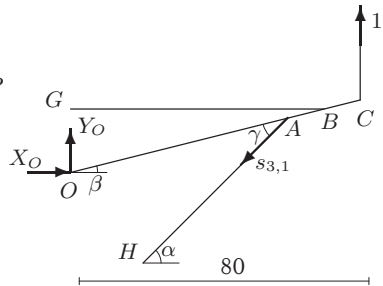


Рис. 242

Из уравнения моментов относительно опоры O (рис. 242) следует $1 \cdot 80 - s_{3,1}OA \sin \gamma = 0$. Решаем уравнение: $s_{3,1} = 2,51$.

Усилие $s_{3,2}$ в стержне 3 от единичной нагрузки, приложенной к опоре стержня 2, находим из уравнения моментов (рис. 243): $1 \times \times OB \sin \beta - s_{3,2}OA \sin \gamma = 0$.

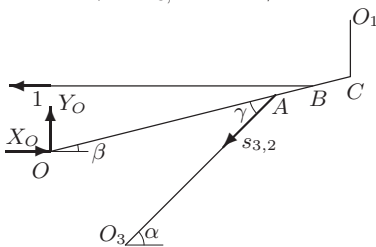


Рис. 243

Получим $s_{3,2} = 0,55$. Для вычисления коэффициентов канонических уравнений все результаты удобно занести в таблицу

k	l_k	$s_{k,1}$	$s_{k,2}$	$s_{k,P}$
1	25,00	1,00	0	0
2	70,00	0	1,00	0
3	56,57	2,51	0,55	-10,06

4. По формулам Максвелла–Мора (4.39), (4.40) определяем коэффициенты канонических уравнений δ_{ij} и Δ_{iP} .

$$\begin{aligned}\delta_{11} &= \sum_k \frac{l_k s_{k,1}^2}{EF} = \frac{l_1 \cdot 1^2 + l_3 s_{3,1}^2}{EF} = \frac{382,57}{EF}, \\ \delta_{12} &= \sum_k \frac{l_k s_{k,1} s_{k,2}}{EF} = \frac{l_3 s_{3,1} s_{3,2}}{EF} = \frac{78,22}{EF}, \\ \delta_{22} &= \sum_k \frac{l_k s_{k,2}^2}{EF} = \frac{l_2 \cdot 1^2 + l_3 s_{3,2}^2}{EF} = \frac{87,11}{EF}, \\ \Delta_{1P} &= \sum_k \frac{l_k s_{k,1} s_{k,P}}{EF} = \frac{l_3 s_{3,1} s_{3,P}}{EF} = -\frac{1430,28}{EF}, \\ \Delta_{2P} &= \sum_k \frac{l_k s_{k,2} s_{k,P}}{EF} = \frac{l_3 s_{3,2} s_{3,P}}{EF} = -\frac{312,87}{EF}.\end{aligned}$$

5. Составляем и решаем систему (4.41) канонических уравнений метода сил. Для двух неизвестных она имеет вид

$$\begin{aligned}\delta_{11}S_1 + \delta_{12}S_2 + \Delta_{1P} &= 0, \\ \delta_{21}S_1 + \delta_{22}S_2 + \Delta_{2P} &= 0.\end{aligned}\quad (4.50)$$

С учетом найденных значений коэффициентов получаем систему

$$\begin{aligned}382,57 S_1 + 78,22 S_2 - 1430,28 &= 0, \\ 78,22 S_1 + 87,11 S_2 - 312,87 &= 0.\end{aligned}\quad (4.51)$$

Решаем систему канонических уравнений (4.51). Находим неизвестные метода сил — усилия в стержнях: $S_1 = 3,68$ кН, $S_2 = 0,29$ кН.

6. Из уравнения моментов (4.48) относительно неподвижного шарнира всех сил основной системы, включая найденные усилия, определяем усилие в опорном стержне 3. Получаем $S_3 = -0,65$ кН.

В качестве проверки усилие S_3 можно найти из выражения

$$\begin{aligned}S_3 &= s_{3,1}S_1 + s_{3,2}S_2 + s_{3,P} = \\ &= 2,51 \cdot 3,68 + 0,55 \cdot 0,29 - 10,06 = -0,65 \text{ кН}.\end{aligned}\quad (4.52)$$

Последнюю формулу можно понимать как выражение суммы усилий в стержне 3 от трех отдельных факторов: усилий в первом и втором стержне и от нагрузки P . При этом $s_{3,1}$ — это усилие в третьем стержне от действия единичной нагрузки в первом стержне, а так как S_1 — это действительное усилие в стержне 1, то произведение $s_{3,1}S_1$ есть усилие в стержне 3 от действия нагрузки в стержне 1. Аналогично понимаются и другие слагаемые суммы (4.52). Так как $S_3 < 0$, то третий стержень сжат. Два другие: $S_1 > 0$, $S_2 > 0$ — растянуты. Усилия совпадают с ответами, полученными по первому методу (с. 173).

Maple–программа, реализующая метод сил, дана на с. 402.

4.7.2. Свободное твердое тело на упругих стержнях

Постановка задачи. Жесткая пластина закреплена $n > 3$ упругими стержнями. К пластине приложены известные нагрузки. Найти усилия в стержнях.

Так как пластина имеет $n - 3$ лишний опорный стержень, то для решения задачи, помимо трех уравнений статики, необходимо привлечь еще $n - 3$ дополнительное уравнение — уравнение совместности деформаций. Как и в предыдущей задаче, возможны два способа решения: метод сил и метод деформаций. В этой задаче, с $n - 3$ степенью статической неопределенности, метод сил в данном случае не отличается от рассмотренного. Используя метод Максвелла–Мора, требуется получить коэффициенты δ_{ij} и Δ_{ip} .

Геометрический же метод отыскания уравнений совместности деформаций здесь трудно применим. Если в предыдущей задаче с шарнирным телом картина деформации сводится лишь к повороту и совместность деформаций следует из простого подобия треугольников, то в данном случае при плоском движении пластины, вызванном деформацией стержней, получить аналогичный результат затруднительно. Предлагаем поэтому универсальный *кинематический метод*. Этот метод свободен от утомительных геометрических построений треугольников, с обязательными в таких случаях допущениями о замене малых дуг отрезками прямых, визуальным определением знака деформации и всего того, что делает в общем-то несложную задачу одной из самых запутанных в курсе сопротивления материалов (см. задачу 59). Для решения задачи потребуются привлечь кинематические соотношения для скоростей точек тела при плоском движении.

План решения

1. Освобождаем пластину от связей (упругих стержней), заменяя их действие реакциями, направленными по стержням (от опоры на пластине к стержню). Составляем три уравнения равновесия для всех сил, действующих на пластину.

2. Задаем пластине возможное перемещение, определяемое тремя неопределенными параметрами — двумя компонентами скорости какой-либо точки пластины и угловой скоростью поворота.

3. Выражаем скорости шарниров стержней на пластине через заданные параметры.

4. Находим проекции скоростей шарниров на направления стержней. Для стержней выбираем условное направление, например от точки на пластине к опорной точке.

5. Исключая из n проекций три заданных параметра (две скорости и угловая скорость), получаем $(n - 3)$ соотношения между скоростями.

6. Выражаем скорости шарниров через продольные деформации.

7. Используем закон Гука $\Delta l_k = S_k l_k / (E_k F_k)$, где S_k — усилие в k -м стержне, $E_k F_k$ — жесткость стержня, а l_k — его длина. Получаем $n - 3$ дополнительное уравнение для усилий S_k , $k = 1, \dots, n$.

8. Решаем систему n уравнений, три из которых — уравнения статики, а $(n - 3)$ — соотношения совместности деформаций. Находим усилия в стержнях.

Задача 60. Однородная пластина весом $P = 5$ Н удерживается в вертикальной плоскости четырьмя стержнями одинаковой жесткости (рис. 244). На пластину действует момент $m = 0,1$ Нм. Размеры даны в сантиметрах. Найти усилия в стержнях.

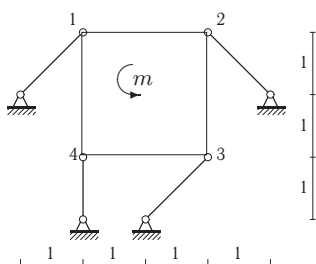


Рис. 244

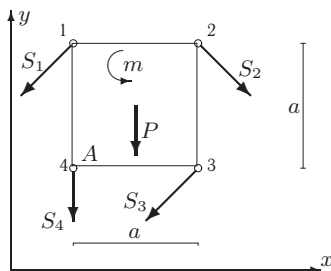


Рис. 245

Решение

1. Освобождаем пластину от связей (упругих стержней), заменяя их действие реакциями, направленными по линиям стержней om пластины к опорам (рис. 245). Прикладываем силу тяжести пластины P к ее центру.

Составляем три уравнения равновесия пластины. В эти уравнения входят все силы, действующие на пластину, включая реакции стержней. Вводим систему координат. Сумму моментов берем относительно произвольной точки, например шарнира 4 на пластине, который обозначим как A :

$$\begin{aligned} \sum X_i &= -S_1 \cos 45^\circ + S_2 \cos 45^\circ - S_3 \cos 45^\circ = 0, \\ \sum Y_i &= -S_1 \cos 45^\circ - S_2 \cos 45^\circ - S_3 \cos 45^\circ - P - S_4 = 0, \\ \sum M_A &= S_1 a \cos 45^\circ - S_2 a \sqrt{2} - S_3 a \cos 45^\circ - P \frac{a}{2} + m = 0, \end{aligned} \quad (4.53)$$

где $a = 2$ см.

2. Задаем пластине возможное перемещение, определяемое тремя неопределенными параметрами — двумя компонентами v_{4x} , v_{4y} скорости точки 4 пластины и угловой скоростью поворота ω_z .

3. Выражаем скорости точек 1, 2, 3 на пластине через заданные параметры. Используя формулу (2.5), с. 55, где в качестве полюса берем

шарнир 4, получаем

$$\vec{v}_1 = \vec{v}_4 + \vec{v}_{1,4} = \vec{v}_4 + \vec{\omega} \times \vec{r}_1,$$

$$\vec{v}_2 = \vec{v}_4 + \vec{v}_{2,4} = \vec{v}_4 + \vec{\omega} \times \vec{r}_2,$$

$$\vec{v}_3 = \vec{v}_4 + \vec{v}_{3,4} = \vec{v}_4 + \vec{\omega} \times \vec{r}_3.$$

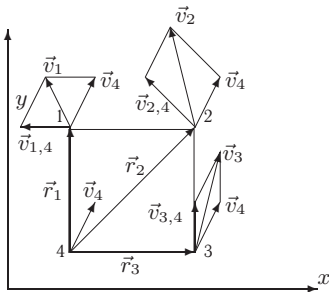


Рис. 246

В проекциях на оси x и y получаем компоненты скоростей

$$v_{1x} = v_{4x} - \omega_z a,$$

$$v_{1y} = v_{4y},$$

$$v_{2x} = v_{4x} - \omega_z a,$$

$$v_{2y} = v_{4y} + \omega_z a,$$

$$v_{3x} = v_{4x},$$

$$v_{3y} = v_{4y} + \omega_z a.$$

4. Находим проекции скоростей шарниров на направления стержней (от точки на пластине к опорной точке) по формуле скалярного произведения $v_{k\tau} = \vec{n}_k \cdot \vec{v}_k$, где \vec{n}_k — направляющие косинусы единичных векторов, направленных по стержням. Имеем следующие значения: $\vec{n}_1 = \vec{n}_3 = [-\cos 45^\circ, -\cos 45^\circ]$, $\vec{n}_2 = [\cos 45^\circ, -\cos 45^\circ]$, $\vec{n}_4 = [0, -1]$. Отсюда получаем

$$v_{1\tau} = -v_{1x} \cos 45^\circ - v_{1y} \cos 45^\circ = -\sqrt{2}/2(v_{4x} + v_{4y} - \omega_z a),$$

$$v_{2\tau} = v_{2x} \cos 45^\circ - v_{2y} \cos 45^\circ = \sqrt{2}/2(v_{4x} - v_{4y} - 2\omega_z a),$$

$$v_{3\tau} = -v_{3x} \cos 45^\circ - v_{3y} \cos 45^\circ = -\sqrt{2}/2(v_{4x} + v_{4y} + \omega_z a),$$

$$v_{4\tau} = -v_{4y}.$$

5. Исключая из четырех проекций три заданных параметра v_{4x} , v_{4y} и ω_z , получаем соотношение между скоростями

$$3v_{1\tau} + 2v_{2\tau} - v_{3\tau} - 2\sqrt{2}v_{4\tau} = 0.$$

Процедуру исключения лучше поручить системе Maple, в которой предусмотрен удобный оператор `eliminate`. Программа может иметь, например, следующий вид

```
> eq1:=v1=-sqrt(2)/2*(vx+vy-wa):
> eq2:=v2= sqrt(2)/2*(vx-vy-2*wa):
> eq3:=v3=-sqrt(2)/2*(vx+vy+wa):
> eq4:=v4=-vy:
> eliminate({eq1,eq2,eq3,eq4},{vx,vy,wa});
```

$$\left\{ \begin{aligned} & \{vy = -v_4, wa = -\frac{\sqrt{2}v_2}{3} - \frac{\sqrt{2}v_3}{3} + \frac{2v_4}{3}, vx = -\frac{2\sqrt{2}v_3}{3} + \frac{v_4}{3} + \frac{\sqrt{2}v_2}{3}\}, \\ & \{-v_3 + 2v_2 - 2\sqrt{2}v_4 + 3v_1\} \end{aligned} \right\}$$

6. Выражаем скорости шарниров через продольные деформации. Для малых промежутков времени $v_{k\tau} = -\Delta l_k / \Delta t$. Минус в соотношении возник из-за того, что орты стержней взяты по направлению сжатия, а правило знаков для усилий предполагает, что сжимающие усилия меньше нуля. Отсюда следует уравнение совместности деформаций

$$3\Delta l_1 + 2\Delta l_2 - \Delta l_3 - 2\sqrt{2}\Delta l_4 = 0.$$

7. Используем закон Гука

$$\Delta l_k = S_k l_k / (E_k F_k). \quad (4.54)$$

По условию задачи все стержни имеют одинаковую жесткость $E_k F_k = EF$. Длины стержней (в сантиметрах): $l_1 = l_2 = l_3 = \sqrt{2}$, $l_4 = 1$. В результате получаем искомое уравнение для усилий

$$3S_1 + 2S_2 - S_3 - 2S_4 = 0.$$

8. Это уравнение вместе с (4.53) образует систему, решение которой (в ньютонах) имеет вид

$$S_1 = -2,215, S_2 = -0,298, S_3 = 1,917, S_4 = -4,578.$$

Замечание. Если в условии задачи сказано, что деформируемые стержни нагреваются, то для учета температурного фактора в уравнениях (4.54) достаточно добавить дополнительные слагаемые:

$$\Delta l_k = S_k l_k / (E_k F_k) + \Delta T_k l_k \alpha,$$

где α — коэффициент температурного расширения, ΔT_k — температура нагрева стержня в градусах. Для латуни, например, $\alpha = 19 \times 10^{-6}$ град $^{-1}$.

Maple-программа, реализующая метод сил, дана на с. 405.

4.8. Изгиб балки

Задача об изгибе балки является самой распространенной в курсах сопротивления материалов и, вероятно, одной из наиболее важных. Она решается и в техникумах, и в университетах. В зависимости от вида и величины внешних нагрузок, типов опор и поставленных вопросов существует множество вариантов условий. Нет такого учебника или задачника, где бы задача о балке не была упомянута. Существуют и

программы для ее решения¹. Популярность этой задачи связана с ее актуальностью. В большинстве технологических процессов, в машиностроении или строительстве присутствуют элементы, достаточно точно моделируемые изгибаемой балкой. Важно определить ее прочность, деформации и прогибы. Масштабы здесь бывают самые различные — от изгибаемого элемента манипулятора наноробота размером в несколько миллиметров до балки всякого моста пролетом в сотни метров. Балкой можно моделировать волокна в композитах и крыло самолета, корпус морского лайнера и антенну спутника. Все зависит, конечно, от необходимой точности решения.

Несмотря на разнообразие постановок задачи и большое число их вариантов, существует устоявшаяся методика ее решения. Для различных курсов она может в деталях отличаться правилом знаков или обозначениями, однако эти различия несущественны, надо только выбрать какое-то одно правило, одну методику и руководствоваться при решении одним учебником, желательно наиболее полным².

В простейшей постановке требуется определить моменты и перерезывающие силы в балке как функции продольной координаты — построить эпюры (графики) этих величин.

Если рассмотреть равновесие выделенной двумя сечениями части балки, то действие отброшенных частей, согласно аксиоме о связях, надо заменить их реакциями. В общем случае в каждом сечении должно быть три реакции: перерезывающая сила, нормальная сила и реактивный момент. Обычно в задаче о балке предполагается, что внешние силы вертикальные, поэтому нормальных реакций в сечении не возникает, и эти реакции при решении не изображают.

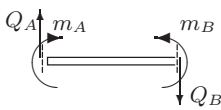


Рис. 247



Рис. 248

В курсах сопротивления материалов, предназначенных для машиностроительных специальностей, принято считать, что моменты изгибаю-

¹В системе Mathcad решение задачи о балке дано в книгах [36], [45]. В Интернете свободно распространяется, например, программа В. Перевалова <http://vuz.exponenta.ru/PDF/EPURA.swf> и автора <http://vuz.exponenta.ru/balka2.rar>. Программа расчета балки в системе Maple 6 содержится в книге А. В. Матрoсова [42].

²Академик Седов Леонид Иванович говорил, что научиться чему-либо можно только по толстым книгам. Широко известная книга самого Седова Л. И. по механике сплошных сред состоит из двух томов [53].

щие балку выпуклостью вниз, положительные, в противном случае они отрицательные¹. Для положительных перерезывающих сил Q принято правило, согласно которому они стремятся повернуть балку по часовой стрелке (рис. 247).

При этом оказывается, что знак моментов согласуется с принятым в математике знаком кривизны плоской кривой. Кривая с выпуклостью вверх имеет отрицательную кривизну, вниз — положительную ("правило дождя", рис. 248).

Для определения моментов и сил в сечении можно поступить как в разделе статики теоретической механики — выполнить в каком-либо месте сечение балки, выбрать левую или правую ее часть (в зависимости от числа сил) и рассмотреть равновесие этой части. Условием равновесия являются уравнения равновесия всех внешних сил, приложенных к рассматриваемой части балки, включая и реакции связей, соответствующих искомым моментам и силам. Уравнения имеют, например, вид

$$\sum F_{ky} + Q_A = 0, \quad \sum M_{kz} + m_A = 0, \quad (4.55)$$

где F_{ky} и M_k — внешние силы и моменты этих сил относительно места сечения. Решая систему, можно получить значения моментов $M_A(x)$ и перерезывающих сил $Q_A(x)$ и построить графики зависимостей этих величин от координаты x места сечения. Вводится правая система координат. Ось x направляется вдоль балки, y — вверх, z — перпендикулярно плоскости чертежа на наблюдателя. Положительные значения моментов и перерезывающих сил откладываем по оси y (вверх). Таким образом, эпюра моментов строится на сжатом волокне.

Однако в сопротивлении материалов часто применяют немного более простой метод. Реакции связей в сечении не изображают и сразу выписывают выражения для моментов и сил, суммируя их с учетом принятого правила знаков. Согласно (4.55), суммы отличаются от реакций знаком. Поэтому положительные направления для суммируемых сил и моментов по отношению к направлениям реакций обратные. Эти направления отметим в сечении стрелками и дугами со значками Q^+ и m^+ . Поясним все на примерах.

Будем рассматривать балки постоянного сечения из однородного материала. Таким образом, жесткость балки на изгиб EJ принимается постоянной.

Постановка задачи. Построить эпюры моментов и перерезывающих сил в балке, находящейся под действием вертикальных сосредоточенных и распределенных нагрузок и моментов.

¹В строительных и архитектурных вузах принято обратное правило знаков для моментов.

План решения

1. Определяем реакции опор балки. Для этого отбрасываем связи и заменяем их действие двумя вертикальными реакциями R_A и R_B . Горизонтальная реакция одной из связей равна нулю, так как все нагрузки вертикальные. Составляем два уравнения моментов всех сил относительно точек A и B . Решаем уравнения. Определяем R_A и R_B .

2. Выполняем проверку решения. Составляем уравнение проекции на вертикальную ось. Если уравнение удовлетворяется тождественно, то реакции найдены верно¹.

3. Определяем границы участков балки длиной L для построения эпюр. Границами участков являются концы балки, опоры, точки приложения сосредоточенных сил и моментов, начальные и конечные точки распределенной нагрузки.

4. Выбираем начало координат на левом конце балки². Разрезаем участок балки в некоторой произвольной точке. Отбрасываем правую часть балки, заменяя ее действие реакциями — моментом M и поперечной силой Q . Реакции на рисунке не изображаем, а указываем правило знаков для их определения. Суммируем вертикальные силы, приложенные к рассматриваемой части балки, получаем значение поперечной силы в сечении. Суммируем моменты всех сил (с учетом знака) относительно сечения, получаем момент. Построение повторяем для всех участков³.

5. Используя найденные выражения функций на отдельных участках, строим искомые зависимости для всей балки при $0 \leq x \leq L$.

6. Проверяем выполнение дифференциальной зависимости

$$Q_x = dm_z/dx.$$

В частности, в тех сечениях балки, где эпюра моментов имеет максимум или минимум и $dm_z/dx = 0$, там поперечная сила обращается в ноль. Если $Q_y < 0$, то на этом участке момент убывает, и, наоборот, при $Q_y > 0$ момент растет.

Задача 61. Построить эпюры моментов и перерезывающих сил в балке, находящейся под действием вертикальной сосредоточенной нагрузки $P = 5$ кН, распределенной силы $q = 4$ кН/м и момента $m = 8$ кНм (рис. 249). Размеры даны в метрах.

¹ Допускается погрешность, сопоставимая с точностью вычислений. Так, если вычисления производились с точностью до третьего знака после запятой, то невязка в уравнении проверки, меньшая по модулю, чем 0,0005, считается допустимой.

² Можно поместить начало координат на любом конце участка и направить ось x горизонтально в произвольном направлении, однако мы не рекомендуем использовать эту практику, особенно при отсутствии опыта.

³ Для последнего участка можно отбросить левую часть.

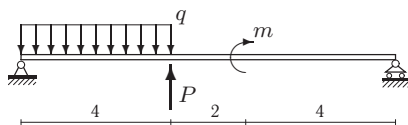


Рис. 249

Решение

1. Определяем реакции опор балки. Для этого отбрасываем связи и заменяем их действие двумя вертикальными реакциями R_A и R_B (рис. 250). Равномерно распределенную нагрузку заменяем равнодействующей — сосредоточенной силой $4q$, приложенной в середине участка.

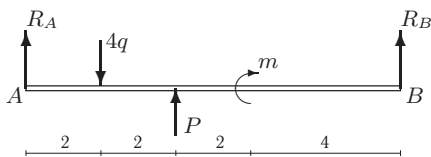


Рис. 250

Составляем два уравнения моментов относительно точек A и B :

$$\begin{aligned}\sum M_A &= R_B \cdot 10 + P \cdot 4 - q \cdot 4 \cdot 2 - m = 0, \\ \sum M_B &= -R_A \cdot 10 - P \cdot 6 + q \cdot 4 \cdot 8 - m = 0.\end{aligned}$$

Решаем уравнения. Определяем $R_A = 9$ кН и $R_B = 2$ кН.

2. Выполняем проверку решения. Составляем уравнение проекции на вертикальную ось:

$$R_A + R_B + P - q \cdot 4 = 9 + 2 + 5 - 16 = 0.$$

Уравнение удовлетворяется тождественно, реакции найдены верно.

3. Определяем границы участков балки для построения эпюр. Балка имеет три участка I, II, III (рис. 251). Границами участков являются концы балки, опоры A и B .

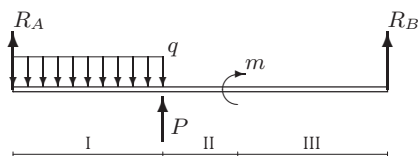


Рис. 251

4. Выбираем начало координат на левом конце балки. Разрезаем участок I рамы в некоторой произвольной средней точке. Для первого

участка $0 \leq x \leq 4$. Отбрасываем правую часть балки, заменяя ее действие реакциями — моментом $M(x)$ и поперечной силой $Q(x)$ (на рис. 252 отброшенную часть не изображаем). Из условия равновесия оставшейся части определяем реакции $M(x)$ и $Q(x)$ как функции x . В месте сечения показываем не реакции, а их положительные направления в выражении для момента и перерезывающей силы.

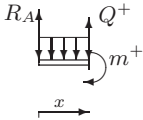


Рис. 252

Выписываем значения момента и перерезывающей силы на участке I при $0 \leq x \leq 4$, руководствуясь принятым правилом знаков (рис. 252). К этому участку приложена реакция R_A и равномерно распределенная нагрузка. Равнодействующая распределенной нагрузки равна qx и имеет плечо $x/2$ относительно сечения. Получаем следующие выражения: $M(x) = R_A x - qx^2/2$, $Q(x) = R_A - qx$.

Для второго участка $4 \leq x \leq 6$ (рис. 253). Распределенную нагрузку q длиной 4 м заменяем равнодействующей $4q$, приложенной в центре ее участка (рис. 254), на расстоянии $x - 2$ от сечения.

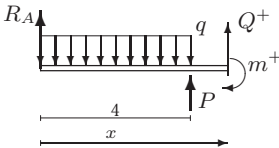


Рис. 253

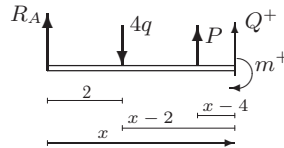


Рис. 254

Выписываем значения момента и перерезывающей силы на участке II при $4 \leq x \leq 6$, руководствуясь принятым правилом знаков. Моменты сил, приложенных к участку балки, берем относительно сечения, т. е. правого конца, со знаком +, если поворот силы виден по часовой стрелке (сила создает изгиб балки выпуклостью вниз).

$$M(x) = R_A x - 4q(x - 2) + P(x - 4),$$

$$Q(x) = R_A - 4q + P.$$

Для третьего участка $6 \leq x \leq 10$. Если на первых двух участках рассматривалось равновесие левой отсеченной части, то на третьем участке правая часть содержит только реакцию опоры R_B , и выражения для момента и перерезывающей силы получаются короче. Выделим правую часть балки, рассеченной по произвольному месту третьего участка. Начало координат лучше не менять — расстояние x , как и на участках I и II, берется от левой опоры (рис. 255).

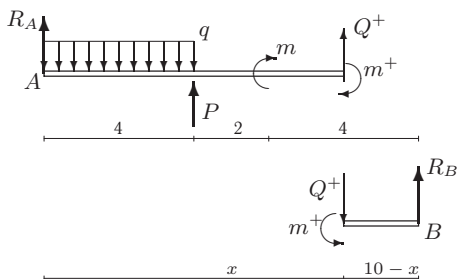


Рис. 255

Выписываем значения момента и перерезывающей силы на участке III при $6 \leq x \leq 10$, руководствуясь принятым правилом знаков.

$$M(x) = R_B(10 - x),$$

$$Q(x) = -R_B.$$

5. Используя найденные выражения функций на отдельных участках, строим искомые зависимости для всей балки при $0 \leq x \leq L$. Вычисляем ординаты эпюр на границах участков. В тех точках, где эпюры имеют разрывы (результат действия сосредоточенных нагрузок), записываем значения на конце одного участка и в начале другого. Участки отмечаем римской цифрой в верхнем индексе:

$$M(0) = 0, \quad Q(0) = 9 \text{ кН}, \quad M(4) = 4 \text{ кНм}, \quad Q^I(4) = -7 \text{ кН}, \\ Q^{II}(4) = -2 \text{ кН}, \quad Q(6) = -2 \text{ кН}, \quad M^{II}(6) = 0, \quad M^{III}(6) = 8 \text{ кНм}.$$

Для большей наглядности эпюры рисуем одну под другой, совмещая также с рисунком балки с нагрузкой (рис. 256).

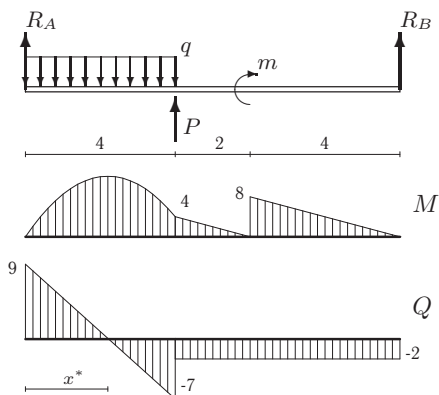


Рис. 256

6. Проверяем выполнение дифференциальной зависимости $Q(x) = dM(x)/dx$. На эпюре Q есть участок I, где перерезывающая сила Q обращается в нуль. Согласно дифференциальной зависимости в этом месте момент имеет максимум. Находим значение $x = x^*$, где $Q(x) = 0$. Для этого приравняем нулю выражение $Q(x)$: $Q(x) = R_A - qx^* = 0$, откуда находим $x^* = 2,25$ м. Вычисляем соответствующее значение момента:

$$M(x^*) = R_A x^* - qx^{*2}/2 = 10,125 \text{ кНм.}$$

Заметим также, что в тех точках, где на балку действуют сосредоточенные силы, эпюра моментов имеет точку излома, а эпюра перерезывающих сил — скачок. Скачок на эпюре моментов соответствует сосредоточенному моменту.

Замечание. Значение момента в середине участка длиной l , где действует равномерно распределенная нагрузка q , можно вычислить по формуле $(M_1 + M_2)/2 + ql^2/8$, где M_1 и M_2 — значения эпюры на концах участка.

Для расчета балки на с. 424 и 450 приведены две различные Maple-программы. Первая программа 40 написана специально для балки на двух опорах. Во второй программе 45 балка рассматривается как частный случай многопролетной разрезной балки. Файл данных `balka.txt` для расчета балки по программе 41 описан на с. 439.

Задача 62. Многопролетная балка, состоящая из четырех частей, опирается на четыре подвижные и одну неподвижную опоры. На балку действуют момент $m = 35$ кНм, вертикальные силы $P_1 = 6$ кН и $P_2 = 5$ кН, равномерно распределенные нагрузки $q_1 = 2$ кН/м, $q_2 = 4$ кН/м (рис. 257). Размеры даны в метрах. Определить реакции опор и построить эпюры моментов и перерезывающих сил.

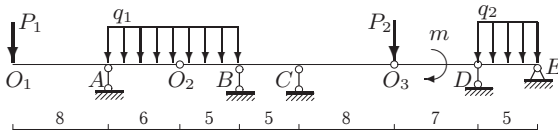


Рис. 257

Решение

1. Разбиваем балку по шарнирам на отдельные части. Шарниры заменяем вертикальными реакциями Y_1, Y_2, Y_3 , опоры — реакциями опор $R_i, i = 1, \dots, 5$. Нагрузка на балку вертикальная, очевидно, горизонтальные реакции при этом будут равны нулю. Для упрощения рисунков внутренние и внешние горизонтальные реакции не изображаем. Нагрузку P_2 , приложенную к шарниру O_3 , относим произвольно к

левой или правой части, более того, такую силу можно даже поделить на части и отнести часть к одной стороне, другую часть — к другой. Это же относится и к реакции R_4 .

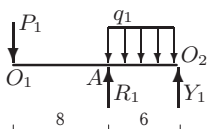


Рис. 258

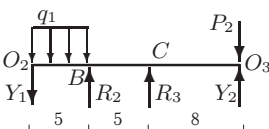


Рис. 259

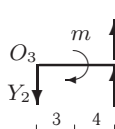


Рис. 260

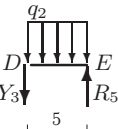


Рис. 261

Составляем уравнения равновесия части $O_1 - O_2$ (рис. 258). Равнодействующая распределенной нагрузки равна $6q_1$ и приложена в середине участка AO_2 .

$$\begin{aligned}\sum M_A &= Y_1 \cdot 6 - q_1 \cdot 6 \cdot 3 + P_1 \cdot 8 = 0, \\ \sum Y_i &= Y_1 + R_1 - q_1 \cdot 6 - P_1 = 0.\end{aligned}\quad (4.56)$$

Составляем уравнения равновесия части $O_2 - O_3$ (рис. 259):

$$\begin{aligned}\sum M_B &= Y_1 \cdot 5 + q_1 \cdot 5 \cdot 5/2 + R_3 \cdot 5 + Y_2 \cdot 13 - P_2 \cdot 13 = 0, \\ \sum Y_i &= -Y_1 + R_2 + R_3 + Y_2 - q_1 \cdot 5 - P_2 = 0.\end{aligned}\quad (4.57)$$

Составляем уравнения равновесия части $O_3 - D$ (рис. 260):

$$\begin{aligned}\sum M_{O_3} &= Y_3 \cdot 7 + R_4 \cdot 7 - m = 0, \\ \sum Y_i &= -Y_2 + R_4 + Y_3 = 0.\end{aligned}\quad (4.58)$$

Составляем уравнения равновесия части $D - E$ (рис. 261):

$$\begin{aligned}\sum M_D &= R_5 \cdot 5 - q_2 \cdot 5 \cdot 5/2 = 0, \\ \sum Y_i &= -Y_3 + R_5 - q_2 \cdot 5 = 0.\end{aligned}\quad (4.59)$$

2. Решаем систему восьми уравнений (4.56–4.59) с восемью неизвестными. Получаем следующие реакции: $Y_1 = -2$ кН, $Y_2 = 5$ кН, $Y_3 = -10$ кН, $R_1 = 20$ кН, $R_2 = 11$ кН, $R_3 = -3$ кН, $R_4 = 15$ кН, $R_5 = 10$ кН.

3. В качестве проверки составим уравнение моментов для всей балки в целом (рис. 262) относительно левого ее конца. Имеем

$$\begin{aligned}\sum M_O &= 8R_1 + 19R_2 + 24R_3 + 39R_4 + 44R_5 - 32P_2 - \\ &- q_1 \cdot 11 \cdot (8 + 11/2) - q_2 \cdot 5 \cdot (5/2 + 39) - m = 0.\end{aligned}$$

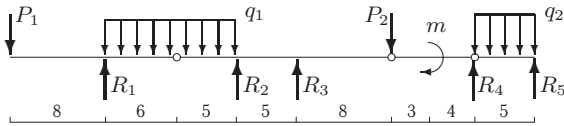


Рис. 262

В системе уравнений равновесия использованы уравнения проекций, поэтому для проверки было выбрано уравнение моментов.

4. Строим эпюры моментов и перерезывающих сил для каждой из отдельных частей балки. Эпюры моментов и перерезывающих сил, полученные объединением отдельных эпюр, изображаем на рис. 263

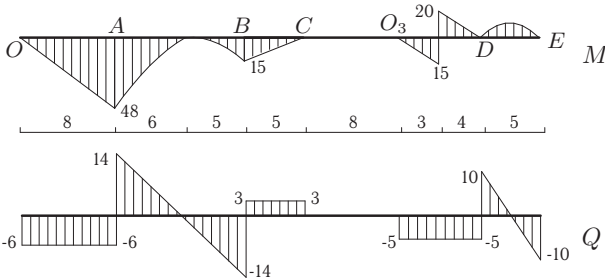


Рис. 263

Эпюры моментов и перерезывающих сил согласованы с дифференциальной зависимостью $Q = dM/dx$. Максимумы моментов соответствуют нулям функции Q , скачки на Q находятся в местах излома M . Там, где $Q = \text{const}$, момент линейный.

Maple-программа для решения этой задачи дана на с. 448.

Задача 63. Построить линию влияния реакции опоры V_B в многопролетной балке (рис. 264). Найти минимальное и максимальное значения V_B при совместном действии равномерной временной $q^{BP} = 5 \text{ кН/м}$ и постоянной нагрузки $q^n = 2 \text{ кН/м}$. Размеры даны в метрах.

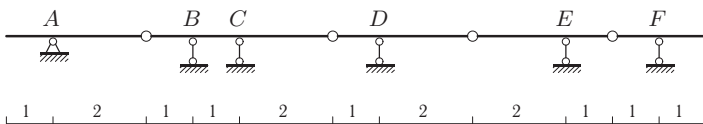


Рис. 264

Решение

Значения реакций опор конструкции или усилие к какому-либо ее элементу зависят от места приложения нагрузки и ее величины. Исследование этой зависимости необходимо для анализа работы конструкции при различных вариантах нагружений. Линия влияния — это зависимость некоторой исследуемой величины от места приложения единичной нагрузки.

Нагрузку делим на постоянную и временную. Постоянная нагрузка — равномерно распределенная по всей длине балки нагрузка, вызванная весом конструкции. Временная нагрузка — равномерно распределенная на определенных участках нагрузка от действия внешних факторов. Если рассматривается модель моста, то временной нагрузкой может быть вес автомобиля или поезда.

1. Освобождаем балку от связи в опоре B . Даем единичное вертикальное перемещение балке в точке B (рис. 265).

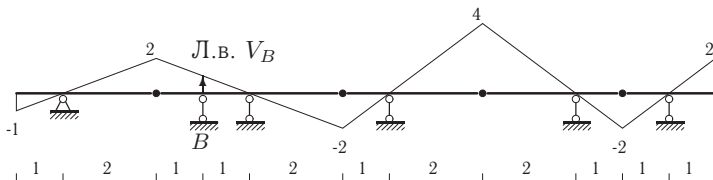


Рис. 265

2. Изображаем ломаную линию балки в отклоненном состоянии. Каждый отдельный элемент балки остается прямолинейным, все шарниры на опорных стержнях (кроме одного, в точке B) остаются неподвижными. Сочленяющие шарниры могут подниматься, опускаться или оставаться на месте. Балки, закрепленные на двух опорах, остаются горизонтальными. Построенная ломаная линия балки представляет собой линию влияния заданной реакции.

3. Вычисляем усилие от действия постоянной нагрузки q_n , равномерно распределенной по *всей* балке по формуле $V_B^n = \omega q_n$, где ω — площадь линии, ограниченной линией влияния $V_B(x)$:

$$\omega = (-1 \cdot 1 + 2 \cdot 4 - 2 \cdot 3 + 4 \cdot 4 - 2 \cdot 2 + 2 \cdot 1) / 2 = 7,5 \text{ м.}$$

Получаем $V_B^n = 7,5 \cdot 2 = 15 \text{ кН}$.

4. Вычисляем максимальное значение $V_B^{вр}$ усилия от действия временной нагрузки. Прикладываем равномерно распределенную нагрузку $q_{вр}$ к той части балки, где ординаты линии влияния $V_k(x)$ положительные (рис. 266). Суммируя площади треугольников, получаем $\omega^+ = (2 \cdot 4 + 4 \cdot 4 + 2 \cdot 1) / 2 = 13 \text{ м}$. Находим максимальное значение реакции: $V_B^{вр} = \omega^+ q_{вр} = 13 \cdot 5 = 65 \text{ кН}$.

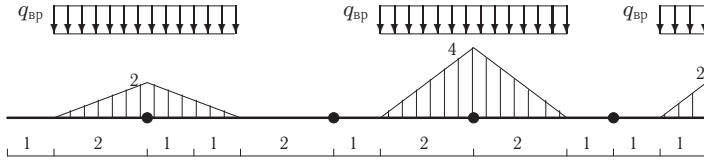


Рис. 266

5. Вычисляем минимальное значение $V_k^{\text{BP}} \min$ усилия от действия временной нагрузки. Прикладываем распределенную нагрузку $q_{\text{вр}}$ к той части балки, где ординаты линии влияния $V_k(x)$ отрицательные (рис. 267). Находим сумму отрицательных площадей $\omega^- = (-1 \cdot 1 - 2 \cdot 3 - 2 \times 2)/2 = -5,5$ м и минимальное значение реакции $V_B^{\text{BP}} \min = \omega^- \cdot q_{\text{вр}} = -5,5 \cdot 5 = -27,5$ кН.

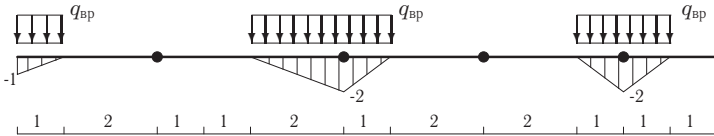


Рис. 267

6. Вычисляем экстремальные значения усилия от совместного действия временной и постоянной нагрузки: $V_B \max = V_B^{\text{п}} + V_B^{\text{BP}} \max = 15 + 65 = 80$ кН, $V_B \min = V_B^{\text{п}} + V_B^{\text{BP}} \min = 15 - 27,5 = -12,5$ кН.

Замечание. Линию влияния можно использовать не только для оценки максимального и минимального значения усилия при различных нагружениях, но и для расчета конструкции при заданной нагрузке. Загрузим, например, балку вертикальными силами, приложенными к концам балки и к сочленяющим шарнирам (рис. 268). Дано: $P_1 = 11$ кН, $P_2 = 12$ кН, $P_3 = 13$ кН, $P_4 = 14$ кН, $P_5 = 15$ кН, $P_6 = 16$ кН. Используя ординаты линии влияния (рис. 265), вычислим усилие в опоре B по формуле $V_B = -1 \cdot P_1 + 2 \cdot P_2 - 2 \cdot P_3 + 4 \cdot P_4 - 2 \cdot P_5 + 2 \cdot P_6 = -1 \cdot 11 + 2 \cdot 12 - 2 \cdot 13 + 4 \cdot 14 - 2 \cdot 15 + 2 \cdot 16 = 45$ кН.

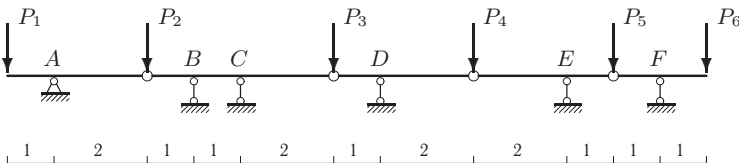


Рис. 268

Для того чтобы получить это же значение обычными методами статике с разбиением балки на отдельные тела и составлением уравнений равновесия, потребовалось бы решить систему с 10 неизвестными.

На рис. 269 изображены линии влияния других опор балки, а в таблице 1 — соответствующие значения минимальных и максимальных нагрузок.

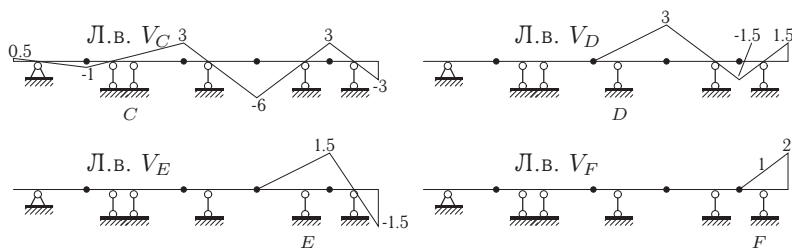


Рис. 269

Maple-программа для решения этой задачи дана на с. 453.

Таблица 1

Реакция	V_{\max}^{BP}	V_{\min}^{BP}	V^{II}	V_{\max}	V_{\min}
V_C	46,250	-75,000	-11,500	34,750	-86,500
V_D	41,250	-7,500	13,500	54,750	6,000
V_E	15,000	-3,750	4,500	19,500	0,750
V_F	10,000	0,000	4,000	14,000	4,000

4.9. Статически определяемая рама

Задача 64. На раму, состоящую из четырех шарнирно соединенных частей, действуют две одинаковые силы $P = 10$ кН и распределенные нагрузки $q_1 = 5$ кН/м, $q_2 = 4$ кН/м. Все стержни имеют одинаковую жесткость на изгиб EJ . Построить эпюру моментов в раме (рис. 270). Найти вертикальное смещение Δ_{yC} шарнира C .

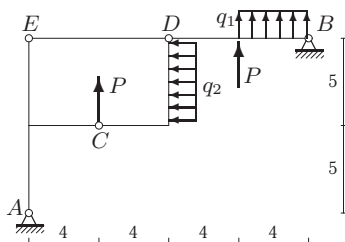


Рис. 270

Решение

1. Разбиваем систему на части по шарнирам C, D, E . Силу, приложенную к шарниру C , относим к части CD ¹. Действие отброшенных частей и опор заменяем их реакциями. Реакции частей, соединенных шарниром, взаимно противоположны по направлению и равны по модулю (рис. 271).

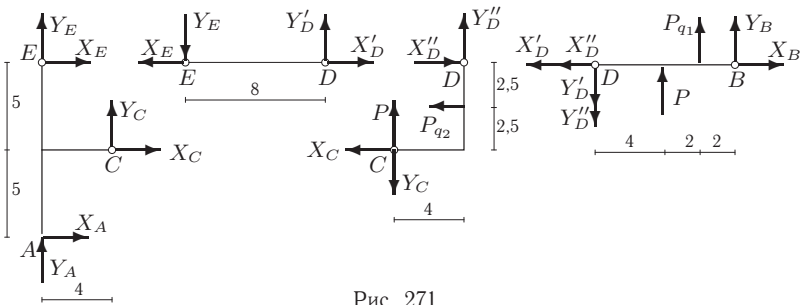


Рис. 271

2. Составляем три уравнения равновесия для каждой из частей. Равнодействующие распределенных нагрузок равны $P_{q1} = 4q_1 = 20$ кН, $P_{q2} = 5q_2 = 20$ кН. Точки приложения этих сил находятся в серединах соответствующих участков.

Равновесие части AE :

$$\begin{aligned}\sum X_i &= X_A + X_C + X_E = 0, \\ \sum Y_i &= Y_A + Y_C + Y_E = 0, \\ \sum M_A &= -10X_E - 5X_C + 4Y_C = 0.\end{aligned}\quad (4.60)$$

Равновесие части DE :

$$\begin{aligned}\sum X_i &= -X_E + X'_D = 0, \\ \sum Y_i &= -Y_E + Y'_D = 0, \\ \sum M_E &= 8Y'_D = 0.\end{aligned}\quad (4.61)$$

Равновесие части CD :

$$\begin{aligned}\sum X_i &= -X_C + X''_D - P_{q2} = 0, \\ \sum Y_i &= -Y_C + Y''_D + P = 0, \\ \sum M_C &= -5X''_D + 4Y''_D + 2,5P_{q2} = 0.\end{aligned}\quad (4.62)$$

¹ Можно показать, что решение не изменится, если силу приложить к другой части или даже поделить ее между частями.

Равновесие части DB :

$$\begin{aligned}\sum X_i &= -X'_D - X''_D + X_B = 0, \\ \sum Y_i &= -Y'_D - Y''_D + Y_B + P + P_{q1} = 0, \\ \sum M_D &= 8Y_B + 6P_{q1} + 4P = 0.\end{aligned}\quad (4.63)$$

3. Решаем систему 12 уравнений (4.60)–(4.63)¹. Находим все 12 неизвестных: реакции опор и внутренние реакции. Имеем следующие значения: $Y'_D = 0$, $Y_E = 0$, $Y_B = -20$ кН, $Y''_D = 10$ кН, $Y_C = 20$ кН, $X_C = -2$ кН, $X_A = -7$ кН, $X_B = 27$ кН, $X'_D = 9$ кН, $X''_D = 18$ кН, $X_E = 9$ кН, $Y_A = -20$ кН.

4. Строим эпюру моментов в раме от действия нагрузки.

Используем правило, согласно которому положительные значения ординат эпюры откладываются на *сжатом* волокне². Используем метод сечений. Из условия равновесия части рамы определяем моменты в сечении, аналогично тому, как это было показано в решении задачи 61, с. 182. Построенная эпюра M_P изображена на рис. 273.

5. Прикладываем к точке рамы, перемещение которой разыскивается, единичную силу по направлению искомого перемещения. В данном случае вертикальная сила 1 прикладывается к шарниру C (рис. 272).

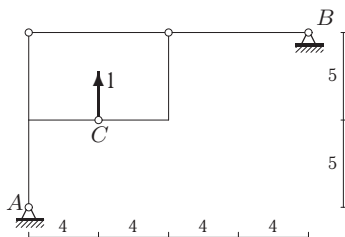


Рис. 272

6. Определяем реакции опор рамы от действия единичной силы. Разбиваем по шарнирам раму на части, прикладываем вместо отброшенных

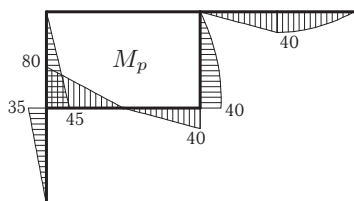


Рис. 273

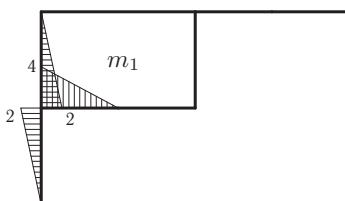


Рис. 274

¹Решение в системе **Maple** см. на с. 445.

²В архитектурно-строительных вузах принято обратное правило — эпюры моментов строятся на *растянутом* волокне.

тел их реакции. Реакции ненагруженных стержней направляем вдоль их осей.

6. Составляем уравнения равновесия для каждой из частей. В данном случае получаем систему шести уравнений¹.

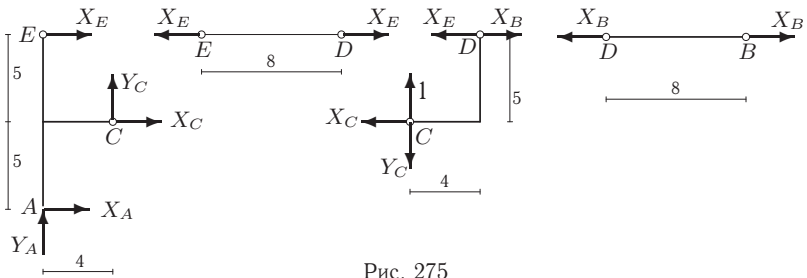


Рис. 275

Равновесие части AE описывается системой уравнений, совпадающей с (4.64) при $Y_E = 0$:

$$\begin{aligned}\sum X_i &= X_A + X_C + X_E = 0, \\ \sum Y_i &= Y_A + Y_C = 0, \\ \sum M_A &= -10X_E - 5X_C + 4Y_C = 0.\end{aligned}\quad (4.64)$$

Так как уравнения равновесия частей ED и DB удовлетворяются тождественно, запишем уравнения для CD :

$$\begin{aligned}\sum X_i &= -X_C - X_E + X_B = 0, \\ \sum Y_i &= -Y_C + 1 = 0, \\ \sum M_C &= 5X_E - 5X_B = 0.\end{aligned}\quad (4.65)$$

7. Решаем систему. Находим реакции опор и внутренние реакции от действия единичной силы: $X_C = 0$, $Y_C = 1$, $Y_A = -1$, $X_B = 0,4$, $X_A = -0,4$, $X_E = 0,4$. Так как к раме была приложена условная безразмерная сила, то и реакции также безразмерные.

8. Строим эпюру моментов от действия единичной силы (рис. 274).

¹Если уравнения равновесия рамы и ее частей от действия внешней нагрузки записывать в матричной форме для последующего решения на компьютере, то составление аналогичной системы для единичной нагрузки сводится к замене столбца правой части, куда входит только одна величина — единичная сила.

9. Перемножаем эпюры M_P и m_1 . Пользуясь правилом Верещагина, вычисляем интеграл Максвелла–Мора:

$$\delta_y = \frac{1}{EJ} \int_L M_P m_1 ds = \frac{1}{EJ} \left(\frac{1}{2} \cdot 5 \cdot 35 \cdot \frac{2}{3} \cdot 2 + \frac{1}{2} \cdot 5 \cdot 45 \cdot \frac{2}{3} \cdot 2 + \frac{1}{2} \cdot 4 \cdot 80 \cdot \frac{2}{3} \cdot 4 \right) = \frac{693,333}{EJ}. \quad (4.66)$$

Так как M_P измеряется в кНм, момент m_1 от безразмерной нагрузки в метрах, то интеграл получается в кНм³. Если модуль упругости, заданный обычно в МПа, перевести в кН/м², а момент инерции J задать в м⁴, то получим, что знаменатель измеряется в кНм². Следовательно, перемещение получается в метрах. Например, для двутавра № 30 ГОСТ 8239-89 с моментом инерции $7080 \text{ см}^4 = 7080 \cdot 10^{-8} \text{ м}^4$ из стали с модулем упругости $E = 2 \cdot 10^5 \text{ МПа} = 2 \cdot 10^8 \text{ кН/м}^2$ получим перемещение $0,04896 \text{ м} = 4,896 \text{ см}$.

Знак интеграла указывает направление перемещения относительно заданной единичной силы. Если интеграл меньше нуля, то перемещение точки направлено вниз.

Замечание 1. Система уравнений (4.60)–(4.63) записана в расчете на произвольную нагрузку, но в данном случае расчет можно сократить, воспользовавшись тем, что прямолинейный стержень ED не нагружен, и его реакции по концам направлены вдоль стержня и равны по модулю, $X_E = X'_D$. Так как для этого стержня, фактически выполняющего роль связи, не надо писать уравнения равновесия, то система становится на три уравнения меньше. Исчезают и неизвестные $Y_E = 0$, $Y'_D = 0$, $X_E = X'_D$.

Замечание 2. Система 12 уравнений (4.60)–(4.63) для определения всех реакций связей несколько сложна. Однако для определения (или проверки) четырех реакций внешних связей A и B достаточно составить только четыре уравнения. Записываем уравнение моментов относительно точки D части DB : $8Y_B + 4P + 4q_1 \cdot 6 = 0$ (последнее уравнение в (4.63)), находим $Y_B = -20 \text{ кН}$. Затем составляем три уравнения равновесия всей конструкции в целом (рис. 276):

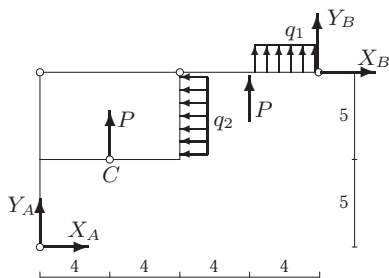


Рис. 276

$$\begin{aligned}\sum X_i &= X_A + X_B - 5q_2 = 0, \\ \sum Y_i &= Y_A + Y_B + 2P + 4q_1 = 0, \\ \sum M_A &= -10X_B + 16Y_B + 14 \cdot 4q_1 + 7,5 \cdot 5q_2 + 12P + 4P = 0.\end{aligned}\quad (4.67)$$

Решение этой системы дает те же значения, что и на с. 193.

Расчет рамы в системе **Maple** с построением эпюр моментов и перерезывающих сил дан в программе 41, с. 433.

4.10. Статически неопределимая рама

Задача 65. Построить эпюры моментов, нормальных и перерезывающих сил в раме (рис. 277). В точке *A* рама имеет жесткую заделку, в точке *B* — неподвижный шарнир. Даны нагрузки: $P = 6$ кН, $q = 8$ кН/м.

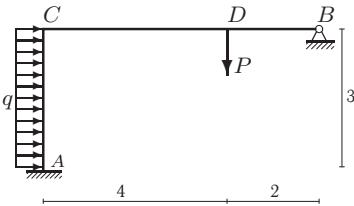


Рис. 277

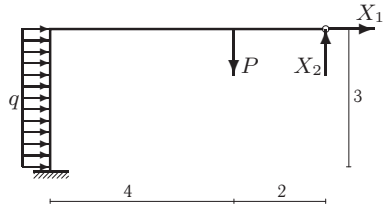


Рис. 278

Решение

1. Определяем число n лишних связей (число статической неопределимости) по формуле $N_F = 3K - \text{Ш}$, где K — число контуров в раме, Ш — число шарниров с учетом их кратности. Кратность шарнира равна числу тел, соединенных шарниром минус единица.



Рис. 279



Рис. 280



Рис. 281



Рис. 282

На рис. 279, 280 шарниры однократные, на рис. 281 изображен двухкратный шарнир, на рис. 282 — трехкратный (земля также считается телом). В данной задаче рама имеет один контур и один однократный шарнир. Имеем в данном случае $N_F = 3K - \text{Ш} = 3 -$

– 1 = 2. Рама дважды статически неопределима¹. Отбрасывая лишние связи, например в шарнире B , и заменяя их действие неизвестными реакциями X_1, X_2 , получаем основную систему, рис. 278. Заметим, что основная система не единственная. На рис. 283, 284, 285 приведены еще три варианта. В последнем случае в угол C рамы врезан шарнир, и, одновременно, к образовавшимся частям рамы приложены равные по модулю противоположно направленные пары X_2 .

Неудачный выбор лишних связей может привести к недопустимой кинематической изменяемости основной системы.

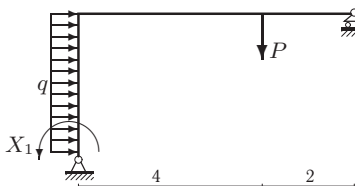


Рис. 283

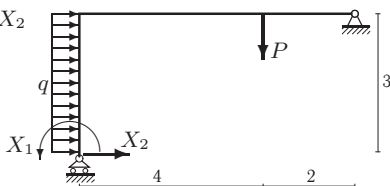


Рис. 284

На рис. 286 — пример неправильной основной системы. Заделка в точке A превращена в горизонтально скользящую и отброшена горизонтальная связь в шарнире B . Рама имеет горизонтальную подвижность.

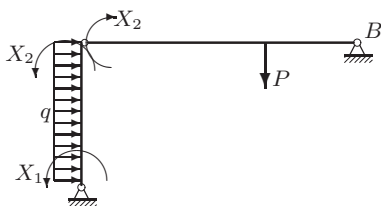


Рис. 285

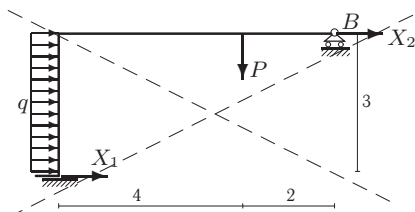


Рис. 286

2. Строим эпюры моментов M_P , перерезывающих Q_P и нормальных сил N_P в основной системе от действия внешней нагрузки (рис. 287). В данной задаче основная система выбрана консольного вида, поэтому при построении эпюр нет необходимости определять реакции опор X_A, Y_A, M_A . Выполняя сечения рамы, рассматриваем всякий раз часть

¹Еще один пример. Если в статически определимой раме на рисунке 270, с. 191, заварить шарнир C и лишить раму, таким образом, одной степени свободы, то рама будет один раз статически неопределимой. Действительно, число контуров в ней равно двум, три однократных шарнира (включая два опорных) и один двухкратный в точке C . Вычисляем $N_F = 3 \cdot 2 - 5 = 1$.

рамы, не содержащую заделку. Полученные эпюры изображены на рис. 288, 289, 290.

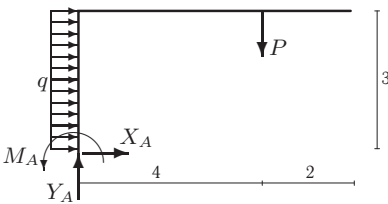


Рис. 287

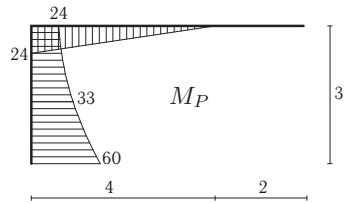


Рис. 288

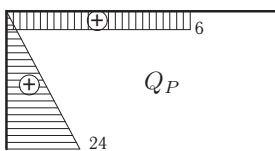


Рис. 289

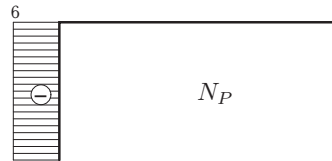


Рис. 290

3'. Строим эпюры моментов m_1 , перерезывающих q_1 и нормальных сил n_1 в основной системе от действия единичной силы, направленной по неизвестной реакции X_1 . Внешние силы и неизвестные реакции к раме не прикладываем (рис. 291).

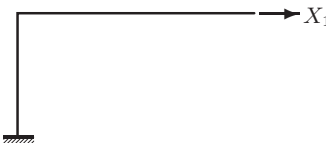


Рис. 291

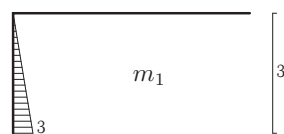


Рис. 292

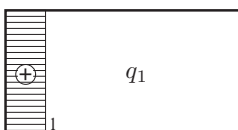


Рис. 293

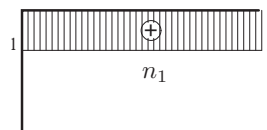


Рис. 294

3''. Строим эпюры моментов m_2 , перерезывающих q_2 и нормальных сил n_2 в основной системе от действия единичной силы, направленной по неизвестной реакции X_2 (рис. 295).



Рис. 295

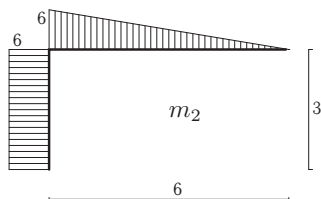


Рис. 296

4. По формуле Максвелла–Мора определяем коэффициенты канонической системы метода сил. Пользуемся правилом Верещагина для

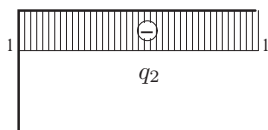


Рис. 297

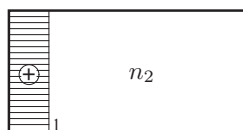


Рис. 298

перемножения эпюр. Площадь криволинейной эпюры умножаем на ординату прямолинейной под центром тяжести криволинейной. Ординату выделим полужирным шрифтом. Площадь параболической эпюры на участках с распределенной нагрузкой q вычисляем по формуле $(2/3)fl$, где l — длина участка, $f = ql^2/8$ — высота параболической эпюры. На участке AC имеем $l = 3$ м, $q = 8$ кН/м, $f = 8 \cdot 3^2/8 = 9$ кНм.

$$\delta_{11} = \int_L \frac{m_1^2 ds}{EJ} = \frac{1}{EJ} \left(\frac{1}{2} \cdot 3 \cdot 3 \cdot \frac{2}{3} \cdot 3 \right) = \frac{9}{EJ},$$

$$\delta_{22} = \int_L \frac{m_2^2 ds}{EJ} = \frac{1}{EJ} \left(6 \cdot 3 \cdot 6 + \frac{1}{2} \cdot 6 \cdot 6 \cdot \frac{2}{3} \cdot 6 \right) = \frac{180}{EJ},$$

$$\delta_{12} = \delta_{21} = \int_L \frac{m_1 m_2 ds}{EJ} = \frac{1}{EJ} \left(-\frac{1}{2} \cdot 3 \cdot 3 \cdot 6 \right) = \frac{-27}{EJ},$$

$$\Delta_{1P} = \int_L \frac{m_1 M_P ds}{EJ} = \frac{1}{EJ} \left(\frac{1}{2} \cdot 60 \cdot 3 \cdot \frac{2}{3} \cdot 3 + \frac{1}{2} \cdot 24 \cdot 3 \cdot \frac{1}{3} \cdot 3 - \frac{2}{3} \cdot 9 \cdot 3 \cdot \frac{1}{2} \cdot 3 \right) = \frac{189}{EJ},$$

$$\Delta_{2P} = \int_L \frac{m_2 M_P ds}{EJ} = \frac{1}{EJ} \left(-\frac{60+24}{2} \cdot 3 \cdot 6 + \frac{2}{3} \cdot 9 \cdot 3 \cdot 6 - \frac{1}{2} \cdot 24 \cdot 4 \cdot \left(\frac{2}{3} \cdot 6 + \frac{1}{3} \cdot 2 \right) \right) = -\frac{872}{EJ}.$$

Для проверки умножения можно воспользоваться следующими очевидными тождествами:

$$\int_L \frac{(m_1 + m_2)^2 ds}{EJ} = \delta_{11} + 2\delta_{12} + \delta_{22},$$

$$\int_L \frac{(m_1 + m_2) M_P ds}{EJ} = \Delta_{1P} + \Delta_{2P}.$$
(4.68)

5. Записываем и решаем каноническое уравнение метода сил

$$\begin{aligned} 9X_1 - 27X_2 + 189 &= 0, \\ -27X_1 + 180X_2 - 872 &= 0. \end{aligned}$$
(4.69)

Находим неизвестные метода сил $X_1 = -11,76$ кН, $X_2 = 3,08$ кН.

6. Строим эпюры моментов M , перерезывающих Q и нормальных сил N в раме по формулам

$$\begin{aligned} M &= M_P - 11,76m_1 + 3,08m_2, \\ Q &= Q_P - 11,76q_1 + 3,08q_2, \\ N &= N_P - 11,76n_1 + 3,08n_2. \end{aligned}$$
(4.70)

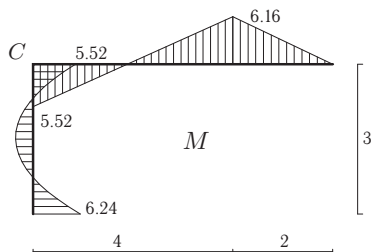


Рис. 299

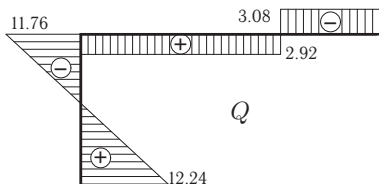


Рис. 300

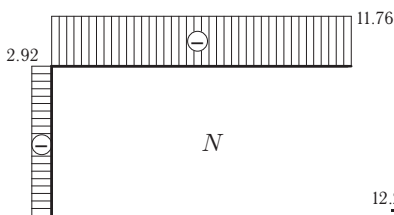


Рис. 301

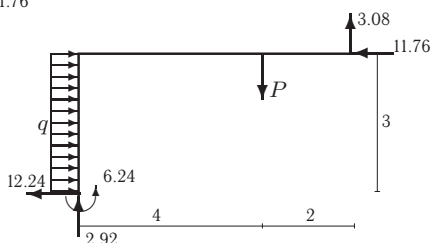


Рис. 302

7. Выполняем статическую проверку. Прикладываем к раме внешнюю нагрузку, действие опор заменяем реакциями, значение которых снимем с эпюр M , Q и N (рис. 302). Соблюдаем правило знаков для перерезывающих сил (положительные силы вращают участок по часовой стрелке), нормальных сил (положительные силы растягивают участок) и моментов (эпюра моментов построена на сжатом волокне). Проверяем выполнение уравнений равновесия:

$$\begin{aligned}\sum X_i &= -12,24 + q \cdot 3 - 11,76 = -12,24 + 24 - 11,76 = 0, \\ \sum Y_i &= 2,92 - P + 3,08 = 2,92 - 6 + 3,08 = 0, \\ \sum M_A &= 3,08 \cdot 6 + 11,76 \cdot 3 - 24 \cdot 1,5 - 6 \cdot 4 + 6,24 = 0.\end{aligned}\quad (4.71)$$

Статическая проверка выполняется. Рама находится в равновесии.

8. Выполняем деформационную проверку¹. Для этого проверяем равенство нулю произведения построенной эпюры моментов M (рис. 299) в раме на эпюру моментов от единичного состояния m_2 (рис. 296):

$$\begin{aligned}\int_L \frac{m_2 M ds}{EJ} &= \frac{1}{EJ} \left(-\frac{6,24 + 5,52}{2} \cdot 3 \cdot 6 + \frac{2}{3} \cdot 9 \cdot 3 \cdot 6 - \right. \\ &- \frac{5,52}{2} \cdot 4 \cdot \left(\frac{2}{3} \cdot 6 + \frac{1}{3} \cdot 2 \right) + \frac{6,16}{2} \cdot 4 \cdot \left(\frac{2}{3} \cdot 2 + \frac{1}{3} \cdot 6 \right) + \\ &\left. + \frac{1}{2} \cdot 6,16 \cdot 2 \cdot \frac{2}{3} \cdot 2 \right) = 0.\end{aligned}\quad (4.72)$$

Выполнение деформационной проверки означает, что рама в шарнире B не имеет вертикального смещения. Для проверки горизонтального смещения в этой же точке необходимо перемножить m_1 и M . Для проверки угла поворота оси рамы в заделке (он должен быть равен

¹Иногда эту проверку называют кинематической, см. с. 465.

нулю) умножаем M на эпюру в основном состоянии рис. 283 или рис. 284 от действия единичного момента по направлению X_1 .

Maple–программа расчета статически неопределимой рамы методом сил приведена на с. 460.

4.11. Трехшарнирная арка

Арка широко используется в строительстве и машиностроении. Особую роль арка играет в архитектуре. Расчет усилий в арке представляет собой интересную и важную задачу.

Арки различаются формой и видом наложенных связей. Трехшарнирная статически определимая арка состоит из двух шарнирно опертых частей, соединенных шарниром¹. Очертание арки, как правило, бывает круговым, эллиптическим, параболическим или по форме цепной линии (рис. 303). Арки относятся к распорным конструкциям — действие вертикальной нагрузки вызывает в них горизонтальные реакции в опорах. В простых балках и балочных фермах этого эффекта нет. Распор для пологих арок весьма значителен. В тех случаях, когда нельзя обеспечить связи, выдерживающие большие горизонтальные нагрузки, применяют арки со стяжкой (рис. 304).

Будем рассматривать симметричную арку пролетом $2l$ с высотой (стрелой) f .

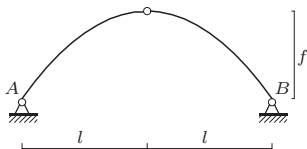


Рис. 303

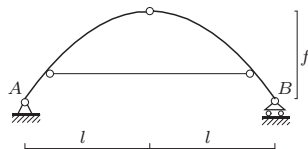


Рис. 304

Задача 66. Построить эпюры моментов, перерезывающих и нормальных сил в трехшарнирной арке параболической формы (рис. 305). Даны нагрузки $q = 2$ кН/м, $P_1 = 9$ кН, $P_2 = 6$ кН. Размеры указаны в метрах. Для $x_1 = 10$ м вычислить значения $M(x_1)$, $Q(x_1)$, $N(x_1)$. Начало координат находится на левой опоре арки.

Решение

1. Определяем вертикальные реакции V_A , V_B опор арки. Освобождаем арку от внешних связей (пятовых опор), заменяя их действие реакциями V_A , V_B и распором H . В отсутствии внешних горизонтальных

¹Опорные шарниры называются пятами арки, сочленяющий шарнир — ключом или замком арки.

нагрузок горизонтальные реакции равны и противоположно направлены (рис. 306). В тех случаях, когда опоры арки находятся на одном уровне,

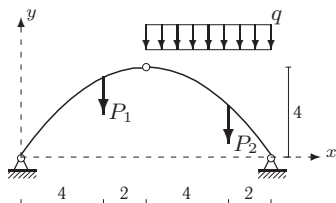


Рис. 305

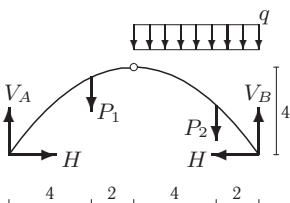


Рис. 306

для определения вертикальных составляющих реакций не требуется разбивать арку на части. Составляем уравнения моментов относительно опор:

$$\sum M_A = V_B \cdot 12 - q \cdot 6 \cdot 9 - P_1 \cdot 4 - P_2 \cdot 10 = 0,$$

$$\sum M_B = -V_A \cdot 12 + q \cdot 6 \cdot 3 + P_1 \cdot 8 + P_2 \cdot 2 = 0.$$

Решаем уравнения: $V_A = 10$ кН, $V_B = 17$ кН. Выполняем проверку решения. Составляем уравнение проекций на ось y :

$$\sum Y_i = V_A + V_B - P_1 - P_2 - q \cdot 6 = 10 + 17 - 9 - 6 - 12 = 0.$$

2. Определяем распор H в арке. Разделяем арку по внутреннему шарниру на две части. Рассматриваем равновесие одной из частей, например левой (рис. 307).

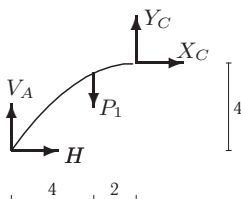


Рис. 307

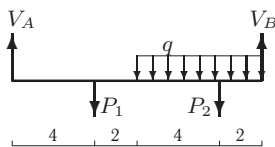


Рис. 308

Действие отброшенной части заменяем неизвестными реакциями связи Y_C , X_C . Составляем сумму моментов сил, действующих на левую часть, относительно шарнира C : $\sum M_B = -V_A \cdot 6 + P_1 \cdot 2 + H \cdot 4 = 0$. Отсюда определяем распор: $H = 10,5$ кН.

3. Строим балочные эпюры моментов M_0 и перерезывающих сил Q_0 . Начало координат поместим в левую опору. Нагрузка на эквивалентную

балку, опертую по концам (шарнира в середине нет), изображена на рисунке 308. Построение эпюр в балке подробно описано на с. 182. На первых трех участках $0 \leq x \leq 4$, $4 \leq x \leq 6$, $6 \leq x \leq 10$ рассматриваем равновесие левой части балки, на последнем при $10 \leq x \leq 12$ — правой части.

На участке 1 (рис. 309) действует одна единственная сила — реакция опоры V_A . Она направлена вверх и, следовательно, создает на этом участке положительную перерезывающую силу, а относительно сечения она же дает положительный момент. Напомним, что в сечении значками M^+ и Q^+ указаны не реакции, а положительные направления для моментов и сил в выражениях для эпюр. Момент и перерезывающая сила на участке 1 балки, где $0 \leq x \leq 4$, имеют вид $M_0(x) = V_A x$, $Q_0(x) = V_A$

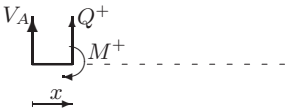


Рис. 309

На участке 2 (рис. 310) добавляется внешняя сосредоточенная сила P_1 , направленная по отношению к V_A в противоположную сторону, поэтому в выражения для Q и M она входит с обратными знаками. Координата сечения x отсчитывается от той же точки, плечо силы P относительно сечения равно $x - 4$. Момент и перерезывающая сила на участке 2 балки при $4 \leq x \leq 6$ имеют вид $M_0(x) = V_A x - P_1(x - 4)$, $Q_0(x) = V_A - P_1$.

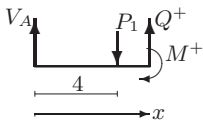


Рис. 310

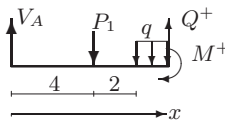


Рис. 311

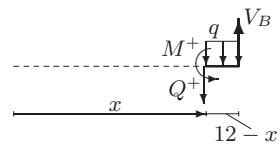


Рис. 312

Выполняя сечение в произвольном месте третьего участка балки (рис. 311), отсекаем часть распределенной нагрузки длиной $x - 6$. При интенсивности q равнодействующая этой части равна $q(x - 6)$ и приложена в середине участка длиной $x - 6$. Плечо равнодействующей распределенной нагрузки относительно сечения $(x - 6)/2$. Момент и перерезывающая сила на участке 3 балки имеют вид

$$M_0(x) = V_A x - P_1(x - 4) - q(x - 6)^2/2,$$

$$Q_0(x) = V_A - P_1 - q(x - 6), \quad 6 \leq x \leq 10.$$

На последнем участке (рис. 312) рассматриваем равновесие правой отсеченной части. Правило знаков меняется на противоположное.

$$M_0(x) = V_B(12 - x) - q(12 - x)^2/2,$$

$$Q_0(x) = -V_B + q(12 - x), \quad 10 \leq x \leq 12.$$

По характерным точкам на оси арки (концы участков) строим балочные эпюры. В тех участках, где действует распределенная нагрузка, эпюра моментов будет параболической, а эпюра перерезывающих сил — линейной с некоторым наклоном. На других участках обе эпюры линейные. Получаем эпюру моментов (рис. 313, в кНм) и эпюру перерезывающих сил (рис. 314, в кН), табл. 2.

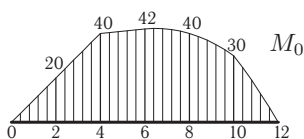


Рис. 313

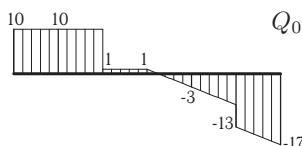


Рис. 314

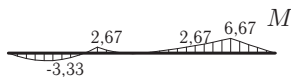


Рис. 315

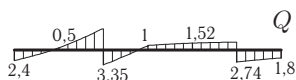


Рис. 316

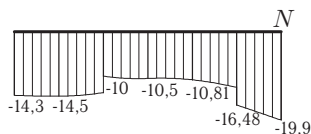


Рис. 317

7. По формуле $N = -Q_0 \sin \varphi - H \cos \varphi$ строим эпюру нормальных сил N в арке (рис. 317).

На третьем участке $6 \leq x \leq 10$ м, перерезывающая сила обращается в ноль при $x = 6 + (V_A - P_1)/q = 6,5$ м. Согласно дифференциальной зависимости $Q_0 = dM_0/dx$, момент в этой точке экстремальный, $M_0(6,5) = 42,25$ кНм.

4. Эпюры строим по отдельным точкам. Разбиваем пролет на $n = 6$ частей. Для точек $x_i = (2l/n)i$, $i = 0, \dots, n$ по формулам

$$\operatorname{tg} \varphi_i = y'_i,$$

$$\sin \varphi_i = \operatorname{tg} \varphi_i / \sqrt{1 + \operatorname{tg}^2 \varphi_i}, \quad (4.73)$$

$$\cos \varphi_i = \sqrt{1 - \sin^2 \varphi_i},$$

где для параболической арки с началом координат в левой опоре

$$y_i = f x_i(2l - x_i)/l^2, \quad (4.74)$$

$$y'_i = 2f(l - x_i)/l^2,$$

вычисляем вспомогательные величины y_i , $\operatorname{tg} \varphi_i$, $\sin \varphi_i$, $\cos \varphi_i$, $i = 0, \dots, 6$. Результаты заносим в таблицу 2.

5. По формуле $M(x_i) = M_0(x_i) - y(x_i)H$, $i = 1, \dots, 7$ строим эпюру моментов M в арке (рис. 315).

6. По формуле $Q = Q_0 \cos \varphi - H \sin \varphi$ строим эпюру перерезывающих сил Q в арке (рис. 316).

Таблица 2

Балочные эпюры, координаты и тригонометрические функции

x	y	M_0	Q_0	y'	$\sin(\varphi)$	$\cos(\varphi)$
0	0,000	0,000	10,000	1,333	0,800	0,600
2	2,222	20,000	10,000	0,889	0,664	0,747
4	3,556	40,000	1,000	0,444	0,406	0,914
6	4,000	42,000	1,000	0,000	0,000	1,000
8	3,556	40,000	-3,000	-0,444	-0,406	0,914
10	2,222	30,000	-13,000	-0,889	-0,664	0,747
12	0,000	0,000	-17,000	-1,333	-0,800	0,600

Выписываем отдельно значения M , Q и N в заданном сечении¹:

$$M = M_0 - yH = 30 - 2,22 \cdot 10,5 = 6,667 \text{ кНм},$$

$$Q = Q_0 \cos \varphi - H \sin \varphi = -13 \cdot 0,747 - 10,5 \cdot 0,664 = -2,741 \text{ кН},$$

$$N = -Q_0 \sin \varphi - H \cos \varphi = -13 \cdot 0,664 - 10,5 \cdot 0,747 = -16,485 \text{ кН}.$$

Замечание. Аналогично рассчитываются арки другой формы. Для круговой арки радиуса R высотой f и пролетом $2l$ с началом координат в левой пятке формулы (4.74) имеют вид

$$y = \sqrt{R^2 - (l - x)^2} + f - R, \quad y'_i = (l - x) / \sqrt{R^2 - (l - x)^2},$$

где $R = (l^2 + f^2) / (2f)$. Если арка тех же размеров имеет форму цепной линии, то

$$y = (2 - \operatorname{ch}(a(1 - x/l)))f, \quad y'_i = (fa/l) \operatorname{sh}(a(1 - x/l)),$$

где $a = \operatorname{arccosh}(2) \approx 1,317$. Такую форму, но с обратным знаком, имеет тяжелая гибкая нить, подвешенная на двух опорах.

4.12. Двухшарнирная арка

Рассмотрим параболическую арку высотой f с пролетом $2l$ на двух неподвижных опорах без шарнира в вершине. Такая арка представляет собой статически неопределимую систему. Используем для расчета метод сил. Если бы одна из опор была бы подвижной, то система была бы статически определимой, однако выбор такой основной системы не самый лучший. В качестве основной системы примем трехшарнирную арку. Алгоритм расчета такой арки приведен в предыдущем параграфе, а формулы для коэффициентов канонического уравнения, как будет показано ниже, получаются простыми. Шарнир, "врезанный" в вершину

¹При $x = 10$ эпюра перерезывающих сил балки имеет разрыв. Принимаем значение $Q_0 = -13$ кН на четвертом участке, т. е. справа от разрыва.



арки, превращает ее в статически определимую. Неизвестной величиной метода сил при таком способе расчета будет момент в замке.

План решения

1. Выбираем основную систему — трехшарнирную арку. Неизвестной величиной X_1 метода сил является момент в замке (рис. 318). К одной части арки момент приложен по часовой стрелке, к другой — против. Определение X_1 означает раскрытие статической неопределенности.

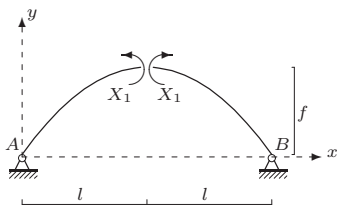


Рис. 318

2. Строим эпюру моментов в основной системе. Она совпадает с эпюрой моментов M для трехшарнирной арки.

3. Записываем каноническое уравнение метода сил

$$\delta_{11}X_1 + \Delta_{1p} = 0, \quad (4.75)$$

где коэффициенты уравнения имеют вид

$$\delta_{11} = \int_L \frac{m_1^2 ds}{EJ} + \int_L \frac{n_1^2 ds}{EF}, \quad \Delta_{1p} = \int_L \frac{m_1 M_P ds}{EJ} + \int_L \frac{n_1 N_P ds}{EF}. \quad (4.76)$$

Эпюру моментов m_1 от единичного момента в замке и единичную эпюру нормальных сил n_1 легко получить в аналитическом виде. Прикладываем к частям арки единичные моменты по направлению действия неизвестной X_1 метода сил. Из условия равновесия всей системы в целом (рис. 319) получаем, что $Y_A = Y_B = 0$.

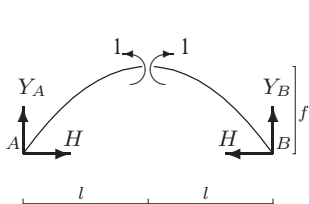


Рис. 319

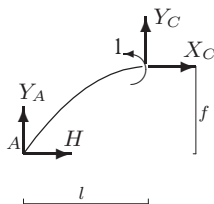


Рис. 320

Распор H_1 от действия единичного момента находим из рассмотрения равновесия левой части арки (рис. 320). Действие отрезанной правой части арки заменяем силами X_C, Y_C . Составляем уравнение моментов относительно шарнира C . Получаем $H_1 = -1/f$.

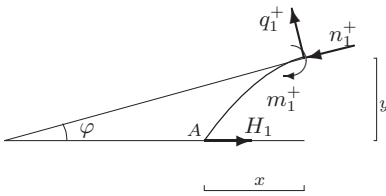


Рис. 321

Для построения эпюр от единичного момента выполняем в произвольном месте арки сечение (рис. 321). Указываем положительные направления для момента, нормальной силы и перерезывающей. Сжимающая нормальная сила принята отрицательной. Получаем простые выражения:

$$\begin{aligned} n_1 &= -H_1 \cos \varphi = (1/f) \cos \varphi, \\ q_1 &= -H_1 \sin \varphi = (1/f) \sin \varphi, \\ m_1 &= -yH_1 = y/f. \end{aligned}$$

Модуль упругости E и момент инерции J сечения относительно оси его центральной оси z приняты постоянными по длине арки. С учетом того, что площадь прямоугольного сечения $F = bh$, $J = bh^3/12$ и, таким образом, $J/F = h^2/12$, определяем коэффициенты

$$\begin{aligned} EJ\delta_{11} &= \int_L \frac{(y/f)^2 dx}{\cos \varphi} + \frac{h^2}{12f^2} \int_L \cos \varphi dx, \\ EJ\Delta_{1p} &= \int_L \frac{(y/f)M_p dx}{\cos \varphi} + \frac{h^2}{12f} \int_L N_p dx. \end{aligned} \quad (4.77)$$

4. Интегралы, входящие в коэффициенты канонического уравнения, вычисляем по формуле трапеций. Пролет арки разбиваем на n одинаковых участков $\Delta x = 2l/n$. Вычисляем подынтегральное выражение в точках разбиения $x_k = k\Delta x$, $k = 0, \dots, n$ и на концах отрезка $[0, 2l]$. Заменяем интеграл суммой:

$$\int_0^{2l} g(x) dx = \Delta x \left(\sum_{k=1}^{n-1} g(x_k) + \frac{g(0) + g(2l)}{2} \right). \quad (4.78)$$

5. Решаем уравнение (4.75) и находим X_1 . Статическая неопределенность раскрыта.

6. Получаем эпюры в двухшарнирной арке:

$$\begin{aligned} M &= M_p + m_1 X_1, \\ Q &= Q_p + q_1 X_1, \\ N &= N_p + n_1 X_1. \end{aligned} \quad (4.79)$$

Задача 67. Построить эпюры M , Q , N в двухшарнирной арке параболической формы (рис. 322) с прямоугольным сечением постоянной жесткости.

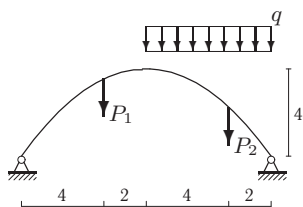


Рис. 322

Дано: равномерно распределенная нагрузка $q = 2$ кН/м и две сосредоточенных: $P_1 = 9$ кН, $P_2 = 6$ кН. Высота сечения арки $h = 0,6$ м. Размеры указаны в метрах. Для $x_1 = 4$ м вычислить значения $M(x_1)$, $Q(x_1)$, $N(x_1)$. Начало координат находится на левой опоре арки.

Решение

Рассчитываемая арка отличается от трехшарнирной статически определимой арки на рис. 315 отсутствием шарнира в средней точке пролета (замковый камень).

1. Система один раз статически неопределима. В качестве основной системы выбираем трехшарнирную арку. Незвестной величиной X_1 метода сил является момент в замке.

2. Строим эпюры моментов и перерезывающих сил в основной системе. Она совпадает с эпюрой M_P моментов для трехшарнирной арки (рис. 315).

3. Записываем каноническое уравнение метода сил (4.75).

4. Находим коэффициенты. Для вычисления интегралов (4.77) по формуле трапеций (4.78) составляем таблицу

Таблица 3

Вычисление интегралов

x	y/f	$(y/f)^2$	$\cos(\varphi)$	M_P	N_P
0	0,000	0,000	0,600	0,000	-14,300
2	0,556	0,309	0,747	-3,333	-14,491
4	0,889	0,790	0,914	2,667	-10,001
6	1,000	1,000	1,000	0,000	-10,500
8	0,889	0,790	0,914	2,667	-10,813
10	0,556	0,309	0,747	6,667	-16,485
12	0,000	0,000	0,600	0,000	-19,900

В последних двух столбцах — ординаты эпюр трехшарнирной арки (рис. 317). С учетом значений коэффициентов $(h/f)^2/12 = 0,186 \cdot 10^{-2}$, $h^2/(12f) = 0,75 \cdot 10^{-2}$ получаем следующие значения: $\delta_{11} = 7,142$, $\Delta_{1P} = 9,547$.

5. Решаем уравнение (4.75) и находим $X_1 = -1,337$. Статическая неопределенность раскрыта.

6. Получаем эпюры в двухшарнирной арке по формулам (4.79). Для суммирования эпюр удобно воспользоваться таблицей:

Таблица 4

Суммирование эпюр

x	$m_1 X_1$	M_p	M	$q_1 X_1$	Q_p	Q	$n_1 X_1$	N_p	N
0	0,000	0,000	0,000	-0,267	-2,400	-2,667	-0,201	-14,300	-14,501
2	-0,743	-3,333	-4,076	-0,222	0,498	0,276	-0,250	-14,491	-14,741
4	-1,188	2,667	1,479	-0,136	-3,351	-3,486	-0,305	-10,001	-10,307
6	-1,337	0,000	-1,337	0,000	1,000	1,000	-0,334	-10,500	-10,834
8	-1,188	2,667	1,479	0,136	1,523	1,659	-0,305	-10,813	-11,119
10	-0,743	6,667	5,924	0,222	-2,741	-2,518	-0,250	-16,485	-16,734
12	0,000	0,000	0,000	0,267	-1,800	-1,533	-0,201	-19,900	-20,101

Эпюры имеют вид

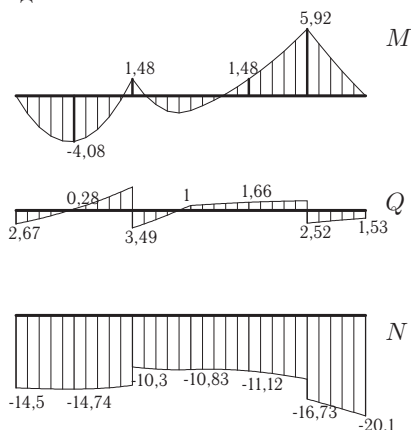


Рис. 323

Маплет для расчета двухшарнирной арки параболической формы приведен на с. 468.

4.13. Устойчивость

Задача 68. Найти критическую силу P центрально сжатого стержня длиной $l = 3,8$ м. Известно симметричное поперечное сечение стержня, составленное из двутавра № 22, ГОСТ 8239-89, и двух уголков

№ 10/7 ГОСТ 8509-89 (рис. 324), и схема закрепления (рис. 325). Модуль упругости материала $E = 2 \cdot 10^5$ МПа. При гибкости, меньшей предельной $\lambda_{\text{пред}} = \pi \sqrt{E/\sigma_{\text{пц}}}$, пользоваться формулой Ясинского¹: $\sigma_{\text{кр}} = a - b\lambda$, где $a = 310$ МПа, $b = 1,14$ МПа, $\sigma_{\text{пц}} = 195$ МПа.

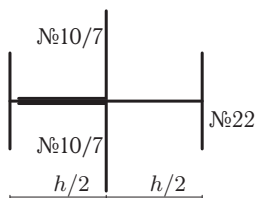


Рис. 324

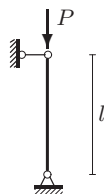


Рис. 325

Решение

1. Вычисляем площадь сечения. Рассмотрим уголок № 10 с толщиной $t = 7$ мм, т. е. уголок 10/7 (табл. 11, с. 489). Пронумеруем части сечения. Пусть двутавр имеет номер 1, уголки 2 и 3. Площадь его сечения согласно данным сортамента равна $13,8 \text{ см}^2$. Аналогично площадь сечения двутавра № 22 (табл. 9, с. 487) равна $30,6 \text{ см}^2$. Площадь всего сечения: $F = F_1 + 2F_2 = 30,6 + 2 \cdot 13,8 = 58,2 \text{ см}^2$.

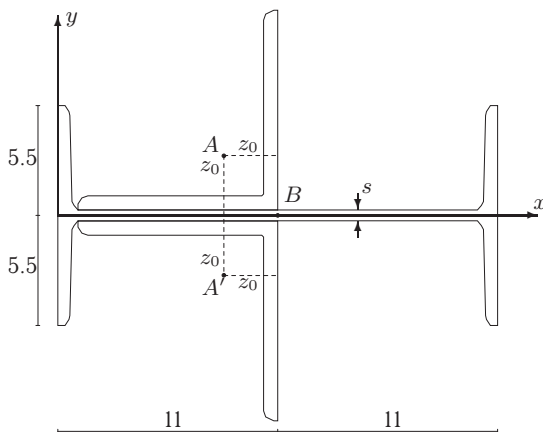


Рис. 326

¹Ясинский Феликс Станиславович (1856–1899) — русский инженер, механик, профессор Петербургского института инженерных путей сообщения.

2. Вычисляем координаты центра тяжести сечения. Выбираем оси координат (рис. 326). Начало координат помещаем на оси симметрии. В симметричном относительно оси x сечении $y_c = 0$. Центр тяжести двутавра находится в его геометрическом центре, $x_1 = 11$ см. Размер уголка $z_0 = 2,71$ см возьмем из таблицы 11. Внимательно следим за единицами измерения размеров в таблицах сортамента. Одни размеры в этих таблицах обычно даны в миллиметрах, другие, в том числе и z_0 , — в сантиметрах.

По формуле (4.2) найдем координату

$$x_c = \frac{F_1 x_1 + 2F_2 x_2}{F} = \frac{2 \cdot 13,8 \cdot (11 - 2,71) + 30,6 \cdot 11}{58,2} = 9,715 \text{ см.}$$

3. Определяем моменты инерции сечения относительно центральных осей. Значения моментов инерции берем из таблиц сортамента. Особое внимание обращаем на соответствие осей в условии задачи и на рисунке сортамента. В данном случае для двутавра момент инерции $J_{x_{c1}}$ относительно оси x в сортаменте обозначен как $J_y = 157 \text{ см}^4$, и, наоборот, момент инерции $J_{y_{c1}}$ относительно оси y в сортаменте обозначен как $J_x = 2550 \text{ см}^4$.

$$J_{x_c} = \sum_{i=1}^3 (J_{x_{c_i}} + (y_c - y_i)^2 F_i) = J_{x_{c1}} + 2 (J_{x_{c2}} + F_2 (z_0 + s/2)^2) = \\ = 157 + 2 (131 + 13,8 (2,71 + 0,27)^2) = 664,099 \text{ см}^4;$$

$$J_{y_c} = \sum_{i=1}^3 (J_{y_{c_i}} + (x_c - x_i)^2 F_i) = 2550 + 30,6 \cdot (11 - 9,715)^2 + \\ + 2 \cdot (131 + 13,8 \cdot (11 - 2,71 - 9,715)^2) = 2918,573 \text{ см}^4.$$

4. Находим наименьший момент инерции $J_{\min} = J_{x_c} = 664,099 \text{ см}^4$ и соответствующий радиус инерции $i = \sqrt{J_{\min}/F} = 3,378 \text{ см}$. При потере устойчивости сечение повернется вокруг оси x .

5. Вычисляем гибкость стержня с учетом коэффициента приведения¹. Для шарнирно опертого стержня $\mu = 1$. Таким образом, получаем гибкость

$$\lambda = \mu l / i = 1 \cdot 380 / 3,378 = 112,494.$$

Гибкость — безразмерная величина, характеризующая соотношение длины и толщины стержня.

6. Если гибкость стержня больше $\lambda_{\text{пк}}$ (длинные тонкие стержни), критическое напряжение вычисляем по формуле Эйлера $\sigma_{\text{кр}} = \pi^2 E / \lambda^2$.

¹Четыре варианта закрепления стержня см. в интерфейсе маплета на рис. 451, с. 472.

Иначе пользуемся формулой Ясинского. Вычисляем $\lambda_{\text{пц}}$

$$\lambda_{\text{пц}} = \pi \sqrt{E/\sigma_{\text{пц}}} = 3,141 \sqrt{2 \cdot 10^5 / 195} = 100,611. \quad (4.80)$$

Так как $\lambda > \lambda_{\text{пц}}$, используем формулу Эйлера

$$\sigma_{\text{кр}} = \pi^2 E / \lambda^2 = 194,288 \cdot 10^3 \text{ МПа}. \quad (4.81)$$

7. Вычисляем критическую силу $P_{\text{кр}} = \sigma_{\text{кр}} F = 1130,754 \text{ кН}$.

Для того чтобы продемонстрировать вычисления по формуле Ясинского, немного изменим условие задачи. Пусть длина стержня будет меньше: $l = 3 \text{ м}$. Гибкость при этом уменьшится:

$$\lambda = \mu l / i = 1 \cdot 300 / 3,378 = 88,81.$$

В этом случае $\lambda < \lambda_{\text{пц}}$ и стержень потеряет устойчивость в пластической области деформирования. Используем эмпирическую формулу Ясинского¹. Подставляя в нее заданные в условии задачи коэффициенты², получаем критическое напряжение

$$\sigma_{\text{кр}} = a - b\lambda = 310 - 1,14 \cdot 88,81 = 208,757 \text{ МПа}$$

и критическую силу

$$P_{\text{кр}} = \sigma_{\text{кр}} F = 1214,963 \text{ кН}.$$

Программа расчета критической нагрузки оформлена в виде маплета на с. 473.

Задача 69. Шарнирно закрепленный по концам чугунный стержень длиной $l = 1,1 \text{ м}$ сжимается силой $P = 100 \text{ кН}$. Требуется найти размеры квадратного сечения стержня при $[\sigma] = 130 \text{ МПа}$ и коэффициент запаса устойчивости.

Решение

Эта задача называется проектировочным расчетом на устойчивость. Будем пользоваться формулой

$$P_{\text{кр}} = [\sigma] F \varphi,$$

где $[\sigma]$ — допускаемое напряжение (см. табл. 12, с. 490), φ — коэффициент продольного изгиба (см. табл. 13, с. 490). Это нелинейное уравнение относительно размера квадратного сечения b . Разрешить его относительно b нельзя, так как коэффициент продольного изгиба φ зависит от гибкости, и зависимость дана в табличном виде, а гибкость, в свою очередь, зависит от поперечного размера сечения. Решим это уравнение методом последовательных приближений.

¹Иногда [14] эту формулу называют по имени Л. Тетмайера (*Ludwig von Tetmajer*, 1850–1905).

²Для других материалов константы даны в табл. 5.

В соответствии с условием закрепления (см. рис. 451, с. 472) коэффициент приведения длины $\mu = 1$. Площадь квадрата вычисляем по формуле $F = b^2$, момент инерции $J_{\min} = b^4/12$. Для первой попытки примем $\varphi_0 = 0,5$.

1) Площадь сечения

$$F = \frac{P}{[\sigma]\varphi_0} = \frac{100 \cdot 10^3}{130 \cdot 10^6 \cdot 0,5} \cdot 10^4 = 15,38 \text{ см}^2.$$

Зная площадь, находим сторону квадрата: $b = \sqrt{F} = \sqrt{15,38} = 3,92$ см и радиус инерции сечения: $i = \sqrt{J_{\min}/F} = \sqrt{(b^4/12)/b^2} = b/\sqrt{12} = 1,13$ см. Вычисляем гибкость: $\lambda = l\mu/i = 110 \cdot 1/1,13 = 97,15$. В таблице 13 такого значения нет. Линейно интерполируя значения таблицы между 0,2 при $\lambda = 90$ и 0,16 при $\lambda = 100$, находим

$$\varphi_1 = 0,2 - \frac{0,2 - 0,16}{10}(97,15 - 90) = 0,171.$$

Относительная разница

$$\frac{|\varphi_1 - \varphi_0|}{\varphi_0} = \frac{|0,171 - 0,5|}{0,5} = 0,66,$$

что больше обычно принятой невязки в 5%.

Примем $\varphi_0 = (0,171 + 0,5)/2 = 0,336$ и повторим расчет.

2) Площадь сечения

$$F = \frac{P}{[\sigma]\varphi_0} = \frac{100 \cdot 10^3}{130 \cdot 10^6 \cdot 0,336} \cdot 10^4 = 22,91 \text{ см}^2.$$

Сторона квадрата $b = \sqrt{F} = \sqrt{22,91} = 4,79$ см. Радиус инерции $i = b/\sqrt{12} = 1,38$ см. Найдем гибкость $\lambda = l\mu/i = 110 \cdot 1/1,38 = 79,6$. Интерполируя по таблице между 0,34 при $\lambda = 70$ и 0,26 при $\lambda = 80$, найдем

$$\varphi_1 = 0,34 - \frac{0,34 - 0,26}{10}(79,6 - 70) = 0,263.$$

Относительная разница

$$\frac{|\varphi_1 - \varphi_0|}{\varphi_0} = \frac{|0,263 - 0,336|}{0,336} = 0,22 > 0,05.$$

Примем $\varphi_0 = (0,263 + 0,336)/2 = 0,3$ и продолжим расчет.

3) Площадь сечения

$$F = \frac{P}{[\sigma]\varphi_0} = \frac{100 \cdot 10^3}{130 \cdot 10^6 \cdot 0,3} \cdot 10^4 = 25,69 \text{ см}^2.$$

Сторона квадрата $b = \sqrt{F} = \sqrt{25,69} = 5,07$ см.

Радиус инерции $i = b/\sqrt{12} = 1,46$ см. Найдем гибкость: $\lambda = l\mu/i = 110 \cdot 1/1,46 = 75,18$. Интерполируя по таблице между 0,34 при $\lambda = 70$ и 0,26 при $\lambda = 80$, найдем

$$\varphi_1 = 0,34 - \frac{0,34 - 0,26}{10}(75,18 - 70) = 0,298.$$

Относительная разница

$$\frac{|\varphi_1 - \varphi_0|}{\varphi_0} = \frac{|0,298 - 0,3|}{0,3} = 0,5 \cdot 10^{-2} < 0,05.$$

Таким образом, сечение имеет размер $b = 5,07$ см и гибкость $\lambda = 75,18 < 100$.

Для определения критической силы используем формулу Ясинского для чугуна¹. Для чугуна имеем $a = 776$ МПа, $b = 12$ МПа, $c = 0,053$ МПа.

$$P_k = F(a - b\lambda + c\lambda^2) = 445,46 \text{ кН.}$$

Коэффициент запаса устойчивости $n_y = P_k/P = 4,45$.

Таблица 5

Константы формулы Ясинского

Материал	a	b
Ст. 2, Ст.3	310	1,14
Ст.5	464	3,26
Сталь 40	321	1,16
Кремнистая сталь	589	3,82
Дерево	29,3	0,194

Maple–программа подбора сечения стержня из условия устойчивости приведена на с. 476.

4.14. Стабильность

Нестабильность процесса, описываемого дифференциальным уравнением, определяется вырождением связи между приращениями производных функции различных порядков. Явление неустойчивости мо-

¹В отличие от стали и дерева здесь нелинейная зависимость.

жет быть связано с условной бифуркацией процесса¹, выпучиванием конструкций из реологических материалов или с неуправляемым численным моделированием, основанном на пошаговом и итерационном алгоритме. Условие неустойчивости соответствует объективным свойствам уравнения процесса и получается из них однозначной процедурой. Выбор порядка неустойчивости (или порядка производных задаваемых возмущений) условен, для полного анализа желательно проанализировать наибольшее разумное число вариантов возмущений. Возможный порядок неустойчивости определяется порядком дифференциального уравнения. Если уравнение (линейное или нелинейное) имеет порядок n , то порядок неустойчивости содержит n натуральных чисел k_1, k_2, \dots, k_n . Сюда может быть включен и ноль. Порядок неустойчивости уравнения второго порядка имеет вид (m/n) , для уравнение первого порядка порядок неустойчивости характеризуется одним числом (m) .

Постановка задачи. *Найти условие неустойчивости порядка (m/n) динамического процесса, описываемого обыкновенным дифференциальным уравнением второго порядка.*

План решения

Для дифференцируемой достаточное число раз функции $x(t)$ имеем дифференциальное уравнение второго порядка.

1. Даем функции $x(t)$, скорости $\dot{x}(t)$ и ускорению $\ddot{x}(t)$ малые независимые приращения. Линеаризуем уравнение. Получаем первое уравнение в приращениях.

2. Второе уравнение в приращениях получаем дифференцированием первого по времени.

3. Записываем систему двух уравнений в матричном виде, относя в правую часть приращения производных порядка m и n .

4. Приравниваем определитель системы нулю. Полученное уравнение является условием неустойчивости. Подставляем в условие заданное значение скорости. Определяем критическую координату. Если в

¹Простой пример условной бифуркации. Пусть некоторый квазистатистический процесс (медленный, поэтому силы инерции не принимаются во внимание) описывается линеаризованным уравнением $a(x, \dot{x})\Delta x + b(x, \dot{x})\Delta \dot{x} = 0$. Если положить $\Delta \dot{x} = 0$, то при $a(x, \dot{x}) = 0$ уравнение допускает ненулевое решение для приращения Δx (функция $x(t)$ получает приращение). Под функцией времени $x(t)$ можно понимать прогиб первоначально прямолинейной конструкции (стержень, пластина, оболочка). Условие $\Delta x \neq 0$ означает выпучивание конструкции. Но согласно принятому предположению это произойдет так, что $\Delta \dot{x} = 0$. Из $a(x, \dot{x}) = 0$ можно найти соответствующее критическое время. Бифуркацию x при условии $\Delta \dot{x} = 0$ В. Д. Ключников назвал псевдобифуркацией нулевого порядка. Псевдобифуркации высших порядков соответствуют скачкам производных высших порядков [31]. См. также задачу 73, с. 221.

условие неустойчивости входят вторые и третьи производные, то их надо выразить из исходного уравнения и его производной.

Задача 70. Дано уравнение некоторого нелинейного динамического процесса¹:

$$16\ddot{x}\sqrt{\dot{x}} + b\dot{x} + x^4 = 0. \quad (4.82)$$

Определить условие неустойчивости процесса порядка (0/3).

Решение

1. Даем функции $x(t)$, скорости $\dot{x}(t)$ и ускорению $\ddot{x}(t)$ малые независимые приращения. Линеаризуем уравнение²

$$16\sqrt{\dot{x}}\Delta\ddot{x} + (8\ddot{x}/\sqrt{\dot{x}})\Delta\dot{x} + b\Delta\dot{x} + 4x^3\Delta x = 0. \quad (4.83)$$

2. Дифференцируем линеаризованное уравнение по времени³

$$16\sqrt{\dot{x}}\Delta\ddot{\dot{x}} + (16\ddot{x}/\sqrt{\dot{x}})\Delta\dot{\dot{x}} + (8\ddot{\dot{x}}/\sqrt{\dot{x}})\Delta\dot{x} - \\ - (4\ddot{x}^2/(\dot{x}\sqrt{\dot{x}}))\Delta\dot{x} + b\Delta\ddot{\dot{x}} + 4x^3\Delta\dot{x} + 12x^2\dot{x}\Delta x = 0. \quad (4.84)$$

3. Записываем систему полученных уравнений (4.83)–(4.84) относительно приращений $\Delta\dot{x}$ и $\Delta\ddot{\dot{x}}$ в матричном виде, относя в правую часть приращения производных порядка 0 и 3:

$$\begin{bmatrix} 8\ddot{x}/\sqrt{\dot{x}} + b & 16\sqrt{\dot{x}} \\ 8\ddot{\dot{x}}/\sqrt{\dot{x}} - 4\ddot{x}^2/(\dot{x}\sqrt{\dot{x}}) + 4x^3 & 16\ddot{x}/\sqrt{\dot{x}} + b \end{bmatrix} \cdot \begin{bmatrix} \Delta\dot{x} \\ \Delta\ddot{\dot{x}} \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix},$$

где $\alpha_1 = -4x^3\Delta x$, $\alpha_2 = -12x^2\dot{x}\Delta x - 16\sqrt{\dot{x}}\Delta\ddot{x}$.

4. Приравниваем определитель системы нулю. Полученное уравнение является условием неустойчивости:

$$128\ddot{x}\dot{x} + 4x^3\dot{x}\sqrt{\dot{x}} - 192\ddot{x}^2 - 24b\ddot{x}\sqrt{\dot{x}} = 0. \quad (4.85)$$

В условие неустойчивости (4.85) входят вторые и третьи производные. Выражаем ускорение из исходного уравнения (4.82)

$$\ddot{x} = -\frac{b\dot{x} + x^4}{16\sqrt{\dot{x}}}. \quad (4.86)$$

¹Например, дифференциальное уравнение движения точки массой 16 под действием силы, зависящей от скорости и положения точки: $16\ddot{x} = -x^4/\sqrt{\dot{x}} - b\sqrt{\dot{x}}$. Классификация этого уравнения методами Maple см. на с. 303.

²В общем случае линеаризация функции n переменных $F(u_1, u_2, \dots, u_n)$ имеет вид $\sum_{i=1}^n F'_i \Delta u_i$, где F'_i — частная производная по i -му аргументу.

³Или дважды дифференцируем исходное уравнение, а затем линеаризуем. Операторы дифференцирования и линеаризации коммутативны.

Дифференцируем по времени (4.86):

$$\ddot{x} = \frac{(b\dot{x} + x^4)\dot{x}}{32\dot{x}\sqrt{\dot{x}}} - \frac{b\ddot{x} + 4x^3\dot{x}}{16\sqrt{\dot{x}}}. \quad (4.87)$$

Подставляем (4.86), (4.87) в условие (4.85). Получаем после преобразований

$$x^3(32\dot{x}^2\sqrt{\dot{x}} - x^5) = 0.$$

При заданном значении скорости $\dot{x} = 1$ определяем два значения критической координаты $x_1 = 2$, $x_2 = 0$. Для этих значений динамическая система неустойчива по отношению к возмущению производных порядка 0 и 3, т. е. как угодно малое изменение этих величин приводит к неограниченному росту скоростей и ускорений. С другой стороны, это же явление можно рассматривать как бифуркацию скоростей и ускорений при условии $\Delta x = 0$, $\Delta x^{(3)} = 0$.

Отметим, что решение не зависит от параметра b . Для других порядков особых точек параметр b войдет в решение. Например, условием неустойчивости порядка (0/2) является $x = (\dot{x}b)^{1/4}$, в чем легко убедиться с использованием Maple-программы определения условия неустойчивости нелинейного обыкновенного дифференциального уравнения второго порядка на с. 477.

Задача 71. Динамический процесс описывается системой дифференциальных уравнений

$$\ddot{x}x^2 + 3\ddot{y}y + 2\dot{x}y + 2\dot{x}x + 3\dot{y}y = 0, \quad (4.88)$$

$$\ddot{x}x + 7\ddot{y}y + 4x + 3y = 0. \quad (4.89)$$

Найти условие неустойчивости порядка (1/2). Вывести уравнение кривой неустойчивости при $\dot{x} = 1$, $\dot{y} = 0$.

План решения

1. Линеаризуем систему уравнений относительно приращений функций Δx , Δy , их скоростей $\Delta \dot{x}$, $\Delta \dot{y}$ и ускорений $\Delta \ddot{x}$, $\Delta \ddot{y}$.

2. Находим матрицу системы линейных относительно $\Delta \dot{x}$, $\Delta \dot{y}$ уравнений для точки неустойчивости порядка (0/2) или относительно Δx , Δy для точки неустойчивости порядка (1/2).

3. Приравнявая нулю определитель полученной матрицы, записываем искомое условие неустойчивости. Если в это условие входят вторые производные функций, вычисляем их из заданной системы дифференциальных уравнений. Для получения окончательного выражения для условия подставляем в него значения заданных скоростей.

Решение

1. Линеаризуем систему уравнений относительно приращений функций их скоростей и ускорений:

$$\begin{aligned} (x^2 + 3y)\Delta\ddot{x} + (2x + 3y)\Delta\dot{x} + 2(\ddot{x}x + y + \dot{x})\Delta x + \\ + (3\ddot{x} + 2x + 3\dot{x})\Delta y = 0, \quad (4.90) \\ x\Delta\ddot{x} + 7x\Delta\dot{y} + (\ddot{x} + 7\dot{y} + 4)\Delta x + 3\Delta y = 0. \end{aligned}$$

2. Находим матрицу системы линейных относительно Δx , Δy уравнений

$$A = \begin{bmatrix} 2(\ddot{x}x + \dot{x} + y) & 3\ddot{x} + 3\dot{x} + 2x \\ \ddot{x} + 7\dot{y} + 4 & 3 \end{bmatrix} \quad (4.91)$$

3. Приравнявая нулю определитель полученной матрицы, записываем искомое условие неустойчивости:

$$6(\ddot{x}x + \dot{x} + y) - (3\ddot{x} + 3\dot{x} + 2x)(\ddot{x} + 7\dot{y} + 4) = 0. \quad (4.92)$$

Для получения зависимости $x \sim y$ исключаем отсюда скорости и ускорения. Подставляем в (4.92) выражение $\ddot{x} + 7\dot{y} + 4 = -3y/x$, найденное из уравнения (4.89), и ускорение $\ddot{x} = -(2xy + 2\dot{x}x + 3\dot{x}y)/(x^2 + 3y)$ из уравнения (4.88). После преобразований и упрощений при $\dot{x} = 1$, $\dot{y} = 0$ окончательно получаем

$$2x^2 + 3xy - 6y^2 = 0.$$

Очевидно, это уравнение описывает две прямые, проходящие через начало координат

$$y = \frac{3 \pm \sqrt{57}}{12} x.$$

На этих прямых, горизонтально пересекаемых со скоростью $\dot{x} = 1$ точкой, изображающей процесс, связь приращений Δx , Δy и их скоростей $\Delta\dot{x}$, $\Delta\dot{y}$ и ускорений $\Delta\ddot{x}$, $\Delta\ddot{y}$ вырождается.

Программа определения условия неустойчивости в системе **Maple** дана на с. 479. Там же приведены графики полученного решения.

Задача 72. Найти кривые неустойчивости второго порядка линейного дифференциального уравнения в частных производных

$$L = y(xy - 1)u_x + x(xy + 1)u_y + xy u = 0. \quad (4.93)$$

Неустойчивость второго порядка в данном случае соответствует вырождению связи между функцией $u(x, y)$, частными производными первого порядка u_x , u_y и производными второго порядка u_{xx} , u_{xy} , u_{yy} , принятыми за произвольные возмущенные величины. Исследуемые

уравнения линейны, поэтому дополнительная линеаризация не требуется. Принято, что $u(x, y)$ — дифференцируемая достаточное число раз функция.

План решения

1. Записываем систему из трех уравнений

$$\begin{aligned} L &= 0, \\ L_x &= 0, \\ L_y &= 0, \end{aligned} \quad (4.94)$$

где два последних уравнения получены дифференцированием первого.

2. Представляем полученную систему как систему линейных уравнений относительно u , u_x , u_y . Выписываем матрицу этой системы.

3. Приравниваем нулю определитель алгебраической системы, получаем уравнения кривых неустойчивости.

Решение

1. Записываем систему из трех уравнений

$$\begin{aligned} y(xy - 1)u_x + x(xy + 1)u_y + uyx &= 0, \\ (2xy + 1)u_y + (y^2 + xy)u_x + x(xy + 1)u_{x,y} + y(xy - 1)u_{x,x} + u_y &= 0, \\ (xy + x^2)u_y + (2xy - 1)u_x + x(xy + 1)u_{y,y} + y(xy - 1)u_{x,y} + ux &= 0, \end{aligned}$$

где два последних уравнения получены дифференцированием первого по x и y соответственно.

2. Представляем полученную систему как систему линейных уравнений относительно u , u_x , u_y . Выписываем матрицу этой системы

$$\begin{bmatrix} xy & y(xy - 1) & x(xy + 1) \\ y & y^2 + xy & 2xy + 1 \\ x & 2xy - 1 & xy + x^2 \end{bmatrix} \begin{bmatrix} u \\ u_x \\ u_y \end{bmatrix} = \begin{bmatrix} 0 \\ \alpha u_{xx} + \beta u_{xy} \\ \alpha u_{xy} + \beta u_{yy} \end{bmatrix}, \quad (4.95)$$

$\alpha = \alpha(x, y)$ и $\beta = \beta(x, y)$ — известные функции.

3. Приравниваем нулю определитель матрицы:

$$-xy(1 - y^2 + x^2) = 0.$$

Кривые неустойчивости определяются гиперболой

$$y^2 - x^2 = 1$$

и двумя прямыми, совпадающими с осями координат: $x = 0$, $y = 0$. На этих кривых как угодно малое возмущение величин, стоящих в правой части (4.95) — в данном случае кривизн решения, — приводит к неограниченному росту приращений функции u и ее частных производных.

Аналогичный алгоритм применим и для нелинейных уравнений, однако систему нужно записывать относительно приращений функции

и ее производных, и после линеаризации в систему войдут невозможные значения функции u , значения которых нужно в общем случае вычислять численно.

Maple–программа определения кривых неустойчивости уравнения в частных производных приведена на с. 481.

Задача 73. Найти точки неустойчивости второго порядка и критическое время продольно сжатого постоянной силой P , шарнирно опертого стержня из реологического материала

$$\dot{p}p^\alpha = f(\sigma), \quad (4.96)$$

$p = \varepsilon - \sigma/E$ — деформация ползучести, α — константа материала (коэффициент упрочнения), $f(\sigma)$ — заданная функция¹. Длина стержня l , рис. 327.

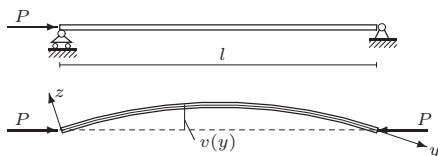


Рис. 327

Решение

Выведем сначала формулу Эйлера для критической нагрузки упругого стержня. Рассмотрим малое отклонение стержня от прямолинейного состояния. Деформации при этом получают приращения. Ось y материальной системы координат ("вмороженной" в тело) направим по срединной линии стержня, ось z — по нормали к ней. На основании гипотезы плоских сечений

$$\Delta\varepsilon = \Delta\varepsilon_0 + z\Delta v_{,yy}, \quad (4.97)$$

где $\Delta\varepsilon$ — приращение осевой деформации на расстоянии z от срединного сечения, $\Delta\varepsilon_0$ — приращение деформации срединного сечения, равные нулю, если в результате возмущения нагрузка на стержень не изменилась, Δv — приращение прогиба стержня, $\Delta v_{,yy}$ — вторая производная приращения прогиба по осевой координате y . Умножаем (4.97) на z и интегрируем по площади сечения. Так как приращение деформации срединной поверхности стержня $\Delta\varepsilon_0$ не зависит от z , а для симметричного относительно оси y сечения статический момент $\int z dF$

¹Показывает зависимость деформации ползучести от уровня напряжений (степенная, гиперболический синус и др.)

равен нулю, то получим

$$\int_F \Delta \varepsilon z dF = J \Delta v_{,yy}, \quad (4.98)$$

$J = \int_F z^2 dF$ — момент инерции поперечного сечения стержня. Уравнение равновесия (моментов) отсеченной части стержня относительно точки на срединной линии сечения дает

$$\int_F \Delta \sigma z dF = -P \Delta v, \quad (4.99)$$

где $P = \int_{\sigma} dF$ — нагрузка, действующая на стержень в продольном направлении. В случае линейной упругости приращения напряжений и деформаций связаны законом Гука

$$\Delta \sigma = E \Delta \varepsilon.$$

Умножим это уравнение на z и проинтегрируем по площади F . Интегралы от $z \Delta \varepsilon$ и $z \Delta \sigma$ выразим через (4.98) и (4.99). Получим

$$EJ \Delta v_{,yy} + P \Delta v = 0.$$

Форму прогиба выберем в виде, удовлетворяющем условию шарнирного опирания по концам ($\Delta v = 0$ при $y = 0$ и $y = l$):

$$\Delta v = u \sin ky, \quad (4.100)$$

где k — параметр волнообразования, $k = m_1 \pi / l$, m_1 — число полуволн по длине стержня. Условия отсутствия моментов на опорах $\Delta v_{,yy} = 0$ при этом удовлетворяются автоматически. Получим

$$u \sin ky (-k^2 EJ + P) = 0.$$

Из условия $u \neq 0$ найдем критическую нагрузку (формула Эйлера¹) $P_0 = k^2 EJ$. Соответствующее напряжение обозначим $\sigma_0 = k^2 EJ / F$. Пусть стержень выполнен из реологического материала (4.96). В результате отклонения от прямолинейного положения приращения напряжений и деформаций ползучести, согласно (4.96), удовлетворяют соотношению

$$\Delta \dot{p} p^\alpha + \alpha \Delta p \dot{p} p^{\alpha-1} = f'(\sigma) \Delta \sigma, \quad (4.101)$$

где p , \dot{p} и σ относятся к оси стержня, а Δp , $\Delta \dot{p}$ и $\Delta \sigma$ означают малые приращения соответствующих величин по толщине стержня.

¹Leonhard Euler (1707–1783) — российский и немецкий математик, по происхождению швейцарец.

Введем обозначения безразмерного напряжения, отнесенного к Эйлеровому значению:

$$\omega = \sigma/\sigma_0,$$

и параметра, зависящего только от напряжений,

$$\zeta = (f'/f)E\omega/(1 - \omega). \quad (4.102)$$

Из (4.99) с учетом выражения $\Delta p = \Delta \varepsilon - \Delta \sigma/E$ и уравнения (4.98) следует

$$\int_F \Delta p z dF = -P \Delta v/E + J \Delta v_{,yy}.$$

Умножим (4.101) на z и проинтегрируем по площади F :

$$p^\alpha \int \Delta \dot{p} z dF + \alpha \dot{p} p^{\alpha-1} \int \Delta p z dF = f'(\sigma) \int \Delta \sigma z dF.$$

Разделим обе части уравнения на p^α и, учитывая, что величины без значков Δ не зависят от z , с помощью (4.99) получим отсюда

$$J \Delta \dot{v}_{,yy} + P \Delta \dot{v}/E + \dot{p}(P \Delta v/E + J \Delta v_{,yy})\alpha/p = -P \Delta v f' p^{-\alpha}. \quad (4.103)$$

Если выбирать форму прогиба в виде синусоиды (4.100), то из (4.103) следует, что скорость прогиба должна иметь такой же вид $\Delta \dot{v} = \dot{u} \sin ky$, при этом $\Delta \dot{v}_{,yy} = -k^2 \dot{u} \sin ky$. Заменяв в правой части p^α на f/\dot{p} , получим

$$(-k^2 J + P/E)\dot{u} + \dot{p}(P/E - k^2 J)u\alpha/p = -P u \dot{p} f'/f.$$

Эйлерова нагрузка упругого стержня $P_0 = E J k^2$. Делим последнее равенство на $P/E - k^2 J = P_0(\omega - 1)/E$:

$$\dot{u} + u \dot{p}(\alpha/p - (f'/f)E\omega/(1 - \omega)) = 0.$$

Пользуясь обозначением (4.102), получим уравнение для u :

$$\dot{u} p + u \dot{p}(\alpha - p\zeta) = 0. \quad (4.104)$$

Точка неустойчивости второго порядка соответствует вырождению связи приращений прогиба, скорости прогиба с заданным (как угодно малым) приращением ускорения [59]. Искусственно поднимаем порядок (4.104), дифференцируя его по времени. Помня, что по условию задачи нагрузка постоянная, $\zeta = \text{const}$, а деформация ползучести удовлетворяет (4.96), после некоторых преобразований¹ получим

$$\ddot{u} p^2 + \dot{u} p \dot{p}(\alpha - p\zeta) + u \dot{p}^2 \alpha(p\zeta - \alpha - 1) = 0. \quad (4.105)$$

¹Программа 55, с. 483.

Введем обозначение $\xi = p\zeta$. Систему (4.104), (4.105) запишем в матричном виде

$$A\bar{U} = \bar{V}, \quad (4.106)$$

где

$$A = \begin{bmatrix} \dot{p}(\alpha - \xi) & \\ -\alpha \dot{p}^2(\alpha - \xi + 1) & \dot{p}p(\alpha - \xi) \end{bmatrix}, \quad \bar{U} = \begin{bmatrix} u \\ \dot{u} \end{bmatrix}, \quad \bar{V} = \begin{bmatrix} 0 \\ p^2 \ddot{u} \end{bmatrix}.$$

Равенство нулю определителя матрицы A соответствует неустойчивости второго порядка:

$$\dot{p}\xi^2 - 3\dot{p}\alpha\xi + \dot{p}\alpha(2\alpha + 1) = 0. \quad (4.107)$$

Находим корни уравнения

$$\xi = \frac{3\alpha \pm \sqrt{\alpha^2 - 4\alpha}}{2}. \quad (4.108)$$

Это решение существует при $\alpha > 4$. Для того чтобы найти критическое время, проинтегрируем (4.96) при постоянных нагрузках и нулевых начальных условиях:

$$t = p^{\alpha+1}/((\alpha + 1)f).$$

Соответствующую деформацию ползучести подставим сюда из (4.108), взяв меньшее значение:

$$p = \frac{3\alpha - \sqrt{\alpha^2 - 4\alpha}}{2\zeta}.$$

С увеличением нагрузки критическое время уменьшается. При $P \rightarrow P_0$ параметр ζ (4.102) стремится к бесконечности, а критическое время к нулю, что соответствует физическому смыслу задачи. Эйлера нагрузка приводит к мгновенному выпучиванию стержня.

Аналогично можно получить точки неустойчивости других порядков. Если точка неустойчивости второго порядка является корнем полинома (4.107), который обозначим за Q_2 , для точки третьего порядка (задается возмущение третьей производной, а прогиб, скорость и ускорение прогиба неограниченно растут в момент возмущения) найдем полином Q_3 как определитель матрицы 3×3 . Выпишем первые четыре полинома:

$$\begin{aligned} Q_1 &= \xi - \alpha, \\ Q_2 &= \xi^2 - 3\xi\alpha + \alpha(2\alpha + 1), \\ Q_3 &= \xi^3 - 6\alpha\xi^2 + \alpha(4 + 11\alpha)\xi - \alpha(2\alpha + 1)(3\alpha + 2), \\ Q_4 &= \xi^4 - 10\alpha\xi^3 + 5\alpha(2 + 7\alpha)\xi^2 - 5\alpha(2\alpha + 1)(5\alpha + 2)\xi + \\ &\quad + \alpha(2\alpha + 1)(3\alpha + 2)(4\alpha + 3). \end{aligned}$$



Имеется дифференциальная рекуррентная зависимость

$$Q_N = Q_{N-1}(\xi + 1 - N(1 + \alpha)) + Q'_{N-1}\xi, \quad N = 2, 3, 4, \dots,$$

где штрихом обозначена производная по ξ . Корни полиномов для определения точек неустойчивости порядков 1–7 в зависимости от коэффициента упрочнения α изображены на рис. 328. Полиномы нечетных порядков имеют корни для любых α , четных — при достаточно больших коэффициентах упрочнения. Для построения кривых в **Maple** не потребовалось искать корни, в этом случае оказался особо удобным оператор `implicitplot`¹ для изображения неявных функций. Точка неустойчивости первого порядка $\xi = \alpha$ совпадает с критерием Работнова–Шестерикова [31].

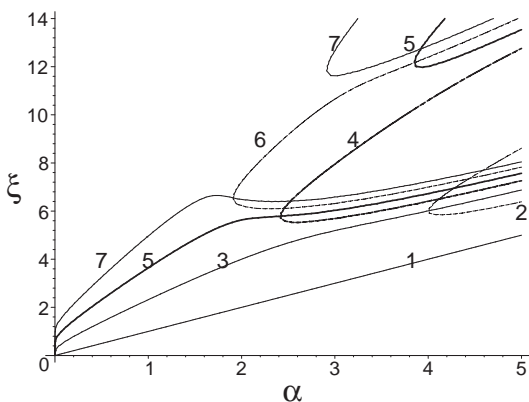


Рис. 328

Maple–программа определения точек неустойчивости различных порядков процесса сжатия реологического стержня приведена на с. 483.

¹С. 273.

ВВЕДЕНИЕ В MAPLE

Математическая система **Maple** создана как мощный инструмент для аналитических и численных математических расчетов. Она может быть использована в научных и учебных целях. В **Maple** можно решать алгебраические, дифференциальные и рекуррентные уравнения, брать интегралы, строить графики. Число функций, операторов, различных специализированных пакетов в этой системе огромно. Полная документация к **Maple**, вероятно, существует только в электронном виде — во встроенной системе помощи. На русском языке есть справочная и учебная литература по различным версиям этой системы [3, 7, 10, 11, 16–18, 21, 29, 37, 42, 54, 57]. Надо заметить, что от версии к версии система **Maple** развивается, к ней добавляются новые пакеты и функции, расширяются возможности численного и аналитического счета, улучшается совместная работа с другими системами. Однако на элементарном уровне язык программирования и основные принципы практически не меняются. Поэтому если говорить о литературе по **Maple**, то для начинающих пользователей не стоит обращать внимание на номер версии, объявленной в той или иной книге. Так, например, книга Б. М. Манзона [37], написанная для **Maple V**, вполне актуальна и для 13-й версии, а обширную справочную литературу В. П. Дьяконова стоит держать на столе независимо от того, какую версию она описывает. Из списка литературы следует также отметить фундаментальный труд по уравнениям математической физики Голоскокова Д.П. [11], и книгу Соловьева М.Е. (к сожалению, труднодоступную) с решениями задач механики сплошной среды [54].

Будем считать, что программа установлена по умолчанию в каталог `C:\Program Files\Maple 13`. В последних версиях **Maple** предусмотрены две графические оболочки для работы в системе. Как первая `swmaple.exe`, так и вторая `maplew.exe` расположены в каталоге `\Maple 13\bin.win`. Программы отличаются в основном способом ввода и отображения информации. Они обе обращаются к одним и тем же процедурам, и результаты вычислений почти не отличаются. Программа `swmaple.exe` (**Maple** классическая, или **Classic Worksheet**) по своему интерфейсу совпадает с прежними версиями **Maple** и удобна для пользователей, привыкших работать с командной строкой. Кроме того, она менее требовательна к памяти и может работать как в Windows, так и в системах типа UNIX. Современная версия `maplew.exe` по умолчанию считается основной и называется **Standard Worksheet**.



При инсталляции ярлык именно этой программы выносится на рабочий стол, и пользователь, как правило, начинает работать в ней, не подозревая о существовании **Classic Worksheet**. Версия **Standard Worksheet** немного медленней загружается, но скорость счета обеих программ, вероятно, одинаковая. В большинстве случаев разница замечена не была. Существенным плюсом программы `maplew.exe` является возможность пользоваться различными современными инструментами аналитических вычислений и программирования. Например, доступна удобная программа визуального программирования маплетов `Maplet Builder`, о которой пойдет речь в гл. 6, с. 335, анализатор обыкновенных дифференциальных уравнений `ODE Analyzer`¹ и другие средства, работающие по принципу маплетов. Программы **Standard Worksheet** и **Classic Worksheet** отличаются и расширением своих рабочих файлов.

Классическая оболочка работает с расширением `mws`, принятым еще в первых версиях 80-х и 90-х годов, современная графическая — с расширением `mw` и `mws`. Переключить ассоциацию расширений файлов можно с помощью программы `\Maple 13\bin.win\ToggleAssociation.exe` (рис. 329).



Рис. 329

Для пользователей `swmaple.exe` рекомендуем сразу зайти в `\Maple 13\bin.win` и правой клавишей мыши вынести на рабочий стол ярлык этой программы — кленовый лист желтого цвета с цифрой 13. Ярлык программы `maplew.exe` такой же, но красного цвета.

5.1. Ввод информации и простейшие вычисления

Запустив программу `swmaple.exe`, пользователь видит экран с мигающим курсором на пустой строке, начинающейся со знака `[>`. Это знак приглашения для ввода информации. Форма его идет еще со времен DOS, уже забытой операционной системы, следы от которой можно

¹Интерактивную программу `ODE Analyzer` можно вызвать и из **Classic Worksheet**, см. с. 296.

наблюдать и в современной Windows (см. кнопку "Пуск", команду "Выполнить..", cmd). Команда `interface(prompt="* ")` позволяет при необходимости заменить стандартное приглашение "галочка" на любое другое, в данном случае на * и два пробела. В бесплатной демонстрационной версии программы Maple V Release 4¹ в качестве приглашения стоит слово DEMO (рис. 330).

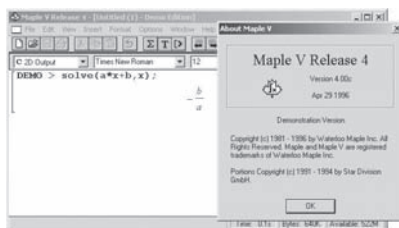


Рис. 330

После приглашения можно ввести вычисляемое выражение или некоторую функцию, или команду, например, `[> sin(Pi);`. В конце надо поставить двоеточие или точку с запятой и нажать клавишу Enter. Если в конце введенного выражения стоит точка с запятой, то результат можно сразу увидеть на экране, если двоеточие — то команда будет выполнена, но результат на экран не выводится. Часто это используется для того, чтобы не загромождать экран промежуточными вычислениями. Слева от приглашающей галочки стоит квадратная скобка. Это символ блока вычислений. При нажатии Enter выполняются все действия в блоке, появляется результат и новый, еще пустой блок `[>`. Если необходимо выполнить сразу несколько операций, то блоки можно объединять. Для этого служит функциональная клавиша F4. Ставим курсор в блок, нажимаем F4, и текущий блок объединяется со следующим. Можно нажать F4 несколько раз, тогда объединятся несколько блоков. Иногда это необходимо при программировании. Команды в одном операторе цикла, например, не могут быть в разных блоках. Это же относится и к процедурам, модулям, условным операторам и т. д. Поставить разрыв в блоке можно с помощью клавиши F3.

Операция присваивания (не путать с равенством!) обозначена как в языке Pascal — двоеточием с равенством. Ввод зависит от регистра букв. За этим надо следить. Например, выражения `[> Pi` и `[> pi` имеют разный смысл. Это же относится и ко всем встроенным функциям, константам и операторам.

¹www.exponenta.ru или <http://vuz.exponenta.ru/MVR4DEMO.rar>

Вот, например, как выглядит простейшая операция возведения суммы a и b в квадрат:

```
> x:=(a+b)^2;
> evalm(x);

      x := (a + b)2
      a2 + 2 a b + b2
```

Здесь объединены в один блок две команды — присваивания выражения и операции раскрытия скобок `evalm`. Возведение в степень можно обозначить также как `**`. Если степень отрицательная или является некоторым выражением, то показатель следует заключить в скобки, иначе возможны досадные ошибки. Например, $a^{1/3}$ даст результат $\frac{a}{3}$, а чтобы извлечь кубический корень (что, вероятно, и предполагалось) степень надо было бы записать в скобках: $a^{(1/3)}$.

В дальнейшем для упрощения записи символ блока — квадратные скобки слева — ставить в текстах программ не будем, тем более что выбор блоков не меняет результат вычисления, если, конечно, не ставить разрыв там, где это нельзя (например, внутри операторов цикла, условных операторов или в процедурах).

В главе 7 приведены программы для различных расчетов. Обратим внимание, что все они начинаются с команды обнуления регистров памяти `restart`. После `restart` можно ставить точку или точку с запятой, это безразлично. Но следует следить за реакцией программы. Если после выполнения этой команды на экране не появилась какая-нибудь надпись типа `restrt`, то все в порядке, команда сработала. Если же в записи команды допущена ошибка (здесь для примера пропущена буква `a`), то **Maple** воспринимает эту запись как некую пользовательскую переменную, выведенную на экран. В общем такая ошибка совсем не опасна, если переменные программы уникальны и ранее при запуске **Maple** не использовались, но лучше этого избегать. Чаще всего начинающие пользователи пишут `Restart`, а **Maple** — регистр чувствительная программа и различает строчные и прописные буквы (за редким исключением) в написании команд и переменных.

5.2. Программирование

Систему **Maple** можно использовать просто для вычисления или преобразования каких-то выражений, вычисления производных или интегралов, решения уравнений, построения графиков и т. д. Однако в систему заложено значительно больше. Встроенный язык программирования, напоминающий известные языки `C`, `Pascal`, позволяет писать собственные программы с вводом данных, выводом результатов, достаточно развитым интерфейсом. Для этого в языке **Maple** предусмотрены

все полагающиеся операторы: операторы цикла, условные операторы и т. п. К сожалению, пользовательские программы, созданные в **Maple**, не автономны, exe-файлы **Maple** не создает. Для работы с программами нужна установленная на компьютере система **Maple**.

5.2.1. Оператор цикла

Как и в других алгоритмических языках, в операторе цикла надо указать переменную цикла, начальное ее значение (**from**), шаг (**by**), конечное значение (**to**) и условие выполнения (**while**). Тело цикла, даже если оно состоит из одного оператора, располагается между **do** и **od**. Закрывающая скобка **od** может быть заменена просто на **end** или **end do**. Не все элементы оператора цикла обязательные. По умолчанию под начальным значением понимается 1, в этом случае **from=1** можно не писать. Аналогично шаг цикла равен по умолчанию 1, а условие выполнения **while**, если дано конечное значение для переменной цикла, следует указывать только при необходимости досрочного выхода из цикла. И, наоборот, если дано условие **while**, то конечное значение можно не писать.

Пример:

```
> k:=0;
```

```
> for i from 1 by 2 to 12 while (k<3) do k:=i od;
```

В результате выполнения цикла (он закончится досрочно, как только переменная **k** станет равной 3) на экране появятся две записи $k:=1$ и $k:=3$, отражающие факт присвоения переменной значений 1 и 3. Перед выполнением цикла переменной **k** необходимо было присвоить какое-нибудь значение, иначе при проверке условия $k<3$ система даст сообщение о невозможности проверки условия (*Error, cannot determine if this expression is true or false*). Переменная цикла после его выполнения сохраняет свое последнее значение. В данном случае это 5. Чтобы вывести значение **i** на экран, не требуется обычный оператор типа **print** или **write**. Достаточно записать имя переменной и поставить точку с запятой:

```
> i;
```

Если вместо точки с запятой поставить двоеточие после конца цикла, то действия внутри цикла на экран не выводятся. А если цикл двойной, например,

```
> for i to 5 do for j to 6 do i*j od;od;
```

то независимо от точки с запятой или двоеточия в конце циклов результат промежуточных действий внутри цикла (необходимый, в частности, для отладки) можно будет видеть, если только применить оператор вывода, например **print(i*j)**.

В другом варианте цикла дается список значений, которые должна принять переменная цикла

```
> for i in [2,1,3] do k:=i end;
```

В результате выполнения этого цикла получим

```
k:=2
```

```
k:=1
```

```
k:=3
```

Можно, как и в предыдущем случае, дать условие на досрочный выход из цикла, например `while (k<3)`. Если вместо упорядоченного списка (в квадратных скобках) задать множество (в фигурных скобках), последовательность вывода будет не всегда предсказуемой. В данном примере результаты будут выданы по возрастанию `k`.

Оператор немедленного завершения цикла — `break`.

5.2.2. Условный оператор

Простейший условный оператор состоит всего из трех служебных слов: `if`, `then` и `end`. Условие (логическое выражение) ставится после `if`. В конце условного оператора надо поставить: `end`, `fi` или `end if`. Например: `if k<6 then k6 end`. Чуть более сложная форма условного оператора дает возможность выбора из двух вариантов: `if k<6 then k6 else k0 end`. Если выполняется условие `k<6`, то на экран выводится значение переменной `k6`, иначе появится значение `k0`. Условный оператор может работать как оператор выбора (как переключатель `case` на языке Pascal):

```
> if k=1 then k1
```

```
> elif k=2 then k2
```

```
> elif k=4 then k4
```

```
> elif k=5 then k5 else k0 end if;
```

В зависимости от значения `k` реализуется вывод на экран одного из четырех выражений `k1`, `k2`, `k3` или `k5`. Иначе, если не удовлетворяется ни одно условие — оператор выдает `k0` (см. пример на с. 473).

Еще один условный оператор имеет следующую форму обращения: `'if'(U, T, F)`. Здесь `U` — некоторое условие, `T` — выражение, которое возвращает оператор, если условие истинно, `F` — выражение, которое возвращает оператор, если условие ложно. Такой оператор удобно вставлять непосредственно в формулу как некоторую функцию одной переменной `U` и с двумя значениями `T` и `F`. Например, при `a = 1` следующее выражение

```
> sin('if'(a > 2, Pi, Pi/2));
```

возвратит 1, т. е. $\sin(\pi/2)$, так как выражение `a > 2` ложно.

Определяющее условие записывается с помощью операций отношения `<`, `>`, `<=`, `>=`, `=`, `<>` и может включать в себя логические операции `and`, `or`, `not` и логические константы `true`, `false`.

5.2.3. Процедуры

Описание простейшей процедуры состоит из имени, перечисления аргументов и тела процедуры. В следующей процедуре f с аргументами u и v вычисляется выражение uv^2 :

```
> f:=proc(u,v) u*v^2 end;
```

Обращение к процедуре может быть, например, такое: $f(5, 2)$ или, что то же, $\text{apply}(f, 5, 2)$ ¹. При этом будет возвращено число 20.

Для более надежной работы процедуры желательно обезопасить ее от ложного срабатывания на неверных данных. Если по своему смыслу процедура должна иметь целые аргументы, а ей по ошибке сообщили вещественные или какие-либо другие (что вполне возможно в сложных программах), то желательно предусмотреть некоторую защиту. Этой защитой может быть описание типов ее аргументов. Число типов в Maple огромно, более 200. В том числе есть тип `float` — вещественные числа. Следующая процедура может иметь только вещественные аргументы, а в случае ошибочного типа аргумента она возвращает значения, назначенные по умолчанию:

```
> f:=proc(u::float:=3,v::float:=4) u+v end;
```

Если вызвать процедуру с вещественными аргументами $f(1., 4.)$, то будет получена искомая сумма 5,0, если же задать целые аргументы $f(3, 4)$, то получится ответ 7. Аргумент некорректного типа заменяется на значение по умолчанию. Если значение по умолчанию не задавать, то процедура вообще не работает, вычисления остановятся и появится предупреждение о неправильном вводе: *Error, invalid input: f expects its 1st argument, x, to be of type float, but received 3.*

Более того, можно вызвать процедуру с меньшим числом аргументов, недостающие она заменит на значения по умолчанию. Например, обращение $f(10.)$ дает значение 14,0.

Другая защита процедуры ограничивает число аргументов при вызове процедуры. Как ни странно, но к процедуре можно безнаказанно обратиться с большим числом аргументов, лишние аргументы она просто не воспринимает. Например, для предыдущей процедуры допустимо обращение $f(2., 3., 1)$. При этом опять будет получено число 5. Чтобы избежать этого, достаточно поставить знак $\$$ в конце списка аргументов: $f:=\text{proc}(u,v,\$) u+v \text{ end}$. В случае обращения $f(2, 3, 1)$ будет дано сообщение об ошибке: *Error, invalid input: too*

¹Команда `apply` особенно удобна, когда с помощью оператора `map` требуется к одной величине применить целый список функций и получить список результатов. Например:

```
> map(apply, [cos, tan+1, sqrt, cot], x);
```

```
[cos(x), tan(x) + 1, sqrt(x), cot(x)]
```

Здесь `sqrt` — корень квадратный, `tan` — тангенс, `cot` — котангенс.

many and/or wrong type of arguments passed to F; first unused argument is 1.

Для добавления в текст процедуры краткого ее описания (комментария) служит опция `description`. Пример:

```
> f:=proc(u,v) description "Сумма": u+v end;
```

Прочитать описание процедуры, количество и тип ее аргументов можно с помощью команды `Describe(f)` или, более подробно, `eval(f)`.

Если в теле процедуры требуется использовать какие-либо вспомогательные, временные переменные, то их лучше описать как `local`, иначе все неописанные переменные, введенные в теле процедуры, **Maple** описывает как локальные, загромождая экран многочисленными предупреждениями. Примеры см. на с. 356, 480. Те параметры или переменные, которые заданы вне процедуры, и их значения требуются для счета в теле процедуры, следует описать как глобальные `global`. Примеры даны на с. 440, 442, 480.

Часто требуется получить результат как в символьном виде, так и в численном. Это можно осуществить, указав на аргументы процедуры вида `uneval`. Пример:

```
> f:=proc(x:uneval,y:uneval) x*(y-x) end:
```

```
> A,B:=3,4:G:=f(A,B);eval(G);
```

$$G := A(B - A)$$

3

Параметрам `A` и `B` значения 3 и 4 присваиваются списком (иначе, но чуть длиннее, можно так: `A:=3: B:=4`). Сначала функция вычисляется в символьном виде, а потом, по команде `eval` происходит подстановка численных значений и вычисление. Вместо `eval(G)` можно поставить `eval(%)`, где процент означает результат предыдущей операции, т. е. `G`. Обращение к предыдущим результатам (дitto операция) имеет глубину до трех предшествующих результатов: `%%` и `%%%`.

Часто возникает необходимость в процедуре, которая бы работала с разным число аргументов, например, вычисляла сумму кубов элементов последовательности разной длины. Для этого удобно в качестве аргумента использовать список. Пример:

```
> S:=proc(x) add(x[j]^3,j=1..nops(x)) end:
```

```
> S([1,2,3]),S([1+2,cos(z)]);
```

$$36, 27 + \cos(z)^3$$

В первом случае ищем сумму из трех слагаемых, во втором — из двух. Для того чтобы оба результата были выведены в строку, операторы `S([1,2,3])` и `S([1+2,cos(z)])` отделены запятой. Для вывода в столбец между ними надо ставить точку с запятой. Отметим также,

что индекс суммирования j не потребовалось объявлять локальным, он локален в операторе суммирования `add`.

5.2.4. Функции

В Maple функция — упрощенный тип процедуры. Как и в процедуре, указываются название, аргументы и тело функции. Аргументы указываются в скобках. Если аргумент один — скобки можно не писать. Функция может быть векторно-значной. В этом случае результаты перечисляются через запятую и заключаются в скобки. Пример функции, вычисляющей мощность объединения и пересечения множеств:

```
> A:={a,b,c}:
> B:={a,b,d}:
> f:=(x,y) -> (nops(x union y),nops(x intersect y));
> f(A,B);
      f := (x, y) -> (nops(x ∪ y), nops(x ∩ y))
                    4, 2
```

Получен результат: мощность объединения множеств равна 4, пересечения — 2. Если уже имеется некоторое выражение, то превратить его в функцию можно с помощью команды `unapply`, указав, какие параметры будут аргументами. Пример:

```
> W:=2*t^2+u:
> G:=unapply(W,t):
> G(1);
```

Результатом будет выражение $2+u$. Для того чтобы полученная функция работала только с числами, следует записать тип аргумента `unapply(W,t:numeric)`.

5.3. Преобразование и упрощение выражений

Основное достоинство Maple — работа с формулами и символьными выражениями — может превратиться в недостаток. Численные расчеты в привычных системах (Delphi, C) по определению обычно приводят к числу или набору чисел. Здесь же результат может быть представлен формулой на несколько страниц. Пользоваться такой формулой невозможно. Автоматически система не упрощает результат хотя бы потому, что система не знает, что хочет пользователь, какой вид результата ему нужен. Отсюда ясно, что операторы для преобразования и упрощения выражений играют весьма важную роль для пользователя.

5.3.1. Оператор `simplify`

Оператор `simplify` является основным оператором для упрощения выражений. Вот как легко он справляется с такими выражениями¹:

```
> z:=(6*a^2-3*a*b-3*b^2)/(a^2+a*c-a*b-b*c);
> simplify(z);
```

$$\frac{3(2a+b)}{a+c}$$

Другой пример. Дано выражение $2 \sin x \cos x + 1 / (2 \sin x \cos x)$. Упростим его сначала, используя оператор `simplify` без дополнительных опций (обычно так всегда и делается), а затем, с опцией, напоминающей оператору `simplify` тождество $2 \sin x \cos x = \sin 2x$:

```
> Z:=2*sin(x)*cos(x)+1/sin(x)/cos(x)/2;
> simplify(Z);
> simplify(Z, {2*sin(x)*cos(x)=sin(2*x)});
```

$$\frac{\frac{1}{2} \frac{-1 - 4 \cos(x)^2 + 4 \cos(x)^4}{\sin(x) \cos(x)}}{\frac{1 + \sin(2x)^2}{\sin(2x)}}$$

Второй вариант, очевидно, короче, однако можно было предположить другой ответ. Применив к полученному результату `collect(% , sin(2*x))`, получим наиболее очевидное упрощение

$$\sin(2x) + \frac{1}{\sin(2x)}.$$

Опций может быть несколько. Пусть $x = (\sin(t) + \cos(t))^2$.

```
> simplify(x, [2*sin(t)*cos(t)=sin(2*t),
> sin(t)^2+cos(t)^2=1]);
```

Получим $\sin(2t) + 1$.

Работа по упрощению выражений всегда результативней, когда система знает, с какими переменными она работает. Например, совершенно очевидное упрощение выражения $\ln(x^3 a)$ оператор `simplify` не будет делать до того, как тип переменных будет указан. Укажем, что $a > 0$. Сделать это можно так: `assume(a>0)`. Получим некоторое упрощение

```
> simplify(ln(x^3*a));
ln(a~) + ln(x^3)
```

¹Пример из старинного задачника «Руководство алгебры и собрание алгебраических задач для гимназий, реальных училищ и учительских институтов», А. Малининъ и К. Буренин. Москва, 1899.

В результате упрощение произведено, но не до конца. Вдобавок появился значок \sim , предупреждающий, что тип данной переменной назначен. От этого значка, мешающего восприятию, легко избавиться, включив оператор `interface(showassumed=0)`. Теперь все тильды не будут видны. Если выполнить этот оператор со значением `showassumed=2`, то предупреждающие значки также будут опущены, но в конце программы будет приведен список таких переменных. Далее будем опускать значки \sim , считая, что соответствующий оператор `interface` установлен. Добавим теперь `assume(x>0)`. Получим, наконец,

$$3 \ln(x) + \ln(a).$$

Оператор `simplify` имеет уточняющие опции, призванные улучшать его работу на объектах конкретного вида. Опция ставится через запятую после аргумента: `simplify(z, trig)`. Есть также опции `hypergeom`, `ln`, `sqrt`, `polar`, `Ei`, `radical`, `GAMMA`, и др., но, как показывает практика, работа оператора с ними, как правило, не улучшается.

Примеры применения оператора `simplify` см. на с. 370, 373, 478.

5.3.2. Оператор `combine`

Задача оператора — упростить суммы, произведения в выражении, свернув их к одному. В каком-то смысле этот оператор — обратный оператору `expand`, раскрывающему выражения. Вот несколько примеров:

```
> combine((u^v)^2);
```

$$u^{(2v)}$$

```
> combine(exp(a)^2*exp(3*b));
```

$$e^{(2a+3b)}$$

```
> combine(exp(sin(x)*cos(y))*exp(cos(x)*sin(y)));
```

$$e^{\sin(x+y)}$$

```
> combine(cos(y)^2-sin(y)^2);
```

$$\cos(2y)$$

Вот пример преобразования (трудно назвать упрощением), в результате которого, вопреки поставленной цели, оператор `combine`, наоборот, получает сумму из компактного выражения:

```
> combine(4*cos(x)^3);
```

$$\cos(3x) + 3 \cos(x)$$

В некоторых случаях надо описать переменные:

```
> interface(showassumed=0): assume(x>0, a>0);
```

```
> combine(ln(x)+ln(a));
```



$$\ln(xa)$$

Оператор работает и с инертными (невывчисляемыми) формами операторов (`Int`, `Sum`, `Limit`). Сумма интегралов заменяется на интеграл от суммы. Используется свойство линейности операторов. Приводятся подобные члены в подынтегральном выражении:

```
> combine(Int(2*x+a, x)+3*Int(x, x));
```

$$\int 5x + a dx$$

5.3.3. Операторы `factor`, `ifactor`

Оператор `factor` раскладывает выражение на множители. Пример:

```
> factor(x^2-1);
```

$$(x - 1)(x + 1)$$

По умолчанию оператор работает на поле вещественных чисел. Для работы с комплексными числами следует указать опцию `complex`:

```
> Digits:=3: #Ограничиваем точность до 3-х цифр
```

```
> factor(x^3+1, complex);
```

$$(x + 1.) (x - 0.500 + 0.866 I) (x - 0.500 - 0.866 I)$$

Если требуется, можно уточнить допустимые выражения при разложении на множители (по умолчанию все делается в целых числах)

```
> factor(2-a^4, sqrt(2));
```

$$-(a^2 - \sqrt{2})(a^2 + \sqrt{2})$$

Если же записать просто `factor(2-a^4)`, то разложение на множители не будет произведено.

Оператор `ifactor` раскладывает целые и рациональные числа на множители. Примеры:

```
> ifactor(-792);
```

$$-(2)^3 (3)^2 (11)$$

```
> ifactor(240/169);
```

$$\frac{(2)^4 (3) (5)}{(13)^2}$$

5.3.4. Оператор `collect`

Оператор `collect(R, x)` записывает выражение R по степеням x . Пример. Дано выражение

```
> h := a*ln(b)-ln(b)*b-b^2:
```

Выделяем в нем степени $\ln(b)$:

```
> collect(h, ln(b));
```

$$(a - b) \ln(b) - b^2$$

Выделяем степени b :

```
> collect(h,b);
```

$$a \ln(b) - \ln(b) b - b^2$$

Сравним действия операторов `collect` и `sort`:

```
> collect( exp(a)*y+a^2*y^4+a+1/y+y^0.5, y );
```

$$e^a y + a^2 y^4 + a + \frac{1}{y} + y^{0.5}$$

```
> sort(%, y);
```

$$a^2 y^4 + e^a y + y^{0.5} + \frac{1}{y} + a$$

Особенно эффективна опция `factor` разложения на множители коэффициента, выделяемого оператором `collect`. Для сравнения приведем два варианта обращения к оператору `collect`:

```
> collect(a*x+a^2*x+x^2+x^3, x);
```

```
> collect(a*x+a^2*x+x^2+x^3, x, factor);
```

$$x^3 + x^2 + (a + a^2) x$$

$$x^3 + x^2 + a(1 + a) x$$

5.3.5. Оператор `isolate`

Этот оператор изолирует выражение в левой части уравнения. Например, в уравнении $x + x^2 = x^4 - 1/x$ соберем все x в левой части:

```
> isolate(x+x^2=x^4-1/x, x);
```

$$\frac{-x^2 - x^3 + x^5 - 1}{x} = 0$$

Иногда эта операция приводит к решению уравнения:

```
> isolate(x+x^2=x^3-1/x, x);
```

$$x = -1$$

Оператор удобен, когда надо найти или выделить в левой части не какое-нибудь выражение, а например сумму $x + \sin(x)$

```
> eq:=x+sin(x)=2/(x+sin(x))+1;
```

```
> isolate(eq, x+sin(x));
```

$$x + \sin(x) = 2$$

К сожалению, в некоторых простейших случаях этот оператор не срабатывает. Так, он не может выделить сумму $x + a$ из следующего уравнения

```
> isolate(x+a=(x+a+1)*2+1, x+a);
```

5.3.6. Оператор `subs`

Сделать замену в алгебраическом выражении позволяют операторы `subs` и `algsubs`. Различие между ними видно из следующего примера:

```
> X:=sin(2*a-1)-cos(2*a-1+y):
```

```
> subs (2*a-1=A, X) ;
      sin(A) - cos(2a - 1 + y)
> algsubs (2*a-1=A, X) ;
      sin(A) - cos(A + y)
```

Очевидно, оператор `algsubs` выполняет замену лучше. Но в обоих случаях выполнение замены не отражается на значении `X`, в чем можно убедиться, вызвав эту переменную. Для того чтобы `X` изменился, необходимо присвоить результат подстановки этой переменной: `X:=algsubs (2*a-1=A, X)`.

5.3.7. Операторы ввода и вывода

Для записи в файл какой-либо информации сначала надо открыть файл для записи оператором `fopen` с опцией `WRITE`. Если вместо названия файла указать `terminal`, то вывод будет произведен на экран. Оператор `writedata` информацию записывает, а оператор `fclose` закрывает файл. Можно записать матрицу чисел, строки, целые и вещественные числа. В текстовых данных допустима команда перевода строки `\n`, при этом пробел после команды необязателен. В одной переменной могут быть данные разных типов. В следующем примере в переменной `y` содержится строка и два вещественных числа. Для них в операторе `writedata` предусмотрено строковое место, целое и вещественное. Значение 2.9 выводится округленным до целого отбрасыванием дробной части. Отметим обязательные скобки в данных. Если скобки двойные, как в переменной `y`, то вывод производится в строку.

Для вывода массива чисел их надо организовывать в виде `array` или `matrix`. Если они записаны в `Matrix`, то необходима конвертация `M:=convert (M, array)`.

```
> fd := fopen("C:\\test1.txt",WRITE) :
> M:=array([[1,20,3],[2,3,-4.9]]) :
> x:["Вектор"] :
> y:["Ответы\nz=",2.9,3.2]] :
> writedata(fd,M) :
> writedata(fd,x,string) :
> writedata(fd,y,[string,integer,float]) :
> fclose(fd) ;
```

Содержимое записанного файла:

```
1  20  3
2   3 -4.9
Вектор
Ответы
z=  2  3.2
```

Для чтения данных в операторе `fopen` следует указать опцию `READ`, для дозаписи — `APPEND`. Для записи данных можно использовать оператор форматного вывода `fprintf`. Открывает файл для записи тот же оператор `fopen`, а закрывает — `fclose`. Данные выводятся в определенном формате, который указывается в операторе в кавычках перед списком вывода точно так же, как и в языке Си. Обозначения для форматов одинаковы для оператора вывода в файл `fprintf` и для оператора вывода на экран `printf`. Будем демонстрировать примеры для оператора `printf`. За вывод целых чисел отвечают форматы `d`, `o`, `x` (иногда говорят: *спецификаторы* типа `d`, `o`, `x`). Формат `d` применяется для чисел в десятичной форме, `o` — в восьмеричной, `x` — в шестнадцатеричной. Пример:

```
> printf("%d %o %x\n", 11, 12, 18);
```

Получаем ожидаемые результаты: 11 14 12. Очевидно, 14 по основанию 8 есть $12 = 1 \cdot 8^1 + 4 \cdot 8^0$, а 12 по основанию 16 есть $18 = 1 \cdot 16^1 + 2 \cdot 16^0$.

Для вывода вещественных чисел служат форматы `f`, `e`, `g`. Формат `f` отвечает за вывод чисел с плавающей запятой и записывается в виде `%m.nf`, где `m` — число позиций, отводимых под все число, `n` — под дробную часть (по умолчанию `n = 6`). Если число `m` больше, чем длина числа (вместе с десятичной точкой), то впереди числа даются пробелы.

Формат `e` отвечает за вывод чисел с плавающей запятой в экспоненциальной форме и записывается в виде `%m.ne`, где `m` — число позиций, отводимых под все число, `n` — под дробную часть. Формат `g` универсальный. Если число нельзя вывести в обычном формате `f`, то оно выводится в экспоненциальном. Примеры:

```
> A:=10*Pi: B:=1000*Pi:
```

```
> printf("%6.3e %6.3f %6.3g %6.3g", A, A, A, B);
```

```
3.142e+01 31.416 31.4 3.14e+03
```

Для вывода одного символа используется формат `s`, для вывода строки — `s`. Алгебраические выражения выводятся на печать в формате `a`. В следующем примере для перевода на новую строку использован командный символ `\n`.

```
> printf("%6s,%c,%10a", "Резуль\nтат", w, sqrt(R));
```

```
Резуль
```

```
тат,w, R^(1/2)
```

Кроме спецификаторов форматов, в `Maple` предусмотрены управляющие флаги. Флаг `+` ставится после значка `%` и указывает, что надо выводить знак величины (по умолчанию плюс не ставится). Флаг `-` (минус) означает выравнивание записи влево (по умолчанию — вправо). Эти флаги можно менять местами:

```
> printf("%+-15.3e", Pi);
```

```
+3.142e+00
```

Флаг 0 заменяет пробелы слева нулями:

```
> printf("%015.3e", Pi);
```

```
0000003.142e+00
```

Приведем пример чтения данных. Пусть в файле `test2.txt`, лежащем в корне диска C, содержатся три строки

```
6.1 2.3
10 20 30 40
21
```

Для того чтобы прочитать данные, необходимо сначала открыть файл для чтения:

```
> fd := fopen("C:\\test2.txt", READ);
```

прочитать:

```
> A:=readdata(fd, float, 3);
```

и закрыть файл:

```
> fclose(fd);
```

В результате в каждой строке будут прочитаны по три числа (последний параметр оператора `readdata`) и занесены во вложенный список: $A := [[6.1, 2.3], [10., 20., 30.], [21.]]$ К полученному списку можно обращаться обычным образом, например: $A[1, 2]$. При этом получится результат 2.3. Обращение к элементу $A_{1,2}$ допустимо также в форме $A[1][2]$. С таким же успехом можно прочитать и строковые или целочисленные данные. Для этого в операторе `readdata` вместо `float` надо поставить `string` или `integer`.

Чтение файла из маплета и пример использования оператора `readbytes` см. в примере 24, с. 334.

5.4. Алгебраические операции

5.4.1. Простые операции

Некоторые простые алгебраические действия можно выполнить в **Maple**, пользуясь встроенными операторами, более сложные требуют подключения специального пакета **LinearAlgebra**, описание которого дано в этом разделе. Но сначала рассмотрим простые действия, например решение алгебраического уравнения. Зададим квадратное уравнение

```
>Уравнение:=x^2-3*x+2;
```

Решим его:

```
> solve(Уравнение);
```

Получим два решения: 1 и 2. Отметим, что в уравнении было совершенно ясно, что искать, поэтому в операторе `solve` не указана неизвестная. Чаще это приходится делать. Пусть дано уравнение с параметром a :

```
> Уравнение2 := x^2 - 4*x + a;
```

Решим его, указав неизвестную

```
> solve(Уравнение2, x);
```

Получим два решения: $2 + \sqrt{4 - a}$ и $2 - \sqrt{4 - a}$. Решение получено, но это не значит, что x приобретет одно из этих значений. Если записать

```
> R := solve(Уравнение2, x); ,
```

где R — некоторая вспомогательная переменная, в которой хранятся решения, то R приобретет найденные значения, причем сразу оба, т. е. x станет списком из двух значений. Обратиться к одному из них можно, указав соответствующий номер: $x := R[1]$ или $x := R[2]$.

Системы уравнений записываются в виде множества (*set*) в фигурных скобках:

```
> Система := {u - v = 4, u + v = 2*b};
```

или в виде списка (*list*) в квадратных скобках: $[u - v = 4, u + v = 2*b]$.

Решаем систему, указав в фигурных скобках неизвестные (здесь из-за параметра b это необходимо):

```
> S := solve(Система, {u, v});
```

Назначаем неизвестным их найденные значения

```
> assign(S);
```

Обе неизвестные приобретают свои значения, переменная u становится равной $b + 2$, а переменная v — равной $b - 2$.

Последнюю систему легко решить и в матричном виде. Здесь также не требуется подключать какие-либо дополнительные математические пакеты. Запишем по строкам матрицу линейной системы уравнений

```
> A := Matrix([[1, -1], [1, 1]]);
```

и вектор правой части

```
> B := Vector([4, 2*b]);
```

Обратная матрица находится удивительно просто: $1/A$. Решим систему методом обратной матрицы

```
> Z := 1/A . B;
```

где Z — некоторая вспомогательная переменная. Назначим неизвестным их значения:

```
> u := Z[1]; v := Z[2];
```

Ответ, конечно, будет тот же: $u = b + 2$, $v = b - 2$. Матрицу можно ввести и по столбцам: $A := \langle\langle 1, 1 \rangle | \langle -1, 1 \rangle \rangle$.

Заметим, что в **Maple** есть похожая на **Matrix** структура: **matrix**. Она также и по таким же правилам формирует матрицу, однако не допускает нахождение обратной матрицы и умножение на вектор с помощью точки. А вектор **vector** к тому же формирует вектор-строку, а не вектор-столбец, как **Vector**.

При работе в **Maple Standard Worksheet** матрицу можно ввести в диалоговом режиме из меню по цепочке **Tools->Tutors->Linear Algebra->Matrix Builder**. Для **Maple Classic Worksheet** диалоговый режим запускается по команде

```
> Student[LinearAlgebra][MatrixBuilder]();
```

В **Maple** есть еще несколько операторов решения уравнений. В операторе модульной арифметики **msolve** используется понятие равенства по модулю натурального числа. Два целых числа сравнимы по модулю натурального числа, если при делении на это число они дают одинаковые остатки. Например, оператор **msolve(2*x=3, 5)** дает решение $x = 4$. Действительно, остаток от деления 8 на 5 равен 3. Решение рекуррентных уравнений дает оператор **rsolve** (см. с. 123). Оператор **isolve** решает уравнения в целых числах. Например, в результате решения уравнения **isolve(3*x+2*y=17, z)** получаем решение $\{y = 7 - 3z, x = 1 + 2z\}$. Здесь указан необязательный параметр z , принимающий целые значения, через который выражается решение. Если его не задать, то эта переменная будет обозначена как **_Z**. Оператор **fsolve** предназначен для численного решения уравнений или систем уравнений. Решение можно искать и в комплексной области. Для этого надо указать искомую переменную и обозначить опцию. Например, оператор **fsolve(x^2=-2, x, complex)** дает два решения $-1.414213562 I, 1.414213562 I$. Мнимая единица в **Maple** является константой и обозначена как **I**.

5.4.2. Пакет LinearAlgebra

Здесь перечислены и кратко описаны основные операторы пакета **LinearAlgebra**. Всего в **Maple 13** имеется 121 оператор¹. Пакет следует вызвать в начале программы командой **with(LinearAlgebra)**. В старых версиях (до 6-й) вместо **LinearAlgebra** имелся аналогичный пакет **linalg**. В новых версиях он сохранен для совместимости, почти все основные операторы работают очень похоже, но новый (с 2000 г.) пакет **LinearAlgebra** существенно сильнее, поэтому старый пакет описывать нет необходимости, хотя иногда он будет использован (с. 358, 360).

Использование команды **with** самый простой, но единственный способ подключить пакет. Пакет можно подключить в так называ-

¹В версии **Maple 7** за 2001 г. было 103 оператора. В **Maple 15** добавлен еще оператор **CARE** для решения матричного алгебраического уравнения Риккати.

емую *длинную форму* вызова. Например, определитель матрицы A можно вычислить так: `LinearAlgebra[Determinant](A)` или так: `LinearAlgebra:-Determinant(A)`. Это относится и к другим операторам и пакетам.

- `Add(A, B, c1, c2)`. Вычисление суммы $c1 \cdot A + c2 \cdot B$, где A, B — матрицы или векторы, $c1, c2$ — скаляры.
- `Adjoint(A)`. Присоединенная¹ матрица — матрица, составленная из алгебраических дополнений для соответствующих элементов матрицы A и затем транспонированная. Используется при вычислении обратной матрицы. Например, дана матрица $A := \text{Matrix}([[1, -2], [3, 4]])$. Оператор `Adjoint(A)` дает следующий результат:

$$\begin{bmatrix} 4 & 2 \\ -3 & 1 \end{bmatrix}.$$

- `BackwardSubstitute(A)`. Процедура обратной подстановки в методе Гаусса² решения системы линейных уравнений или решение системы, представленной в виде расширенной верхней треугольной матрицы A , т. е. матрицы с правыми частями системы. Если

$$A := \text{Matrix}([[1, 3, 2, 1], [0, 1, -2, 1], [0, 0, 1, -2]]), \quad (5.1)$$

то `BackwardSubstitute(A)` есть вектор

$$\begin{bmatrix} 14 \\ -3 \\ -2 \end{bmatrix}, \text{ где } A = \begin{bmatrix} 1 & 3 & 2 & 1 \\ 0 & 1 & -2 & 1 \\ 0 & 0 & 1 & -2 \end{bmatrix}. \quad (5.2)$$

- `BandMatrix(S, n, i, j)`. Ленточная матрица, где S — список списков диагональных элементов, n — число ненулевых диагоналей под главной диагональю, i — число строк, j — число столбцов. Списки могут быть разной длины, недостающие элементы принимаются нулевыми, но если диагональ не помещается в заданные размеры матрицы, то выдается ошибка. Например, $S := [[x1, x2, x3], [y1, y2, y3], [z1]]$. Результат операторов `BandMatrix(S, 1, 4, 4)` и `BandMatrix(S, 0, 4, 4)` будет соответственно следующим:

$$\begin{bmatrix} y1 & z1 & 0 & 0 \\ x1 & y2 & 0 & 0 \\ 0 & x2 & y3 & 0 \\ 0 & 0 & x3 & 0 \end{bmatrix}, \quad \begin{bmatrix} x1 & y1 & z1 & 0 \\ 0 & x2 & y2 & 0 \\ 0 & 0 & x3 & y3 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

¹Союзная, взаимная.

²*Carl Friedrich Gauss* (1777–1855) — немецкий математик.



Но если дать число поддиагоналей больше, например `BandMatrix(S, 2, 4, 4)`, то будет сообщение об ошибке: *Error, (in Matrix) initializer defines more rows (5) than row dimension parameter specifies (4)*. Аналогично при $n = 0$ нельзя давать слишком длинную диагональ `[z1, z2, z3]` — не поместится справа `z3`, или при $n = 1$ ввести `[x1, x2, x3, x4]` — не поместится снизу `x4`. Если вообще не указывать число поддиагоналей и размеры, то матрица всегда образуется, причем лишние нулевые столбцы справа будут удалены.

- `BilinearForm(V1, V2, A, conjugate=false)`. Билинейная форма на векторах $\bar{V}^{(1)}$, $\bar{V}^{(2)}$ и матрице A : $\sum_{i,j} V_j^{(1)} V_i^{(2)} A_{ij}$. По умолчанию матрица A — единичная, в этом случае оператор дает скалярное произведение векторов. Параметр `conjugate` (сопряженность) по умолчанию принят `true`, и под элементами векторов понимаются комплексные числа и результат имеет вид $\sum_{i,j} V_j^{(1)} \bar{V}_i^{(2)} A_{ij}$.
- `CharacteristicMatrix(A, lambda)`. Характеристическая матрица. Вычисляется по формуле $\lambda E - A$, где E — единичная матрица.
- `CharacteristicPolynomial(M, x)`. Характеристический полином матрицы A по переменной x .
- `Column(A, k)`. Столбец k или список столбцов матрицы A . Если $k < 0$, то номер берется с конца, например $k = -1$ соответствует последнему столбцу. Можно указать список, например `Column(A, [-2, 1..3])` — предпоследний столбец и первые три.
- `ColumnDimension(A)`. Число столбцов матрицы A .
- `ColumnOperation(A, [i1, i2], k)`. Линейная операция со столбцами матрицы: на место столбца `i1` помещается линейная комбинация столбцов `Column(A, i1) + k*Column(A, i2)`. Например, операция `ColumnOperation(A, [1, 2], 10)` с матрицей (5.2) дает следующую матрицу:

$$\begin{bmatrix} 31 & 3 & 2 & 1 \\ 10 & 1 & -2 & 1 \\ 0 & 0 & 1 & -2 \end{bmatrix}. \quad (5.3)$$

Матрицу (5.1) можно задать и по столбцам: $A := \langle \langle 1, 0, 0 \rangle \mid \langle 3, 1, 0 \rangle \mid \langle 2, -2, 1 \rangle \mid \langle 1, 1, -2 \rangle \rangle$.

- `ColumnSpace(A)`. Базисные вектора столбцов матрицы.

- `CompanionMatrix(p, x)`. Сопровождающая матрица для полинома p по переменной x . Рассмотрим случай нормированного полинома (коэффициент при старшей степени равен единице). Для полинома $p := x^3 + 2x^2 - 5x - 6$ получим матрицу

$$\begin{bmatrix} 0 & 0 & 6 \\ 1 & 0 & 5 \\ 0 & 1 & -2 \end{bmatrix}.$$

Порядок матрицы для полинома степени n равен n . Под главной диагональю стоят единицы, последний столбец — коэффициенты полинома с обратными знаками, от свободного члена (вверху) до $(n-1)$ -го внизу, кроме 1 при старшем члене. Собственные числа сопровождающей матрицы совпадают с корнями полинома. В данном примере это числа $-1, 2, -3$.

- `ConditionNumber(A)`. Число обусловленности матрицы A . Здесь под этим числом понимается произведение нормы матрицы на норму обратной матрицы, где под нормой матрицы принята `MatrixNorm(A, infinity)`.
- `ConstantMatrix(x, i, j)`. Матрица из i строк и j столбцов, заполненная скалярными значениями x . Если указан один размер `ConstantMatrix(x, i)`, то матрица квадратная. Можно уточнить форму матрицы — например, оператор `ConstantMatrix(z, 2, outputoptions = [shape=triangular[lower]])` дает следующую нижнюю треугольную матрицу

$$\begin{bmatrix} z & 0 \\ z & z \end{bmatrix}.$$

Форму матрицы определяет параметр `shape`. Для верхней треугольной матрицы надо задать параметр `triangular[upper]`. Параметр `shape` может принимать также следующие значения: `antihermitian, antisymmetric, band, constant, diagonal, identity, hermitian, scalar, symmetric, triangular, zero, Hessenberg`. Использование параметра формы облегчает ввод данных. Например, задав опцию `shape=symmetric`, можно уже после задания матрицы ввести `A[1,2]:=x` и автоматически получить при этом $A_{2,1} = x$.

- `ConstantVector`. Вектор с заданными постоянными значениями компонент. Имеется два варианта: вектор столбец `[column]` (по умолчанию) и вектор строка `[row]`. Задаем, например, `Z:=ConstantVector[row](t,3)`. Получаем $Z := ((t, t, t))$. Но если потом попытаться изменить элемент вектора `Z[1]:=1`, то система выдаст сообщение об ошибке: *Error, attempt to assign new value to constant rtable.*

- `Copy`. Копирование матрицы или вектора, разрывающее связь между объектами. Очень удобная операция для сохранения, например, исходных данных. Рассмотрим пример. Пусть задан вектор $V := \text{Vector}([a, b, c])$. Если создать две копии: одну — обычным способом $U:=V$, а другую — этим оператором $W := \text{Copy}(V)$, то, изменив в исходных данных некоторое значение $V[1]:=0$, мы меняем и $U[1]$, но не $W[1]$. Аналогичный эффект можно получить оператором цикла `for i to 3 do x[i]:=V[i]:od`.
- `CrossProduct(V1, V2)`. Векторное произведение векторов $V1$ и $V2$ (дословно, "произведение крестом"). Можно добавить опции, например, вывести результат в вещественной форме: `CrossProduct(V1, V2, outputoptions=[datatype=float])`. Допустима также инфиксная форма $V1 \&x V2$, но в ней нельзя добавлять опции.
- `DeleteColumn`. Стереть столбец матрицы. Указывается номер столбца или список. Например, команда `DeleteColumn(A, [1, 3, 5])` по матрице A создает новую матрицу, которая отличается от исходной тем, что в ней на три столбца меньше (нет столбцов 1, 3 и 5). Список столбцов можно задавать, как и все списки в **Maple**. Список $[2..5]$ соответствует столбцам 2, 3, 4, 5. Если номера отрицательные, то отсчет идет с последнего элемента. Так, для того чтобы стереть два последних столбца, надо задать список $[-2, -1]$ или $[-1, -2]$.
- `DeleteRow`. Стереть строку матрицы (аналогично `DeleteColumn`).
- `Determinant(A)`. Определитель матрицы A^1 .
- `Diagonal(A, k)`. Диагональ матрицы номер k . По умолчанию $k=0$ — возвращается главная диагональ матрицы (не обязательно квадратной, начинается диагональ с элемента $A_{1,1}$). Поддиагонали соответствуют $k < 0$, наддиагонали $k > 0$. Можно получить сразу несколько диагоналей, если вместо k указать список, например $[1, -1]$. Для квадратной матрицы поддиагонали и наддиагонали короче главной диагонали.

¹В пакете `linalg` определитель записывается короче — `det` и работает он более неприхотливо. Если для оператора `Determinant` требуется аргумент типа `Matrix`, то оператор `det` работает и с массивами `array`, и с матрицами `matrix` или `Matrix`. Определитель матрицы, заданной непосредственным присваиванием значений ее элементов $a[1,1]:=1: a[2,2]:=1: a[1,2]:=3$, можно вычислить оператором `det` после одной конвертации $A:=\text{convert}(a, \text{array})$. Так как один элемент не был задан, то при этом получится $1 - 3A_{2,1}$. Для оператора `Determinant` потребуется вторая конвертация $A1:=\text{convert}(A, \text{Matrix})$, и получится значение 1.

- **DiagonalMatrix(L, n, m)**. Диагональная матрица из n строк, m столбцов с диагональю из списка L . Если не указывать размер матрицы, то будет создана квадратная матрица, соответствующая длине списка. Пример: оператор `DiagonalMatrix([1,2],3,4)` создает матрицу

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \quad (5.4)$$

- **Dimension(M)**. Размер матрицы или вектора. Для матрицы (5.4) ответ будет 3,4. Этот же оператор имеет и другое имя: `Dimensions(M)`.
- **DotProduct(V1, V2)**. Скалярное произведение векторов (дословно "произведение точкой"). По умолчанию предполагается поле комплексных чисел, поэтому, если даны векторы $V1:=\text{Vector}([x1,x2])$ и $V2:=\text{Vector}([y1,y2])$, то оператор `DotProduct(V1, V2)` дает следующий результат: $\overline{x1} y1 + \overline{x2} y2$, где черта сверху означает комплексное сопряжение. Оператор с указанной опцией `DotProduct(V1, V2, conjugate=false)` возвращает $y1 x1 + y2 x2$.
- **Eigenvalues(A)**. Собственные числа матрицы. Результат дается в виде вектора-столбца. Можно указать форму вывода, например списком: `Eigenvalues(A, output=list)`, или, если значения комплексные, дать уравнения, из которых эти числа определяются: `Eigenvalues(A, implicit)`.
- **Eigenvectors(A)**. Собственные векторы матрицы. По умолчанию оператор выдает еще и собственные числа в виде столбца. Если требуются только векторы, то следует задать эту опцию: `Eigenvectors(A, output=['vectors'])`. Можно поменять порядок вывода: `output=['vectors', 'values']`.
- **FrobeniusForm(A)**. Матрица Фробениуса, или первая естественная нормальная форма матрицы. По квадратной матрице A можно получить матрицу Фробениуса F и матрицу преобразований Q , обладающие свойством: $Q^{-1}AQ = F$, или, что то же $QFQ^{-1} = A$. Матрица Фробениуса имеет в общем случае квазидиагональный вид. Матрицы на диагонали имеют форму сопровождающей матрицы (`CompanionMatrix`), причем в ряду характеристических многочленов диагональных клеток каждый многочлен, начиная со второго, является делителем предыдущего. Вызов оператора, дающий обе матрицы, имеет вид $(F, Q) := \text{FrobeniusForm}(A, \text{output}=['F', 'Q'])$. Отдельно матрица Фробениуса получается

оператором `FrobeniusForm(A)`. Другое имя этого же оператора — `RationalCanonicalForm`.

- `GenerateEquations(A, X, R)`. Оператор, создающий систему уравнений по ее матрице A и по вектору правой части R . Обозначения для неизвестных содержатся в списке X . Например, оператор `GenerateEquations(A, [x, y, z, t], <1, 2, -1>)` для матрицы (5.3) дает следующие уравнения:

$$[31x+3y+2z+t = 1, 10x+y-2z+t = 2, z-2t = -1]$$

- `GenerateMatrix(S, V)`. Матрица системы линейных уравнений S . В списке V указываются обозначения неизвестных. Например, дана система уравнений $S:=[x+y=3, x-y=1]$ с неизвестными $V:=[x, y]$. Оператор `GenerateMatrix(S, V, augmented=true)` создает матрицу¹

$$\begin{bmatrix} 1 & 1 & 3 \\ 1 & -1 & 1 \end{bmatrix}. \quad (5.5)$$

По умолчанию `augmented=false`, и вектор правых частей задается отдельным вектором.

- `GivensRotationMatrix(V, i, j)`. Матрица поворота, составленная по списку V по правилу $g_{i,i} = g_{j,j} = v_i/\sqrt{v_i^2 + v_j^2}$, $g_{i,j} = -g_{j,i} = v_j/\sqrt{v_i^2 + v_j^2}$, $g_{m,m} = 1$ при $m \neq i$ и $m \neq j$, остальные элементы равны нулю.
- `GramSchmidt(V)`. Процедура ортогонализации Грама–Шмидта² списка векторов V . Например, оператор `GramSchmidt([c1, c2, c3])` по векторам $\vec{c}_1, \vec{c}_2, \vec{c}_3$ строит систему ортогональных векторов по правилу³:

$$\vec{b}_1 = \vec{c}_1, \alpha_{2,1} = -\vec{c}_2 \cdot \vec{b}_1 / \vec{b}_1^2, \vec{b}_2 = \vec{c}_2 + \alpha_{2,1} \vec{b}_1, \alpha_{3,1} = -\vec{c}_3 \cdot \vec{b}_1 / \vec{b}_1^2, \alpha_{3,2} = -\vec{c}_3 \cdot \vec{b}_2 / \vec{b}_2^2, \vec{b}_3 = \vec{c}_3 + \alpha_{3,1} \vec{b}_1 + \alpha_{3,2} \vec{b}_2.$$

Если дополнительно указать опцию `normalized`, то вектора будут нормализованы.

- `HankelMatrix(V, n)`. Матрица Ганкеля⁴ — квадратная матрица порядка n , образуется из элементов списка V (их должно быть

¹Расширенная матрица

²*Gram J.* (1850–1916), *Erhard Schmidt* (1876–1959) — немецкие математики.

³*Зимина О. В.* Линейная алгебра и аналитическая геометрия / Под ред. А. И. Кириллова. — М.: Издательство МЭИ, 2000.

⁴*Hermann Hankel* (1839–1873) — немецкий математик.



нечетное число), расставленных по диагоналям, параллельным побочной. Например, по списку $V := \langle a, b, c, d, e, g, f \rangle$ оператор `HankelMatrix(V, 3)` дает следующую матрицу:

$$\begin{bmatrix} a & b & c \\ b & c & d \\ c & d & e \end{bmatrix}.$$

Размер матрицы ограничен параметром 3, и поэтому два последних элемента списка V не поместились в матрицу, но если не указывать размер, то порядок матрицы будет соответствовать длине списка, в данном случае 4. Элементы матрицы Ганкеля получаются по формуле $h_{i,j} = V_{i+j-1}$. Для сравнения: матрица Тейлора (с. 260) образуется из элементов списка, расставленных по диагоналям, параллельным главной.

- `HessenbergForm(A)`. Приведение квадратной матрицы A к верхней форме Хессенберга¹. Имеет место разложение $A = QHQ^T$, где Q — ортогональная матрица (обратная матрица равна транспонированной $Q^{-1} = Q^T$). Результатом действия оператора `HessenbergForm(A)` является матрица H . Если необходимо получить и матрицу Q , следует использовать полную форму:

$(H, Q) := \text{HessenbergForm}(A, \text{output}=['H', 'Q'])$.

В этой форме обозначения для матриц закреплены только за буквами H и Q . Решение легко проверить, выполнив умножение (обозначается точкой): $Q.H.\text{Transpose}(Q)$. Должна получиться исходная матрица. Пусть, например, имеем матрицу

$$A := \begin{bmatrix} 1 & 2 & 2 \\ 2 & 3 & 3 \\ 3 & 4 & 4 \end{bmatrix}.$$

Получим

$$H, Q := \begin{bmatrix} 1.00 & -2.77 & -0.55 \\ -3.61 & 6.92 & 1.38 \\ 0.00 & 0.38 & 0.08 \end{bmatrix}, \begin{bmatrix} 1.00 & 0.00 & 0.00 \\ 0.00 & -0.55 & -0.83 \\ 0.00 & -0.83 & 0.55 \end{bmatrix}.$$

Матрица Хессенберга является почти треугольной — под главной диагональю есть только одна диагональ, отличная от нулевой. Заметим, что по умолчанию операторы численного счета пакета `LinearAlgebra` выдают излишне большую точность — 18 значащих цифр. Команда `Digits`, меняющая точность вычислений в `Maple`, здесь не может исправить ситуацию. На помощь приходит команда `interface(displayprecision=2)`.

¹*Karl Hessenberg* (1904–1959) — немецкий математик.

- **HilbertMatrix(m, n, x)**. Матрица Гильберта¹ размером $m \times n$ с параметром x . Элементы матрицы вычисляются по формуле $h_{ij} = 1/(i + j - x)$.
- **IdentityMatrix(m)**. Квадратная единичная матрица порядка m . Если задать **IdentityMatrix(n, m)**, то можно получить и прямоугольную единичную матрицу, в которой $a_{i,i} = 1, i = 1..k, k = \max(m, n)$.
- **IsDefinite(A)**. Проверка положительной определенности матрицы. Результат действия оператора *false* или *true*.
- **IsOrthogonal(A)**. Проверка ортогональности матрицы. Результат действия оператора *false* или *true*.
- **IsSimilar(A, B)**. Проверка подобия матриц. Результат действия оператора *false* или *true*. Для подобных матриц A и B существует обратимая матрица C , для которой справедливо соотношение $CA = BC$ или $A = C^{-1}BC$.
- **JordanBlockMatrix(A)**. Жорданова матрица — квадратная блочно-диагональная матрица, определяемая списком жордановых клеток A . Оператор **JordanBlockMatrix([[a, 2], [b, 3]])** дает следующую матрицу:

$$\begin{bmatrix} a & 1 & 0 & 0 & 0 \\ 0 & a & 0 & 0 & 0 \\ 0 & 0 & b & 1 & 0 \\ 0 & 0 & 0 & b & 1 \\ 0 & 0 & 0 & 0 & b \end{bmatrix}. \quad (5.6)$$

Каждая клетка $[a, n]$ представляет собой треугольную матрицу порядка n , собственные значения которой равны a .

- **JordanForm(A)**. Жорданова форма J матрицы A . Справедливо разложение $J = Q^{-1}AQ$. Например, для матрицы

$$A = \begin{bmatrix} 1 & 6 & -4 \\ 1 & 2 & 0 \\ 1 & 3 & 0 \end{bmatrix} \quad (5.7)$$

получим

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 2 & 1 \\ 0 & 0 & 2 \end{bmatrix}.$$

¹David Hilbert (1862–1943) — немецкий математик, автор знаменитых "двадцати трех проблем" математики.



Форма обращения к оператору, возвращающая также матрицу Q , имеет вид $Q, J := \text{JordanForm}(A, \text{output}=['Q', 'J'])$.

- **KroneckerProduct(A, B)**. Произведение Кронекера¹ матриц A и B . Результатом произведения матрицы размером $m \times n$ и матрицы размером $p \times q$ является матрица $mp \times nq$. Пусть даны матрицы

$$A := \begin{bmatrix} 1 & 2 \\ 3 & -1 \end{bmatrix}, \quad B := \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix}.$$

Оператор **KroneckerProduct(A, B)** дает следующую матрицу:

$$\begin{bmatrix} a & b & c & 2a & 2b & 2c \\ d & e & f & 2d & 2e & 2f \\ 3a & 3b & 3c & -a & -b & -c \\ 3d & 3e & 3f & -d & -e & -f \end{bmatrix}.$$

- **LUDecomposition(A)**. Разложение матрицы A в виде произведения трех матриц $A = PLU$, где L — нижняя треугольная матрица, U — верхняя треугольная матрица, P — матрица перестановок². Пример. Для матрицы (5.7) получим следующее разложение:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & \frac{3}{4} & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 6 & -4 \\ 0 & -4 & 4 \\ 0 & 0 & 1 \end{bmatrix}.$$

Если матрица положительно определенная³, то можно получить разложение Холецкого⁴ $A = LL^T$, где L — нижняя треугольная матрица. По умолчанию используется `method='GaussianElimination'` — метод исключения Гаусса.

- **LeastSquares(A, b)**. Решение переопределенной системы уравнений в среднеквадратичном смысле, т. е. нахождение решения системы $A\vec{X} = \vec{b}$ из условия минимума евклидовой⁵ нормы вектора $\|A\vec{X} - \vec{b}\|$.
- **LyapunovSolve(A, Q)**. Решение матричного уравнения Ляпунова⁶ $AX + XA^T = Q$.

¹Leopold Kroneker (1823–1891) — немецкий математик.

²В матрице перестановок в каждой строке и столбце по одной единице. Если для разложения не требуется переставлять столбцы, то матрица единичная.

³Проверяет положительную определенность оператор `IsDefinite(A)`.

⁴Andre-Louis Cholesky (1875–1918) — французский математик.

⁵Евклидова норма вектора $\|\vec{V}\| = \sqrt{\sum_i V_i^2}$ или `Norm(V, 2)`.

⁶Ляпунов Александр Михайлович (1857–1918) — русский математик.

- **LinearSolve(A)**. Решение системы линейных уравнений с расширенной матрицей A . Например, для матрицы (5.5) получим решение (вектор столбец) с компонентами 2 и 1. Можно решить недоопределенную систему (число неизвестных больше числа уравнений), указав метод решения. Пусть $A := \langle \langle 1, 0 \rangle | \langle 2, 1 \rangle | \langle -1, -1 \rangle \rangle$. Укажем явным образом вектор правых частей $b := \langle 0, 1 \rangle$. Оператор **LinearSolve(A, b, method='subs', free='z')** дает следующее решение:

$$A := \begin{bmatrix} 1 & 2 & -1 \\ 0 & 1 & -1 \end{bmatrix}, \quad b := \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad \begin{bmatrix} -2 - z_1 \\ 1 + z_1 \\ z_1 \end{bmatrix}. \quad (5.8)$$

Указана свободная неизвестная z (как параметр решения). Для основных неизвестных матрица в этом методе должна быть верхней треугольной, иначе появляется сообщение об ошибке: *Error, (in LinearAlgebra:-LA Main:-LinearSolve) Matrix shape incompatible with method = subs.* Другие методы решения: none, solve, Cholesky, LU, QR, hybrid, modular, SparseLU, SparseDirect, SparseIterative.

Заметим, что при работе в **Maple Standard Worksheet** можно показать процесс решения системы уравнений, например, методом исключения Гаусса. Для этого есть режим обучения, где на экране отображается каждый шаг алгоритма (рис. 331, 332). Зайти в программу можно из меню: Tools->Tutors->Linear Algebra->Gaussian Elimination. Для **Maple Classic Worksheet**, где такого меню нет, эта обучающая программа запускается по команде `Student[LinearAlgebra][GaussianEliminationTutor]()`.

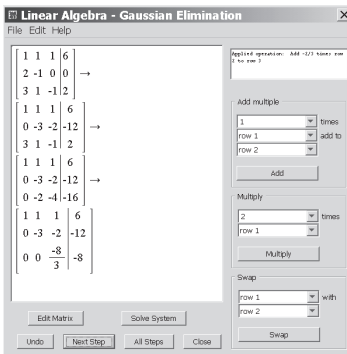


Рис. 331

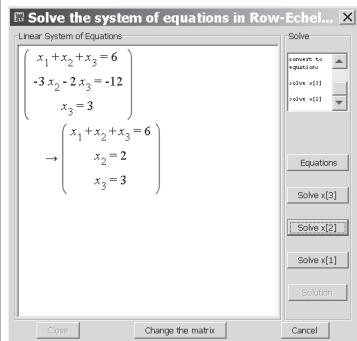


Рис. 332

- **Map**. Оператор, применяющий некоторое действие к элементам матрицы или вектора. Например, для того чтобы вычесть 2 из всех элементов матрицы A , достаточно записать $\text{Map}(x \rightarrow x-2, A)$. Первый аргумент оператора **Map** является процедурой, второй — матрицей. Очевидно, это же можно выполнить и в двойном цикле. По умолчанию действие производится со всеми элементами матрицы, но можно указать и область определения. Пусть требуется умножить на 10 все элементы на диагонали и ниже. В квадратных скобках указываем область:

```
> Map[(i,j) -> evalb(i>=j)](x->x*10, A);
```

Оператор **evalb** конвертирует условие в логическое выражение ("evaluate boolean" — выразить в логической форме).

Характерное отличие оператора **Map** от многих других операторов **Maple** — результат действия не надо присваивать некоторой новой матрице или вектору. Действие оператора меняет саму матрицу (во многих других операторах пакета **LinearAlgebra** для этого применяется опция **inplace**). Поэтому, если дважды подряд выполнить $\text{Map}(x \rightarrow x-2, A)$, то все элементы матрицы уменьшатся на 4.

- **Map2**. То же, что **Map**, но действие производится с параметром. Если на матрицу (5.8) подействовать оператором $\text{Map2}(a, x) \rightarrow \exp(a*x)*x, 2, A$, то получим

$$\begin{bmatrix} e^2 & 2e^4 & -e^{(-2)} \\ 0 & e^2 & -e^{(-2)} \end{bmatrix}.$$

Здесь a — параметр, значение которого взято равным 2, x — элемент матрицы, к которому применяется действие. Если поменять места параметра и аргумента: $\text{Map2}(x, a) \rightarrow \exp(a*x)*x, 2, A$, то получим

$$\begin{bmatrix} 2e^2 & 2e^4 & 2e^{(-2)} \\ 2 & 2e^2 & 2e^{(-2)} \end{bmatrix}.$$

- **MatrixAdd(A, B, k1, k2)**. Линейная комбинация матриц A и B . Действие эквивалентно операции $A*k1+B*k2$. Если указать опцию замещения **MatrixAdd(A, B, k1, k2, inplace)**, то результат будет помещен в матрицу A , иначе — будет создана новая матрица.
- **MatrixExponential(A, t)**. Выполнение операции e^{At} для квадратной матрицы. Частный случай матричной функции

MatrixFunction. По умолчанию $t=1$. Пусть дана матрица

$$A := \begin{bmatrix} 1 & -2 \\ 2 & 1 \end{bmatrix}.$$

Действие **MatrixExponential(A)** дает следующий результат:

$$\begin{bmatrix} e \cos(2) & -e \sin(2) \\ e \sin(2) & e \cos(2) \end{bmatrix}.$$

Вычисления производятся по формуле $\exp(At) = E + At + 1/2!A^2t^2 \dots$, где E — единичная матрица.

- **MatrixInverse(A).** Матрица, обратная к A . В простейших случаях эту операцию можно выполнить простым делением: $1/A$, или A^{-1} , или $A^{**}(-1)$. При этом не потребуется подгружать пакет **LinearAlgebra**. Оператор **MatrixInverse** имеет различные методы выполнения задачи: **LU**, **Cholesky**, **subs**, **integer**, **univar**, **polynom**, **complex**, **rational**, **pseudo**, **none**.

Например, оператор **MatrixInverse(A, method=pseudo)** позволяет найти псевдообратную матрицу B , удовлетворяющую соотношениям $ABA = A$, $BAB = B$.

- **MatrixMatrixMultiply(A, B).** Умножение матриц A и B . Если добавить опцию **inplace**, то результат будет помещен в матрицу A . Матрицы можно также умножать с помощью точки: $A.B$, при этом не потребуется подгружать пакет **LinearAlgebra**.
- **MatrixNorm(A).** Норма матрицы. Допустима форма **Norm(A, 1)**. Имеются четыре варианта нормы:

- **MatrixNorm(A, 1) = $\max V_i$** — максимальная норма столбцов, где норма столбца есть сумма модулей его элементов. Для матрицы

$$A := \begin{bmatrix} -1 & 1 \\ 2 & 2 \end{bmatrix} \quad (5.9)$$

эта норма равна 3;

- **MatrixNorm(A, infinity) = $\max R_i$** — максимальная норма строк, где норма строки есть сумма модулей ее элементов. Для матрицы (5.9) эта норма равна 4;

- `MatrixNorm(A, Frobenius)` = $\sqrt{\sum_{i,j} a_{i,j}^2}$ — норма Фробениуса¹ — квадратный корень из суммы квадратов всех элементов матрицы. Для матрицы (5.9) эта норма равна $\sqrt{10}$;
- `MatrixNorm(A, 2)` = $\sqrt{\max(\lambda_i)}$ — евклидова норма или спектральная норма — квадратный корень из максимального собственного числа λ_i матрицы AA^T . Для матрицы (5.9) эта норма равна $2\sqrt{2}$. Действительно, имеем произведение

$$AA^T = \begin{bmatrix} 2 & 0 \\ 0 & 8 \end{bmatrix}.$$

Максимальное собственное число этой матрицы равно 8, отсюда `MatrixNorm(A, 2)` = $2\sqrt{2}$.

По умолчанию берется норма `infinity`.

- `MatrixPower(A, n)`. Матрица A в степени n . Степень может быть и не целой, и отрицательной. Например, `MatrixPower(A, 2)` = A^2 , а `MatrixPower(A, -1)` — обратная матрица. Отметим, что не всегда матрица `MatrixPower(A, A, 1/2)` равна A .
- `MatrixScalarMultiply(A, b)`. Умножение всех элементов матрицы A на число b . Другое название этой операции — просто `ScalarMultiply(A, b)`, а еще проще можно умножить $A * n$.
- `MatrixVectorMultiply(A, v)`. Умножение матрицы на вектор. Можно также умножать с помощью точки: $A \cdot v$.
- `MinimalPolynomial(A, x)`. Минимальный полином матрицы A по переменной x — полином минимальной степени, делящий любой полином, для которого A является корнем, в том числе согласно теореме Гамильтона–Кэли², и характеристический полином. Иначе: минимальный полином — это аннулирующий полином минимальной степени. Проверим, что матрица A обнуляет свой минимальный полином. Получим полином по переменной x : `Mr:=MinimalPolynomial(A, x)`. Создадим процедуру `P:=unapply(Mr, x)`. Подставив в процедуру матрицу A , т. е. вычислив `P(A)`, получим нулевую матрицу. Делимость характеристического полинома на минимальный проверим операцией `divide(CharacteristicPolynomial(A, x), Mr)`. Ответ получится `true`.

¹*Ferdinand Georg Frobenius* (1849–1917) — немецкий математик, ученик Карла Вейерштрасса.

²*William Rowan Hamilton* (1805–1865) — ирландский математик, *Arthur Cayley* (1821–1895) — английский математик.



- **Minor(A, i, j)**. Минор элемента a_{ij} матрицы **A**. Для матрицы (5.7), с. 251, имеем $\text{Minor}(\mathbf{A}, 3, 3) = -4$, а $\text{Minor}(\mathbf{A}, 3, 2) = 4$.
- **Multiply(A, B)**. Произведение матриц. Матрицы можно умножить и точкой: $A \cdot B$. Если указать опцию **inplace**, то результат будет помещен в матрицу **A**, иначе — будет создана новая матрица.
- **Normalize(V, p)**. Нормализация вектора — деление его компонент на модуль вектора так, что получившийся вектор имеет норму равную 1. Норма вычисляется по формуле $\left(\sum_{i=1}^n |V_i|^p\right)^{1/p}$. Норма номер 2 имеет также название **Euclidean** или **Frobenius**. Норма номер 1 означает деление элементов вектора на сумму модулей элементов. По умолчанию принята норма **infinity**, равная максимальному модулю компонента. Если указать опцию **inplace** или, что то же, **inplace=true** (режим замещения), то результат будет помещен в исходный вектор. Примеры: с. 350, 405.
- **NullSpace(A)**. Ядро матрицы. Если вычислить $\mathbf{KR} := \text{NullSpace}(\mathbf{A})$, то проверка $\mathbf{A} \cdot \mathbf{KR}[1]$ даст нулевой вектор.
- **OuterProductMatrix(U, V)**. Матрица внешнего произведения векторов. Элементы матрицы получаются по формуле $P_{i,j} = U_i V_j$. Например, даны векторы $\mathbf{U} := \text{Vector}([1, 2, 3])$ и $\mathbf{V} := \text{Vector}([x, y])$. Внешнее произведение **OuterProductMatrix(U, V)** имеет вид

$$\begin{bmatrix} x & y \\ 2x & 2y \\ 3x & 3y \end{bmatrix}.$$

Если поменять порядок сомножителей, то получится транспонированная матрица.

- **Permanent(A)**. Перманент квадратной матрицы — величина, вычисляемая по правилу определителя, но без учета смены знаков, т. е. сумма всех произведений элементов матрицы по одному из каждой строки и столбца. Так, определитель матрицы (5.9) равен -4 , а перманент равен нулю. В теории графов [29] так вычисляют число совершенных паросочетаний в двудольных графах по матрице смежности.
- **QRDecomposition(A)**. QR-разложение матрицы — представление матрицы в виде произведения ортогональной и верхней треугольной матрицы.

- `RandomMatrix(m, n)`. Матрица размером $m \times n$ случайных чисел. По умолчанию числа берутся из интервала $-99..99$. Можно указать интервал чисел (`generator`) и форму матрицы: например, оператор `RandomMatrix(2, 3, generator=0..100, outputoptions = [shape=triangular[lower]])` создает матрицу

$$\begin{bmatrix} 19 & 0 & 0 \\ 20 & 16 & 0 \end{bmatrix}.$$

- `RandomVector(n)`. Вектор из n случайных компонент. Создается аналогично `RandomMatrix`. Можно явно указать форму, например вектор-строку, и область изменения случайных чисел: `RandomVector[row](4, generator=rand(1..5)/10)`. Получим

$$\left[\frac{3}{10}, \frac{2}{5}, \frac{2}{5}, \frac{1}{2} \right].$$

Здесь использован оператор `rand` для создания случайных дробей. По умолчанию принята форма `column` — вектор-столбец.

- `Rank(A)`. Ранг матрицы. Определяется методом исключения Гаусса.
- `RationalCanonicalForm(A)`. Первая естественная нормальная форма матрицы, или матрица Фробениуса (`FrobeniusForm`).
- `Row(A, k)`. Строка матрицы. Указывается номер или список номеров. Если $k < 0$, то номер берется с конца: например, $k = -1$ соответствует последней строке матрицы.
- `RowDimension(A)`. Число строк матрицы.
- `RowOperation`. Линейные операции со строками. Действие, аналогичное `ColumnOperation`. Например, если к матрице

$$A := \begin{bmatrix} 4 & -6 & 3 \\ 1 & -1 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$

применить операцию `RowOperation(A, [1, -2], 10)`, означающую, что к строке 1 надо добавить 10 предпоследних строк, то получим следующую матрицу:

$$\begin{bmatrix} 14 & -16 & 13 \\ 1 & -1 & 1 \\ 0 & 1 & 2 \end{bmatrix}.$$

- `RowSpace(A)`. Базисные векторы строк матрицы.

- **ScalarMatrix(s, m, n)**. Диагональная матрица размером $m \times n$ со значениями s по диагонали. Например, оператор **ScalarMatrix(a+x, 2, 3)** задает матрицу

$$\begin{bmatrix} a+x & 0 & 0 \\ 0 & a+x & 0 \end{bmatrix}.$$

- **ScalarMatrix(s, m, n)**. Диагональная матрица размером $m \times n$.
- **ScalarVector(s, i, n)**. Вектор-столбец длиной n , компонента номер i равна s , остальные — нули. Для вектора-строки надо указать **ScalarVector[row](s, i, n)**.
- **SingularValues(A)**. Сингулярные числа матрицы A . Вычисляются по формуле $\sqrt{\lambda_i}$, где λ_i — собственные числа матрицы AA^T . По умолчанию результат оператора — вектор-столбец. Если указать **SingularValues(A, output='list')**, то результат будет в виде списка.
- **StronglyConnectedBlocks(A)**. Диагональные блоки матрицы, содержащие только те элементы, от которых зависят определитель и собственные значения матрицы (сильно связанные блоки). Так, определитель матрицы

$$A := \begin{bmatrix} a & b & c \\ x & y & z \\ 0 & 0 & w \end{bmatrix}$$

не зависит от c и z и равен $ayw - xbw$. Результатом действия оператора **StronglyConnectedBlocks(A)** будут два выделенных блока

$$[[w]], \begin{bmatrix} a & b \\ x & y \end{bmatrix}.$$

Заменим элемент $a_{1,2} = b$ нулем:

$$A := \begin{bmatrix} a & 0 & c \\ x & y & z \\ 0 & 0 & w \end{bmatrix}. \quad (5.10)$$

В этой матрице оператор выделяет три простых блока: $[[w], [a], [y]]$.

- **SubMatrix(A, r, c)**. Матрица, полученная из строк и столбцов матрицы A . Здесь r, c — списки номеров строк и столбцов. Отрицательные значения соответствуют номерам с конца списков строк и столбцов. Например, оператор **SubMatrix(A, [2, -1], [1..3,**

3)) применительно к матрице (5.10) дает следующую матрицу:

$$\begin{bmatrix} x & y & z & z \\ 0 & 0 & w & w \end{bmatrix}.$$

Хотя число столбцов получилось больше, чем в исходной матрице, эту матрицу условно называют подматрицей матрицы \mathbf{A} .

- **SubVector(V, s)**. Вектор, полученный из элементов вектора V . Здесь s — списки номеров элементов. Отрицательные значения соответствуют номерам с конца списка. Например, оператор **SubVector(V, [4..-1, 3, -3..-1, 2])** применительно к вектору **Vector[row]([a, 2, 3, 4, 5, f])** дает вектор $[4, 5, f, 3, 4, 5, f, 2]$.
- **SylvesterMatrix(p, q, x)**. Матрица Сильвестра¹ полиномов $p(x)$ степени m и $q(x)$ степени n . Строки матрицы заполняются коэффициентами полиномов в порядке убывания степеней со сдвигом на каждой новой строке. Число строк, заполненных коэффициентами p , равно n , коэффициентами q — m . Размер полученной квадратной матрицы $m+n$. Пример: $p = x - 2$, $q = x^2 - (a + b)x + ab$. Матрица имеет вид

$$S := \begin{bmatrix} 1 & -2 & 0 \\ 0 & 1 & -2 \\ 1 & -b-a & ab \end{bmatrix}.$$

Определитель этой матрицы равен результату полиномов — произведению попарных разностей их корней: $(b - 2)(a - 2)$. Другой способ получить результат не требует подключения пакета линейной алгебры: **resultant(p, q, x)**.

- **SylvesterSolve(A, B, C)**. Решение матричного уравнения Сильвестра $AX + XB = C$. При $B = A^T$ уравнение совпадает с уравнением Ляпунова, с. 252.
- **ToeplitzMatrix(S)**. Топлицева² матрица порядка n , полученная по списку S . Список должен состоять из $m = 2n - 1$ элементов. Элементы списка выстраиваются на диагоналях, параллельных главной, по правилу $t_{i,j} = S_{n+i-j}$, $n = (m + 1)/2$. Например, оператор **ToeplitzMatrix([a, b, c, d, 1, 2, 3])** дает следующую матрицу:

$$\begin{bmatrix} d & c & b & a \\ 1 & d & c & b \\ 2 & 1 & d & c \\ 3 & 2 & 1 & d \end{bmatrix}.$$

¹James Joseph Sylvester (1814–1897) — английский математик.

²Toeplitz Otto (1881–1940) — немецкий математик.



Для сравнения: матрица Ганкеля (с. 249) образуется из элементов списка, расставленных по диагоналям, параллельным побочной.

- **Trace(A)**. След матрицы — сумма диагональных элементов.
- **Transpose(A)**. Матрица, полученная транспонированием A . Если указать опцию замещения **Transpose(A, inplace)**, то результат будет помещен в матрицу A , иначе — будет создана новая матрица. Две другие формы обращения к оператору: A^+ и A^*T .
- **TridiagonalForm(A)**. Приведение матрицы к трехдиагональной форме. Для матрицы

$$A := \begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 9 \\ 3 & 9 & 0 \end{bmatrix}$$

оператором $(Q, T) := \text{TridiagonalForm}(A, \text{output}=['Q', 'T'])$ получим

$$Q, T := \begin{bmatrix} 1.00 & 0.00 & 0.00 \\ 0.00 & -0.55 & -0.83 \\ 0.00 & -0.83 & 0.55 \end{bmatrix}, \begin{bmatrix} 1.00 & -3.61 & 0.00 \\ -3.61 & 9.54 & 5.31 \\ 0.00 & 5.31 & -5.54 \end{bmatrix}.$$

Разложение можно проверить: $Q \cdot T \cdot \text{Transpose}(Q)$. Получится исходная матрица, но с некоторой погрешностью, которая уменьшается с увеличением заданной точности счета **Digits**.

- **UnitVector(i, n)**. Единичный n -мерный вектор, в котором i -я компонента равна 1, остальные — нулю.
- **VandermondeMatrix(V)**. Матрица Вандермонда¹ порядка n на списке V длиной n . Матрица строится по правилу $M_{i,j} = V_i^{j-1}$. Например, если $V := \langle x, y, 2 \rangle$, то получим

$$M = \begin{bmatrix} 1 & x & x^2 \\ 1 & y & y^2 \\ 1 & 2 & 4 \end{bmatrix}.$$

Определитель матрицы Вандермонда равен $\prod_{i < j} (v_j - v_i)$. В рассмотренном примере $\det(M) = (-2 + x)(-2 + y)(y - x)$.

- **VectorAdd(U, V, k1, k2)**. Линейная комбинация векторов. Действие эквивалентно операции $U * k1 + V * k2$. Если указать опцию замещения **VectorAdd(U, V, k1, k2, inplace)**, то результат

¹*Alexandre-Theophile Vandermonde* (1735–1796) — французский музыкант (скрипач) и математик. Один из основоположников современной алгебры.

будет помещен в вектор U , иначе — будет создан новый вектор. Можно складывать как векторы-строки, так и векторы-столбцы, дополнительно ничего для этого указывать не надо — просто требуется, чтобы слагаемые были одного типа.

- **VectorAngle(V1, V1)**. Угол между векторами. Результат (в радианах) вычисляется с помощью скалярного произведения.
- **VectorMatrixMultiply(u, M)**. Произведение вектора и матрицы. Аналог обычного произведения $u \cdot M$.
- **VectorNorm(V, p)**. Норма или, точнее, p -норма вектора; $p \geq 1$. Если $p = 2$, то это обычная евклидова норма. В общем случае для вектора из n компонент имеем значение $\left(\sum_{i=1}^n |V_i|^p \right)^{1/p}$.
- **VectorScalarMultiply(V, s)**. Покомпонентное умножение вектора на скаляр. Другое название того же оператора **ScalarMultiply(V, s)**.
- **ZeroMatrix(m, n)**. Нулевая матрица размером $m \times n$.
- **ZeroVector(n)**. Нулевой n -мерный вектор. По умолчанию задается вектор-столбец. Вектор-строка имеет вид **ZeroVector[row](n)**.
- **Zip('P', A, B)**. Двухместная операция над элементами матриц или векторов. Под 'P' понимается либо процедура с двумя аргументами, либо просто арифметическая операция '+', '*', '-', '\'. Пусть $P := \text{proc}(x, y) \ x * \sin(y) \ \text{end}$. Рассмотрим матрицы

$$A = \begin{bmatrix} a & x & p \\ b & y & q \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix}.$$

Применим к ним операцию **Zip('P', A, B)**. Получим

$$\begin{bmatrix} a \sin(1) & x \sin(3) & p \sin(5) \\ b \sin(2) & y \sin(4) & q \sin(6) \end{bmatrix}.$$

В общем случае аргументами операции могут быть матрицы (и векторы) разной размерности. Недостающие элементы по умолчанию просто отбрасываются. Так, если $A := \langle \langle a, b \rangle | \langle x, y \rangle \rangle$, то в результате будет отсутствовать последний столбец. Но дополнить матрицу до необходимого размера можно с помощью "запасного" элемента, который в операторе указывается четвертым и ставится на пустое место. Например, для укороченной



матрицы $A := \langle \langle a, b \rangle | \langle x, y \rangle \rangle$ и прежней матрицы B оператор `Zip('P', A, B, z1)` с "запасным" элементом `z1` дает

$$\begin{bmatrix} a \sin(1) & x \sin(3) & z1 \sin(5) \\ b \sin(2) & y \sin(4) & z1 \sin(6) \end{bmatrix}.$$

5.5. Графики и рисунки

5.5.1. Простые возможности операторов `plot` и `plot3d`

Для построения графиков и рисунков, в том числе и анимированных, в **Maple** имеются весьма разнообразные средства. Большинство из них требует подключения соответствующего пакета. Самые простые графики можно построить оператором `plot`. Для него не нужно дополнительных средств. Перечислим некоторые простейшие варианты построения графиков функций.

- Графики элементарных функций. Например: `plot(cos)`. Аргумент можно не указывать (рис. 333). Допустимы и такие непривычные выражения, как `plot(sin+cos)`. Эти графики служат либо как иллюстративный материал, либо как простейший справочный. Такое обращение к оператору `plot` используется редко.
- График (рис. 334) неэлементарной функции $y = \operatorname{erfi}(x)^1$ с указанием пределов изменения аргумента $x = -1..1$ и с ограничением (от $-0,9$ до 2) на ось ординат: `plot(erfi(x), x=-1..1, -0.9..2)`. Если не указывать ограничение на ось y , то будет построен весь график (если, конечно, он не уходит в бесконечность).

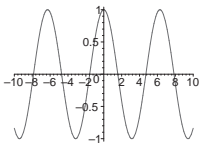


Рис. 333

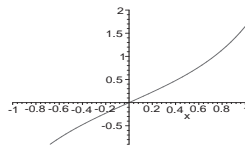


Рис. 334

- Две кривые на одном графике (рис. 335).
`> plot([sin(x)/x, exp(-x)], x=0..8,`
`> color=[red,blue], style=[point,line]);`

¹Комплексная функция ошибок $\operatorname{erfi}(x) = \frac{2}{\sqrt{\pi}} \int_0^x \exp(t^2) dt$. В механике, например, через эту функцию выражается ядро сдвиговой релаксации [41].

Указаны цвет и тип линий. Все опции можно найти, если набрать команду `plot/options`, навести на нее курсор и нажать клавишу помощи F1 или пройти в окне помощи по маршруту Graphics-> 2-D -> Options.

- Параметрически заданная функция. Например, удлиненная циклоида: `plot([t-4*sin(t),1-4*cos(t),t=-9..9], scaling=constrained)`. Необязательная опция `scaling=constrained` означает, что соблюден одинаковый масштаб осей (рис. 336).

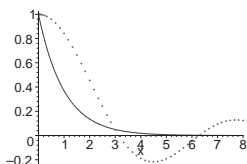


Рис. 335

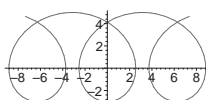


Рис. 336

- Два графика (логарифмическая спираль и лемниската, рис. 337) в полярной системе координат строятся так:

```
> plot([[exp(t/8),t,t=0..7],
>       [2*sqrt(cos(2*t)),t,t=0..2*Pi]],
>       coords=polar,thickness=[1,6]);
```

В опции указано, что толщина первой кривой (логарифмической спирали) равна 1, второй — 6.

- Кусочно-линейная функция, заданная угловыми точками `plot([[0,0],[1,4],[2,2],[4,4]])` (рис. 338)

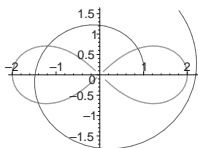


Рис. 337

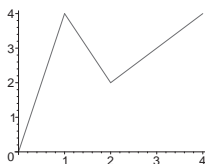


Рис. 338

- Функция, заданная точками (рис. 339)

```
> plot([[[-3,2],[1,3],[2,4],[3.4,2.9],[4,3]],x=-4..5,
>       style=point,symbol=circle,symbolsize=20];
```

Здесь точки помечены окружностями (circle). Можно использовать и другие символы: asterisk, box, cross, diagonalcross, diamond, point, solidbox, solidcircle, soliddiamond.

- Функция двух переменных (рис. 340)

```
> plot3d(sin(4*x)*y, x=0..2, y=1..3, axes=normal);
```

По умолчанию оператор `plot3d` оси не рисует, поэтому указана опция `axes=normal`. Оператор `plot3d` имеет множество опций, узнать о которых можно по справке `Graphics->3-D->plot3d`.

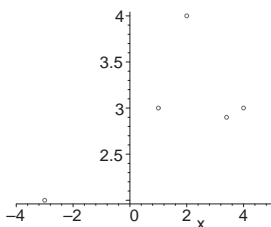


Рис. 339

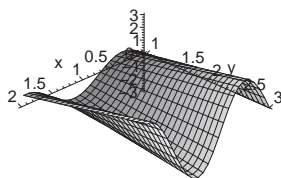


Рис. 340

5.5.2. Основные опции оператора `plot`

Опции оператора `plot` являются необязательными, но для правильного отображения информации часто приходится подбирать различные варианты. В некоторых случаях опции дублируют друг друга, в некоторых они не всегда четко работают. И хотя, несмотря на существующие описания и примеры, наилучший вариант находится опытным путем, базовые определения дать необходимо. Нижеследующие описания не являются точной копией справочной информации **Maple**, поэтому для более ясного представления о всех возможностях оператора рекомендуем все-таки обращаться также к справке по системе.

- `axes=f`, где `f` тип осей: `boxed`, `frame`, `none`, `normal`.
- `axesfont`. Шрифт подписей на осях. Задается в виде `axesfont=[Семейство, стиль, размер]`, например:

```
plot(x, x=0..1, axesfont=[TIMES, ROMAN, 20])
```

Существуют четыре семейства шрифтов: `TIMES`, `COURIER`, `HELVETICA`, `SYMBOL`. Стиль для семейства `TIMES` может быть одним из следующих: `ROMAN`, `BOLD`, `ITALIC`, `BOLDITALIC`. Три возможных стиля для семейств `COURIER`, `HELVETICA`: `BOLD`, `OBLIQUE`, `BOLDOBLIQUE`. Семейство `SYMBOL` стили не поддерживает.

- **caption.** Подпись графика, например, `caption= "Скорость точки А"`. Пишется снизу графика. В классической версии `swmaple.exe` опция может не работать.
- **color.** Цвет графика. Есть два варианта обозначения цвета — старый и новый. Старый вариант пишется строчными буквами с кавычками или без, например, `color=blue`.

Есть также цвета `aquamarine`, `black`, `brown`, `coral`, `cyan`, `gold`, `green`, `gray` или `grey`, `khaki`, `maroon`, `magenta`, `navy`, `orange`, `pink`, `plum`, `red`, `sienna`, `tan`, `turquoise`, `violet`, `wheat`, `white`, `yellow`.

Новых имен для цветов значительно больше, вводить их надо с кавычками, названия всегда начинаются с прописной буквы и большинство их совпадает со старыми именами, например: `Brown`. Цвет можно подобрать свой в формате RGB, т. е. красный, зеленый, синий. Красный цвет имеет вид `МойЦвет:= COLOR(RGB,1,0,0)`; Кривая при этом оформляется так:

```
plot(tan(x), x = -1..1, color=МойЦвет);
```

- **coords.** Тип системы координат. По умолчанию принимается декартова прямоугольная система. Есть также еще следующие: `bipolar`, `cardioid`, `cassinian`, `elliptic`, `hyperbolic`, `invcassinian`, `invelliptic`, `logarithmic`, `logcosh`, `maxwell`, `parabolic`, `polar`, `rose`, `tangen`.
- **filled.** Заполнение замкнутого пространства между кривой и осью x цветом, указанным в опции `color`. Если такого пространства нет, то вся область заливается этим цветом. По умолчанию `filled=false`. Вместо `filled=true` можно просто указать `filled`.
- **gridlines.** Опция указывает наличие сетки, если `gridlines=true`. По умолчанию `gridlines=false`. Линии сетки привязываются к меткам на осях (к основным линиям — толще, к промежуточным — тоньше). В классической версии `swmaple.exe` опция может не работать.
- **labels.** Подписи осей координат. Допустимы русские буквы. Например, `labels=["Время", "Скорость"]`. Однако при сохранении рисунка в виде графического файла eps (Encapsulated PostScript) русские буквы не воспринимаются. При этом версия `swmaple.exe` не определяет еще и цвета графиков и дает черно-белое изображение.
- **labelfont.** Шрифт меток на осях. Задается аналогично `axesfont`.

- **legend.** Расшифровка обозначений кривых на осях. Совершенно необходимая опция, если на одном графике несколько кривых. При этом надо кривые как-то различать: толщиной, стилем, цветом. Снизу от графика будет указан образец линии и соответствующая расшифровка. Например, если на графике две кривые, то можно записать так: `legend=["Координата", "Скорость"]`. С помощью дополнительной опции `legendstyle` можно уточнить оформление подписей.
- **linestyle.** Стилль линии. Варианты: `solid`, `dot`, `dash`, `dashdot`, `longdash`, `spacedash`, `spacedot`. При этом можно указать стилль `linestyle=dash` или его номер (от 1 до 7), например `linestyle=1`. Этот стилль (сплошная линия) принят по умолчанию.
- **numpoints.** Число точек, из которых состоит кривая графика. По умолчанию `numpoints=50`. Опция имеет смысл в основном для построения графиков неявных функций (`implicitplot` из пакета `plots`) для сглаживания кривых. При этом время счета существенно увеличивается. В других случаях оператор `plot` автоматически выбирает число точек.
- **resolution.** Число пикселей, отводимых на график по горизонтали (по умолчанию 200). Фактически меняет масштаб графика.
- **scaling.** Тип масштабирования осей. При сохранении пропорций `scaling=constrained`. По умолчанию принято `scaling=unconstrained`.
- **style.** Стилль линии. Возможны варианты `line`, `point`, `polygon`, `polygonoutline`.
- **symbol.** Тип символа, выводимого в виде точки, если график отображается точками (с. 265).
- **symbolsize.** Размер выводимого символа. По умолчанию `symbolsize=10`. Не применим к символу `POINT`.
- **thickness.** Толщина линии от 0 (самая тонкая, но еще видимая линия) до 15. По умолчанию `thickness=0`. Этот параметр берется оператором по модулю 16, т. е. `thickness=16` соответствует значению по умолчанию.
- **tickmarks.** Способ разметки осей. Если `tickmarks=[m, n]`, то это значит, что по оси x будет m меток и n меток — по оси y . Можно также подписать каким-либо образом нужную ординату или абсциссу. Так, если задать `tickmarks=[default, [1="A1", 12="B2", 19="C0"]]`, то по оси x метки будут расставлены по умолчанию, а ординаты при $y = 1$, $y = 12$ и $y = 19$ будут подписаны, соответственно, как "A1", "B2" и "C0".

- **title.** Заголовок графика, пишется сверху. Для переноса на новую строку можно использовать командный знак переноса `\n`, принятый в формате вывода. Допустимы русские буквы. Однако при сохранении рисунка в виде графического файла eps (Encapsulated PostScript) русские буквы не воспринимаются.
- **titlefont.** Шрифт заголовка, см. `axesfont`, с. 265.
- **view=[xmin..xmax, ymin..ymax].** Границы графика. Здесь `xmin` и `xmax` — соответственно нижняя и верхняя границы оси x . Аналогично `ymin` и `ymax` — границы оси y . Таким образом, можно обрезать график до нужных размеров.

5.5.3. Пакет ImageTools

Для работы с изображениями в **Maple** имеется специальный пакет **ImageTools**. Возможности этого пакета не столь велики, как, например, у аналогичных пакетов **MATLAB**, однако для альтернативного способа построения графиков, импорта файлов изображений и их обработки в нем есть достаточно операторов. Есть также и команды работы со звуком (`Clip`). Приведем программу построения графика функции $y = 50 + 500 \sin(x/5)/x$ (рис. 341).

```
> with(ImageTools):
> f:=(r,c) -> if r=round(-sin(c/5)/c*500+50)
>               then 1 else 0 fi:
> img := Create(90,90,1,f):
> View(1-img);
```

Здесь оператор `Create(90,90,1,f)` создает изображение, фактически массив из 90 строк и 90 столбцов вещественных чисел от 0 до 1. Белый цвет соответствует 1, черный — 0. Функция создается процедурой `f` с аргументами `r` и `c` (`row` — ряд, `column` — столбец). Аргументы этой функции пробегают свои области определения 1..90 и 1..90. Если номер ряда совпадает с округленным значением функции $y(x)$, то в это место выставляется белая точка, иначе — черная.

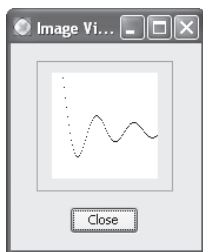


Рис. 341

Потом график строится в негативе `View(1-img)`, хотя можно было бы оставить его и в такой форме, как мелом на черной доске. Результат

выводится в отдельное окно, которое потом требуется закрыть, для того чтобы программа работала дальше.

Другой способ построения изображения — непосредственное присвоение значений массиву `img`. Но сначала надо его создать (пока пустой) оператором `img := Create(90,90)`. Затем в цикле легко получить изображение:

```
> for i to 90 do
>   img[round(-sin(c/5)/c*500+50), i]:=1:
> od:
```

Таким образом, пакет `ImageTools` позволяет получить доступ к любому пикселю изображения. Это можно использовать, например, при обработке готовых графических файлов. В следующей короткой программе для выделения контуров на фотографии применяется фильтр Собеля¹. Этот алгоритм используется во многих графических редакторах, например, Adobe Photoshop².

Для работы фильтра создается матрица $G_1 = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$,

которая собирает восемь пикселей вокруг пикселя с номером ячейки i, j в сумму $X = \sum_{k=1}^3 \sum_{m=1}^3 G_{1,km} A_{km}$, а транспонированная матрица

$G_2 = G_1^T$ собирает другую сумму $Y = \sum_{k=1}^3 \sum_{m=1}^3 G_{2,km} A_{km}$, где A —

подматрица размером 3×3 , вырезанная из матрицы изображения вокруг элемента i, j , т. е. $A_{2,2} = img_{i,j}$. Затем на место i, j изображения ставится новое значение $\sqrt{X^2 + Y^2}$. Так из исходного изображения на рис. 342 выделяется его контурный вариант, рис. 343.



Рис. 342



Рис. 343

¹*Sobel Irwin* — американский ученый, сотрудник Hewlett-Packard Laboratories.

²В меню Adobe Photoshop: Фильтры->Стилизация->Свечение краев или Filter->Stylize->Trace Contour

Оператор `Read` считывает файл изображения, а оператор `RGBtoGray` преобразует его в черно-белый вариант¹. Размер изображения определяется операторами `Width` и `Height`. Для того чтобы воспользоваться оператором выделения подматрицы нужных размеров из матрицы изображений, необходимо преобразовать изображение, которое по форме только похоже на матрицу, в настоящую матрицу `IMG:=convert(im,Matrix)`. Переменная `IMG` теперь имеет тип матрицы и оператор `SubMatrix` может с ней работать. Фильтр Собеля в этой программе немного обобщен. Введен параметр z , который принимает значение 2 для классического фильтра Собеля. С увеличением z растет контрастность изображения. Для аналогичного фильтра Щарра² имеем `s:=[3,10,3]`.

Результат записывается в файл оператором `Write` и сообщается размер записанного файла, в данном примере — 6970. Отметим, что упомянутые здесь операторы чтения и записи принадлежат пакету `ImageTools`. В других пакетах (`AudioTools`, `FileTools`, `Sockets`) есть одноименные операторы, но они работают иначе.

Программа 1 (P1.mws)

```
> restart;with(ImageTools):# Фильтр Собеля
> with(LinearAlgebra):
> img0:= RGBtoGray(Read("C:\\MSU.jpg")):
> Фильтр:=proc(z,im)
> local IMG,s,G1,G2,i,j,A,X,Y,w,h:
> global img;
> w:=Width(im): h:=Height(im):
> img:=Create(h,w):
> IMG:=convert(im,Matrix):
> s:=[z/2,z,z/2]:
> G1:=Matrix([s,[0,0,0],[-s]]);
> G2:=Transpose(G1):
> for i from 2 to h-1 do
>   for j from 2 to w-1 do
>     A:=SubMatrix(IMG,[i-1,i,i+1],[j-1,j,j+1]):
>     X:=add(add(A[k,m]*G1[k,m],k=1..3),m=1..3);
>     Y:=add(add(A[k,m]*G2[k,m],k=1..3),m=1..3);
```

¹ Другой способ преобразования в черно-белое изображение: отделение одного из слоев `GetLayer(Read("C:\\MSU.jpg"),1)`.

² В. Jahne, H. Scharf, and S. Korkel. *Principles of filter design*. In Handbook of Computer Vision and Applications. Academic Press, 1999.

```

>   img[i, j]:=sqrt(X^2+Y^2):
>   od:
> od:
> img:
> end proc:
> Фильтр(1, img0):img1:=1-img: View(img1):
> Write("C:\\MSU2.jpg", img1);

```

6970

5.5.4. Пакет plots

Как всегда, пакет вызывается командой `with(plots)`. Если после команды ставить точку с запятой, то на экране появятся все операторы этого пакета. В Maple 13 их 57. Перечислим некоторые.

- **animate**. Анимация графиков. Например, можно показать процесс рисования эллипса, имитируя движение карандаша по бумаге.

```

> animate( plot, [[4*cos(t), sin(t), t=0..T]],
>          T=0..-2*Pi, scaling=constrained, frames=10);

```

После того как сработает этот оператор, на экране появятся пустые оси координат. Далее в верхней панели инструментов и команд Maple надо найти магнитофонную панель (рис. 412, с. 354) и запустить анимацию клавишей «Пуск» (Play the animation).

- **animatecurve**. Анимация процесса построения кривой. По сути, совпадает с `animate`, но имеет несколько другую форму. Так, например, предыдущий пример можно реализовать следующим образом:

```

> animatecurve([4*cos(t), -sin(t), t=0..2*Pi],
>              scaling=constrained, frames=10);

```

Для того чтобы кривая вычерчивалась по часовой стрелке, знак параметра t в уравнении эллипса был заменен на обратный. Верхний предел для t должен быть положительным.

- **animate3d**. Анимация трехмерных графиков. В отличие от `animate` здесь поверхности не рисуются, а меняются. Последний параметр t отвечает за изменение формы, два первых x и y — аргументы функции:

```

> animate3d(t^2*cos(2*x)+sin(y*t),
>           x=-2..2, y=0..2, t=1..3, frames=50);

```

Если просто поменять местами параметр и аргументы $x=-2..2, t=1..3, y=0..2$, то система воспримет параметр t

аргументом, а аргумент y — параметром, и картина, конечно же, изменится.

- `changecoords`. Замена системы координат. Например, команда `plot([2*t, t, t=0..2*Pi])` построит прямую $x = 2t$, $y = t$, $0 \leq t \leq 2\pi$, заданную параметрически, а команда `changecoords(plot([2*t, t, t=0..2*Pi]), polar)` дает график одного витка спирали Архимеда $\rho = 2\varphi$, $0 \leq \varphi \leq 2\pi$ в полярных координатах.
- `complexplot`. Построение комплекснозначных функций. Оператор `complexplot([1+2*I, 2+I, 1-I], x=0..2, style = point)` отобразит три точки $z_1 = 1 + 2i$, $z_2 = 2 + i$, $z_3 = 1 - i$ на комплексной плоскости. Если же опустить опцию `style=point`, то эти точки будут соединены ломаной. Оператор `complexplot(x+2*x*I, x=0..1)` построит отрезок прямой $y = 2x$.
- `contourplot`. Построение линий уровня функции двух переменных. Оператор `contourplot(x^2+y^2, x=-1..1, y=-1..1)` построит серию концентрических окружностей — линии уровня функции $z = x^2 + y^2$. Если добавить опцию `filledregions=true`, то кольца будут еще и красиво раскрашены. Число линий можно менять параметром `contours`. По умолчанию `contours=8`.
- `display`. Вывод на экран графических структур. Один из основных операторов пакета `plots`. Пример: построены графики, но результат построений выведен во временные переменные `a:=plot(sin): b:=plot(3*x+2, x=0..2):`¹. На экран эти графики (в одних осях) можно вывести оператором `display(a, b, axes=None, style=point)`. Опции относятся сразу к обоим графикам. В данном случае графики строятся точками, а оси не изображаются. Этот оператор с опцией `insequence=true` часто используется для анимации (с. 354, 359, 364, 385).
- `dualaxisplot`. Построение двух графиков с двумя осями ординат. Для первого графика слева, второго — справа. Например, `dualaxisplot(sin(x), x+1, x=0..2)`.

¹Отметим, что если после этих операторов, результат которых присвоен переменным, поставить не двоеточие, а точку с запятой, то результат будет тот же, но на экран будет выведено цифровое содержание графиков вместо самих графиков. Иногда это полезно, особенно для выявления ошибок в программе, но обычно эта информация лишняя.

- `fieldplot`. Векторное поле на плоскости. Например, `fieldplot([1,x],x=-1..1,y=-2..2)`. Возможны разные системы координат. В полярной системе оператор `fieldplot([0,t],r=0..1,t=0..2*Pi,coords=polar)` отображает векторы по касательной к концентрическим окружностям (т. к. первая проекция равна нулю) с увеличивающейся вместе с углом t длиной. Для векторного поля в пространстве есть оператор `fieldplot3d`.
- `gradplot`. Этот оператор строит градиент скалярной функции на плоскости, `gradplot3d` — в пространстве.
- `implicitplot`. Оператор для построения графиков неявных функций. См. примеры на с. 480, 482. Улучшить качество графиков можно, увеличив параметры `numpoints` (по умолчанию 50) или `resolution` (по умолчанию 200). В тех случаях, когда неявная функция распадается на множители, иногда полезна опция `factor`. Причем предварительное разложение на множители на результат не влияет. Эта опция указывается в операторе `implicitplot` в виде `factor=true` или просто `factor` в конце оператора.
- `inequal`. Построение областей, ограниченных линейными функциями. Например, `inequal(2*x-y>0,x+y<=1,x=-2..2,y=-2..2)`. Можно указать цвет, которым будет залита область `optionsfeasible=(color=green)` и цвет вне области `optionsexcluded=(color=red)`.
- `interactive`. Команда для вызова маплета, помогающего строить график функции (рис. 344). Обращаться к команде надо с указанием функции, например `interactive(sin(x)/x)`. Маплет подскажет

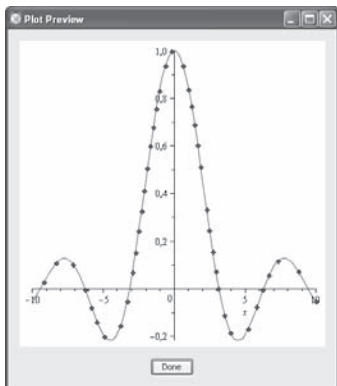


Рис. 344

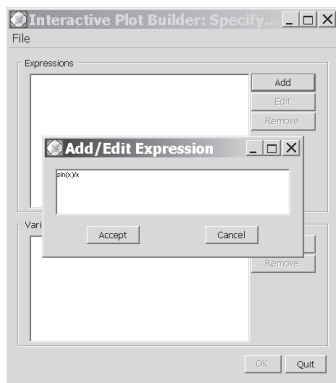


Рис. 345

опции, позволит отредактировать функцию. Результат будет выведен как и при обычной команде `plot` на рабочее поле программы или в отдельное окно, если нажать кнопку `Preview`. При обращении без аргумента `interactive()` надо вести функцию уже внутри маплета (рис. 345). Из графической оболочки **Standard Worksheet** это можно сделать из меню по цепочке `Tools->Assistants->Plot Builder`. При этом срабатывает команда `plots[interactive]()`. Эта команда отображается на экране в момент загрузки Plot Builder из меню, и наоборот: при выполнении команды загружается Plot Builder. В **Classic Worksheet** — это единственный способ вызвать программу построения графиков.

В Plot Builder есть шесть вариантов графиков: 2-D plot, 2-D polar plot, 3-D conformal plot of complex-valued functions, 2-D conformal plot of complex-valued functions, 2-D complex plot, 3-D complex plot. Нужный вариант следует выбрать в отдельном окне (рис. 346).

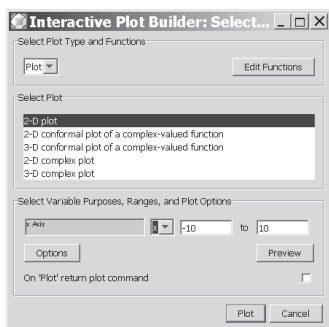


Рис. 346

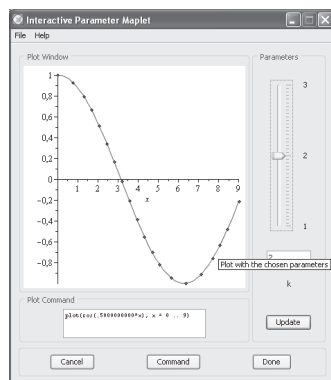


Рис. 347

Здесь же можно указать пределы изменения аргумента и всевозможные опции (кнопка `Options`) — толщина, цвет, тип линии, заголовок и др. Если функция имеет два аргумента, то программа предложит список из шести других вариантов: 3-D plot, 3-D contour plot, 2-D contour plot, 3-D gradient vector-field plot, 2-D density plot, 2-D implicit plot.

- `interactiveparams`. Построение графика с параметром. Команда, аналогичная `interactive`, без возможности указаний опций и изменения функции, но маплет снабжен ползунком, меняющим параметр графика (рис. 347). Например, в команде `interactiveparams(plot, [cos(x/k),`

$x=0..9], k=1..3)$ параметром является k . График выводится в окно маплета, но при нажатии кнопки Done график появится на рабочем поле программы.

- `intersectplot`. Изображение линии пересечения двух поверхностей. Пересечение кругового конуса $x^2 + y^2 = z^2$ и плоскости $x = 1$: `intersectplot(x^2+y^2=z^2, x=1, x=-1..3, y=-3..3, z=-3..3, axes=normal)` (гипербола). Если не указать тип осей, то по умолчанию оси не изображаются.
- `listplot`. Построение графика функции, принимающей значения в равностоящих точках $x = 1, 2, 3, \dots$. Так, оператор `listplot([1.5, -2, 4])` построит ломаную, соединяющую точки (1,1.5), (2,-2), (3,4). Для получения последовательностей с известным общим членом удобно применять оператор `seq`. Можно указывать и абсциссы, но тогда этот оператор будет полностью совпадать по результатам с оператором `plot`. Например, оператор `listplot([[0,1],[1,2],[2,1],[0,1]])` построит замкнутый треугольник с вершинами (0,1), (1,2), (2,1).
- `loglogplot`. Построение графика функции в двойных логарифмических координатах. Например, `loglogplot(x^2, x=1..99)` будет изображена прямая.
- `logplot`. График в полулогарифмических координатах. Десятичный логарифм числа x в Maple обозначен как `log10(x)`.
- `matrixplot`. Отображение значений элементов матрицы в осях x, y, z . Так, оператор `M:=Matrix([[10,20,22],[21.5,30,-0.1]])` задает матрицу, а оператор `matrixplot(M)` строит поверхность, натянутую на точки (1,1,10), (1,2,20), (1,3,22), (2,1,21.5), (2,2,30), (3,2,-0.1). По умолчанию оси координат не указаны, поэтому лучше задать опцию `axes=normal`. Возможно и прямое построение значений матрицы, без предварительного ее задания: `matrixplot([[1,20,2],[0,0,0.3]])`. К этому оператору применимы опции оператора `plot3d`¹. В частности, цветовое решение управляется параметром `lightmodel`, имеющим значения `light1, light2, light3, light4`. По умолчанию плоскости раскрашены в акварельные тона: `lightmodel=none`.
- `odeplot`. Построение графика численного решения обыкновенного дифференциального уравнения. Например, задано уравнение второго порядка

¹Справку по опциям этого оператора можно вызвать командой `?plot3d/option`.

```
> f:=y(x): eq:=f^2*diff(f,x$2)+
>      2*f*diff(f,x)^2+a*f+b*x*diff(f,x)=0;
```

Оператор решения имеет вид $Z:=dsolve(\{eq, y(0)=1, D(y)(0)=0\}, y(x), type=numeric)$. Здесь указаны два начальных условия — на функцию и производную и тип решения. Можно также указать область, в которой решение ищется: `range=0..1`. График решения при заданных параметрах a и b строит оператор `odeplot(Z, x=0..1)`. Производную в уравнении можно задать также операторным образом $D(D(y))(x)$. В операторе построения графика можно указать область `odeplot(p, x=0..1)`, но не шире той области, в которой решение ищется, иначе график будет построен, но загрублен. Если уравнение имеет особенность, то система может предупредить об этом. В данном уравнении¹, например, при $a = b = 1$ появится сообщение: *Warning, cannot evaluate the solution further right of .99999983, probably a singularity*. См. также примеры на с. 375, 384.

- `pareto`. Построение диаграммы Парето². Например: `pareto([1, 11, 3, 4], tags=['MN', 'VK', 'a4', 'T'])`. Аргументами оператора являются два списка, в первом — значения некоторой величины, во втором — наименование этой величины. Списки находятся во взаимно однозначном соответствии. Оператор сортирует значения и рисует прямоугольники, пропорциональные значениям. Рядом выстраивается кривая Парето. В механике оператор `pareto` вряд ли применим, его используют как своеобразную диаграмму в различных статистических и экономических исследованиях.
- `plotcompare`. Построение рядом графиков двух функций для сравнения. Например, `plotcompare(sin(x), sinh(x), x=-3..3)`. Особенно удобно для сравнений комплекснозначных функций. Для этого можно использовать опцию `same_box=true`.
- `pointplot`. Построение отдельных точек на плоскости. Оператор `pointplot([10,1], [2,-1.2])` построит две точки (10, 1) и (2, -1.2). Можно строить большие последовательности, например отображать результаты экспериментов.
- `pointplot3d`. Построение отдельных точек в пространстве.
- `polarplot`. Параметрически заданная кривая в полярных координатах. Логарифмическая спираль: `polarplot([exp(t), t, t=0..Pi])`. Первый аргумент $\rho(t)$, второй — $\varphi(t)$, t — параметр.

¹Уравнение, возникающее в задаче о растекании тонкого пластического слоя, сжимаемого жесткими поверхностями [26].

²*Pareto, Vilfredo* (1848–1923) — итальянский экономист.



- `polygonplot`. Построение полигона — замкнутой ломаной. Треугольник, например, выглядит так: `polygonplot([[0,0],[1,3],[0.1,10]])`.
- `polygonplot3d`. Построение ломаной поверхности, натянутой на заданные точки. Можно построить одновременно несколько несвязанных поверхностей. Например, серия параллельных прямоугольных треугольников: `polygonplot3d(seq([[0,0,i],[2,0,i],[2,2,i]],i=1..5))`. Треугольники отличаются координатой $z = i$, $i = 1..5$.
- `polyhedra_supported()`. Справочная команда, вызывающая типы известных в Maple многогранников для построения оператором `polyhedraplot`. Всего насчитывается 123 наименования, например `TriangularPrism` — треугольная призма, или `SquareAntiprism` — квадратная антипризма¹. Есть и такие малоизвестные фигуры, как `GyrateBidiminishedRhombicosidodecahedron`.
- `polyhedraplot`. Построение многогранников. Например, построение додекаэдра²: `polyhedraplot([0,0,0],polytype=dodecahedron)`. Параметр `polyscale` меняет размер фигуры, по умолчанию `polyscale=1`.
- `semilogplot`. Построение графика функции с логарифмической шкалой по оси x (см. также `logplot`, с. 275).
- `setcolors`. Оператор, назначающий цвета кривых по умолчанию, т. е. в тех случаях, когда на одном графике выводятся несколько кривых, а в операторе цвета кривых явно не указаны. Изначально по умолчанию даны следующие шесть цветов: `MediumOrchid`, `LimeGreen`, `Goldenrod`, `Blue`, `Red`, `DarkTurquoise`. Например, оператор `setcolors(["BlueViolet", "DarkRed", "Khaki"])` назначает три цвета, но если кривых будет больше, то эти цвета будут циклически повторяться. Полный список цветов можно найти по цепочке помощи `Graphics -> Plot Color Names`. Восстанавливает исходный список оператор `setcolors(default)`.
- `setoptions`. Оператор, назначающий опции графику вне оператора `plot`. Например, `setoptions(title='B1',axes=none,color=green)`. Этот же результат можно получить иначе:

¹В антипризме две параллельные грани являются одинаковыми правильными n -угольниками, а $2n$ боковые грани — правильными треугольниками.

²Правильный многогранник, составленный из двенадцати правильных пятиугольников.

```
A:=title='B1',axes=none,color=green: plot(x+2,x=0..1,A).
```

- `spacecurve`. Построение кривой в пространстве. Например, `spacecurve([cos(t), cos(2*t), sin(3*t)], t=0..4, axes=frame)`.
- `sparsematrixplot`. Отображает плотность заполнения матрицы. Ненулевые элементы отмечаются значком. На рисунке 348 приведен пример заполнения матрицы (5.6) на с. 251 (обозначим ее как **A**). `sparsematrixplot(A, matrixview, symbol=circle, symbolsize=20)`. Опция `matrixview` располагает рисунок соответственно матрице: ряды (`row`) — горизонтальные, столбцы (`column`) — вертикальные. При работе в **Maple Standard Worksheet** нумерация столбцов и строк выбрана обратная естественной, поэтому картина получается перевернутая.
- `surfdata`. Построение ломаной поверхности по точкам. Строятся ряды ломаных, а на них уже натягивается поверхность. Например, оператор `surfdata([[0,0,0],[1,0,0]],[[0,1,1],[1,2,0]])` построит сначала отрезок от точки (0,0,0) до точки (1,0,0), затем другой отрезок от (0,1,1) до (1,2,0), и на полученные отрезки натянет ломаную поверхность (рис. 349). Можно указать оси `axes=box` и другие опции оператора `plot3d`. Полный список опций вызывается командой `?plot3d/options`.

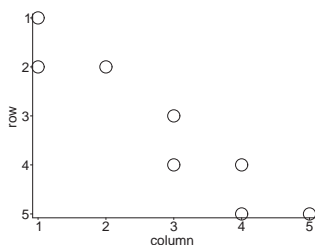


Рис. 348

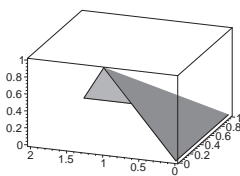


Рис. 349

- `textplot`. Вывод текста в нужную точку плоскости. В результате действия оператора `textplot([1.3,-2,"Moscow"], align=above,right)` на плоскость в точке $x = 1,3$, $y = -2$ будет помещен текст `Moscow`. Параметр `align` регулирует выравнивание текста относительно точки привязки. Возможны следующие значения: `above`, `below`, `right`, `left`. В различные точки можно поместить различные надписи и дать им свой шрифт и цвет:

```
A:=[1,3,"MPEI",font=[TIMES,ROMAN,30]]:
```

```
B:=[2,1,"MSU",align=above,color=red]:
textplot([A,B]);
```

- `textplot3d`. Вывод текста в нужную точку пространства. Работает аналогично `textplot`.
- `tubeplot`. Построение поверхности в тороидальной системе координат¹. Оператор `tubeplot([cos(t),sin(t),t],t=0..4*Pi,radius=0.4)` строит пустотелую спираль (рис. 350). Опция `numpoints` регулирует число полигонов поверхности вдоль тора, `tubepoints` — по окружности его нормального сечения. По умолчанию эти величины равны 50 и 10 соответственно.

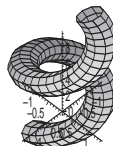


Рис. 350

5.5.5. Основные операторы пакета `plottools`

Операторы этого пакета воспроизводятся в операторе `display` из пакета `plots`.

- `arc`. Дуга окружности. Первый аргумент — центр окружности (список $[x,y]$ из координат), второй — радиус окружности (может быть и отрицательным), третий аргумент — пределы изменения угла. В конце можно добавить опции — цвет, тип линии, толщину. Пример: `display(arc([0,0],3,-Pi/2..Pi/2))`. Для отображения сектора (его можно закрасить) есть оператор `pieslice`, с. 282.
- `arrow`. Стрелка. Задается координатами начала, конца и тремя параметрами формы. Например, `display(arrow([0,10],[10,10],0.5,1.3,.1))` — изображен горизонтальный вектор от точки (0,10) до точки (10,10) толщиной 0.5, шириной наконечника 1.3 и длиной наконечника $10 \cdot 0.1 = 1$ (задается относительной величиной). Цвет можно указать как внутри оператора `arrow`, так и после него, т. е. внутри `display`. Если стрелка одна, то это безразлично, если несколько, то цвет вне `arrow` относится ко всем стрелкам, если в них цвет не указан. Описание `arrow` в пакете `plottools` отличается от аналогичного в `plots`, поэтому надо следить за последовательностью загрузки пакетов. Последний загруженный пакет имеет предпочтение.
- `circle`. Окружность. Первый аргумент — координаты центра, второй — радиус, третий (необязательный) — опции, например цвет.

¹Тороидальные координаты применяются в механике гироскопов [43].

Оператор `circle([0,0], 12, color=red)` изображает красную окружность радиусом 12 с центром в начале координат. По умолчанию радиус равен 1.

- `cone`. Конус. Необходимо задать координаты вершины. Радиус основания и высота по умолчанию равны 1, но их тоже можно задать. Так, конус `display(cone([0, 0, 2], 0.5, 2))` имеет высоту 2 (допустимы и отрицательные значения), радиус 0.5 и координаты вершины (0,0,2).
- `suboid`. Параллелепипед. Задается координатами концов большой диагонали. Например, `display(suboid([0,0,0], [1,1,1]), lightmodel=light2)`. Указана цветовая модель (с. 275).
- `curve`. Кривая, построенная по точкам. Например, ломаная `display(curve([[0,0],[3,4],[1,4]]))`. В конце можно указать опции: цвет, стиль, толщину.
- `cutin`. Оператор, закрашивающий часть заданной поверхности. Название происходит от двух действий — вырезать (`cut`) и вставить (`in`). Степень окраски — второй аргумент, равный 1, если окраска полная. Оператор `display(cutin(sphere(2), 3/5))` оставляет неокрашенной $2/5$ поверхности сферы радиуса 2. Закрашенные части образуют равномерную цветную мозаику. Раскрасить таким образом можно все пространственные поверхности из пакета `plottools`: `cylinder`, `suboid`, `cone`, `sphere`, `parallelepiped`, `torus`, `semitorus`.
- `cutout`. Оператор, в каком-то смысле обратный по результату `cutin`. Этот оператор вырезает ту часть поверхности, которую оставляет `cutin`. Например, `display(cutout(sphere(2), 3/5))`. Если этот результат совместить с аналогичным, полученным `cutin`, то получится полная сфера.
- `cylinder`. Цилиндр. Первый аргумент — координаты центра основания, второй — радиус, третий — высота. Например, `display(cylinder([1,0,2],1,4))`. Если указать опцию `capped = false`, то основания цилиндра не закрашиваются (труба).
- `disk`. Круг. Например, `display(disk([0,1],2.3), color=green)` — зеленый круг радиусом 2.3 с центром в точке (0,1). Если не указать цвет, то получится окружность.
- `dodecahedron`. Додекаэдр¹ (двенадцатигранник). Например, `display(dodecahedron([0,1,1],1.2))`, рис. 351. Координаты

¹См. также с. 277.

центра $(0,1,1)$ и размер 1.2 — необязательные аргументы, по умолчанию центр — начало координат, а размер равен 1.

- **ellipse**. Эллипс. Указывается центр и полуоси. Например, `display(ellipse([0,0], 3, 1), scaling=constrained)`. Эллипс можно закрасить, если внутри оператора `ellipse` указать опции `filled=true, color=blue`. Если не задать тип масштабирования `constrained`, то оси растянутся и внешне любой эллипс будет похож на окружность.
- **ellipticArc**. Дуга эллипса. Параметры те же, что и у эллипса, и еще два — начальный и конечный угол. Пример: `ellipticArc([0,0],3, 1, 0..Pi)` — верхняя половина эллипса. Возможна окраска части эллипса, ограниченной дугой и хордой, стягивающей концы дуги.

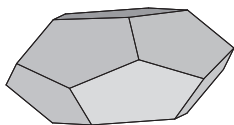


Рис. 351

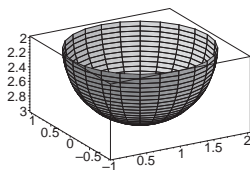


Рис. 352

- **hemisphere**. Полусфера. Имеются два необязательные аргумента — координаты центра и радиус. Перевернутая полусфера соответствует отрицательному радиусу. Например, на рис. 352 изображена перевернутая полусфера радиуса 1 с центром в точке $(0,1,2)$ без дна (`capped=false`): `display(hemisphere([0,1,2], -1, capped=false), axes=boxed, scaling=constrained, orientation = [30,-110]);`
- **hexahedron**. Куб. Аргументы: координаты центра и длина половины ребра. Пример: `display(hexahedron([0,0,0], 0.5, color=blue))`.
- **homothety**. Равномерное растяжение (сжатие) геометрического объекта по всем осям (преобразование подобия). Например, если задать куб `Kub:=hexahedron([0,0,0],1)`, то оператор `display(homothety(Kub,2,[10,2,3]))` растянет его в 2 раза по всем осям и поместит центр в точку $(-10,-2,-3)$. Второй аргумент можно не указывать, по умолчанию он принят $[0,0,0]$, т. е. центр фигуры остается на месте. Частный случай оператора `scale`.

- **hyperbola**. Гипербола. Оператор `display(hyperbola([x1,y1], a, b,-1..1))` строит параметрически заданную гиперболу $x = x_1 \pm a \operatorname{ch}(t)$, $y = y_1 \pm b \operatorname{sh}(t)$, $t = -1 \dots 1$. Здесь x_1, y_1 — центр гиперболы, a, b — ее полуоси; знак плюс соответствует правой ветви, минус — левой.
- **icosahedron**. Икосаэдр (правильный выпуклый многогранник, двадцатигранник, грани — равносторонние треугольники). Строится оператором `display` так же, как куб и додекаэдр: `icosahedron([0,0,0],2)` (рис. 353).
- **line**. Отрезок. Задаются координаты начала и конца: `display(line([0,0], [3,4]))`. Можно строить и в пространстве, указывать цвет (`color`), тип (`linestyle`) и толщину (`thickness`).
- **octahedron**. Октаэдр¹. Строится так же, как куб, додекаэдр и икосаэдр: `display(octahedron([0,0,0],0.8))`.
- **parallelepiped**. Параллелепипед, построенный на трех заданных векторах. Пример: `display(parallelepiped([0.5,0.2,2], [0,1,0], [1,-1,0], [0,0,0]))`. Первые три аргумента — векторы, последний аргумент необязательный. Это координаты точки, от которой вектора отложены (по умолчанию от начала координат).
- **pieslice**. Сектор круга: `display(pieslice([1,0],7,-Pi..Pi/4, color=red))`. Здесь $[1,0]$ — координаты центра, 7 — радиус. Последний аргумент — диапазон изменения угла. Если указать цвет, то фигура заливается цветом, иначе она будет прозрачной.
- **point**. Точка: `display(point([0,0.7,0], color=blue, symbol=box, symbolsize=50))`. Здесь $[0,0.7,0]$ — координаты точки. Указан цвет, вид и размер точки. Вид бывает также `circle`, `diamond`, `cross`. Точку можно изображать и на плоскости.
- **polygon**. Замкнутая ломаная. Указываются координаты вершин: `display(polygon([[0,0], [3,4], [3,1]]))`. Если указать цвет, то полигон заливается этим цветом.

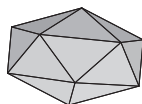


Рис. 353

¹Правильный многогранник, составленный из восьми равносторонних треугольников.

- **project**. Проекция объекта в осях x, y на прямую и в осях x, y, z на плоскость. Прямая задается двумя точками, через которые она проходит, например $[[0,0],[1,1]]$, плоскость — тремя точками. Например, задан икосаэдр $A:=\text{icosahedron}([0,0,10],2)$. Проецируем его на плоскость x, y : `display(project(A, [[0,0,0],[1,0,0],[0,1,0]])`). При этом проекция похожа на сплюснутый оригинал — на плоскости отображается не только контур фигуры, как тень при вертикальном освещении, но и все ребра и грани, если таковые есть, внутри контура. Цветовая палитра проекции и оригинала различны.
- **rectangle**. Прямоугольник. Задается координатами верхнего левого угла и нижнего правого: `display(rectangle([0,3],[4,0], color=red), scaling=constrained)`.
- **reflect**. Операция отражения фигуры относительно прямой или плоскости. Например, построим параболоид вращения $a:=\text{plot3d}(x^2+y^2, x=-4..4, y=-4..4)$. Отразим его относительно плоскости $z = 34$, заданной тремя точками: $b:=\text{reflect}(a, [[0,0,34],[1,0,34],[0,1,34]])$. Построим изображения фигуры и ее отражения: `display(a,b)` (рис. 354).

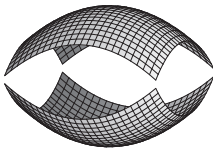


Рис. 354

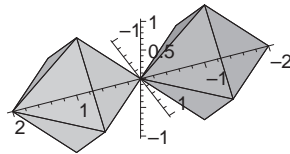


Рис. 355

- **rotate**. Операция поворота. Задается объект и три угла поворота (в радианах) относительно осей x, y, z . Изображаем, например, октаэдр высотой 2 (по умолчанию) с центром в точке $(1,0,0)$, т. е. смещенным по оси x на 1: $a := \text{octahedron}([1,0,0])$. Поворачиваем его на 180° вокруг оси y : $b := \text{rotate}(a, 0, \text{Pi}, 0)$. Получаем результат: `display(a,b, axes=normal)`, (рис. 355). Заметим, что из-за симметрии фигуры этот же результат получится, если сделать поворот вокруг оси z .

Следующая короткая программа показывает процесс вращения окружности радиуса 1 с центром в точке $(2,0)$ вокруг начала координат. В цикле по i создается N последовательность кадров с повернутыми окружностями.

Оператор `display` выводит их на экран в виде простейшего мультфильма. Опция `insequence=true` обеспечивает стирание

одного кадра и вывод следующего. Если `insequence=false` (по умолчанию), то анимации не будет, и на экране появятся сразу все окружности (рис. 356).

```
> N:=10:
> for i to N do
>   b[i]:=rotate(circle([2,0]),2*Pi/N*i):
> od:
> display(seq(b[i],i=1..N),insequence=true);
```

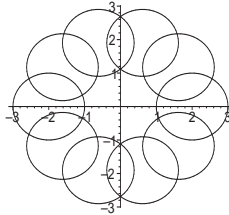


Рис. 356

- **scale.** Операция растяжения (сжатия) объекта. Окружность на плоскости можно превратить в эллипс с полуосями 2 и 1: `display(scale(circle(),2,1), scaling=constrained)`. Аналогично для трехмерных объектов надо ввести три коэффициента растяжения (сжатия). В частном случае, когда по всем осям коэффициенты равны, работает оператор `homothety`, с. 281.
- **semitorus.** Часть тора. Указывается центр (три координаты), интервал центрального угла и два необязательных аргумента: радиус образующей окружности (по умолчанию 1) и радиус вращения этой окружности — радиус тора (по умолчанию равен 2). Пример: `display(semitorus([0,0,0],0..7*Pi/4,2,5))`. Полный тор соответствует углу $0..2*Pi/4$.
- **sphere.** Сфера. Указывается центр и радиус: `display(sphere([0,1,0],5))`. По умолчанию центр — начало координат, радиус равен 1.
- **stellate.** Декоративная операция с пространственными объектами. Каждая плоская грань (в том числе и у сферы, так как сфера изображается в виде своеобразного многогранника) заменяется пирамидой с основанием на грани. Высота пирамиды определяется параметром операции (по умолчанию 2). Если параметр меньше 1, то пирамида строится внутрь поверхности. На рис. 357 изображена полусфера, обработанная оператором `display(stellate(hemisphere(capped=false),1.2))`.

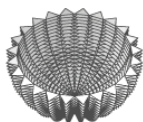


Рис. 357

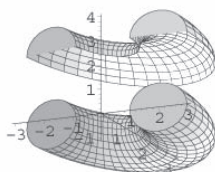


Рис. 358

- **torus**. Тор. Указывается центр (три координаты, по умолчанию — начало координат), два размера: радиус образующей окружности (по умолчанию 1) и радиус вращения этой окружности — радиус тора (по умолчанию равен 2). Пример: `display(torus([1,1,1],1,2))`.

- **transform**. Универсальное преобразование фигуры. Включает в себя и сдвиг, и поворот, и растяжение. В следующем примере показано, как срезать половину тора и сдвинуть его по оси z (рис. 358):

```
> g := semitorus([0,0,0],0..3)
> F := transform((x, y, z) ->[x,y,abs(z)+3]):
> display(g,F(g),axes=normal);
```

Для сдвига по оси x требуется функция $(x, y, z) \rightarrow [x+1, y, z]$ для растяжения — $(x, y, z) \rightarrow [x*2, y*3, -z*2]$ и т. д. Интересно поэкспериментировать с нелинейными функциями от известных объектов (сфера, куб и пр.), результаты иногда получаются самыми неожиданными.

- **tetrahedron**. Тетраэдр. Два необязательных аргумента — координаты центра (по умолчанию начало координат) и размер (по умолчанию 1). Например, тетраэдр `display(tetrahedron([0,0,0],1))` имеет размер ребра $2\sqrt{2}$.
- **translate**. Операция параллельного переноса. Для объекта на плоскости надо указать два числа, в пространстве — три: `display([translate(sphere([0,1,0],5),0,0,3)], axes=normal)`.
- **vrm1**. Конвертация графического объекта в формат vrm1 [1]. Самый простой пример:

```
> vrm1(plot([[0,0],[1,1]]), 'C:\plot.wrl')
```

В результате получим файл `plot.wrl` изображения отрезка прямой с концами в точках $(0,0)$ и $(1,1)$. Смотреть файл можно с помощью

Web-браузера, на который установлен специальный плагин, например, бесплатный [cortona](http://www.cortona3d.com)¹. Содержимое текстового файла `plot.wrl`:

```
#VRML V1.0 ascii
ShapeHints {
  vertexOrdering COUNTERCLOCKWISE
  shapeType      UNKNOWN_SHAPE_TYPE
}
MaterialBinding {
  value PER_FACE
}
DEF BackgroundColor Info {
string "0.000000 0.000000 0.000000"
}
DEF SceneInfo Info {
string "info@maplesoft.com"
}
Coordinate3 {
  point [
    0.000000  0.000000  0.000000,
    1.000000  1.000000  0.000000
  ]
}
Material {
  diffuseColor [
    1.000000  0.000000  0.000000
  ]
  emissiveColor 0.000000  0.000000  0.000000
  transparency 0.000000
  ambientColor 0.200000  0.200000  0.200000
  specularColor 0.000000  0.000000  0.000000
  shininess    0.200000
}
IndexedLineSet {
  coordIndex [
    0, 1
  ]
}
```

¹<http://www.cortona3d.com>

Более эффектно изображение трехмерных объектов, например поверхности в тороидальной системе координат. Изображение выводим в формате `vrml` (рис. 359):

```
vrml(tubeplot([cos(t), sin(2*t), 0, t=Pi..2*Pi, radius
=0.250*(t-Pi)], numpoints = 100), 'C:\\plot3.wrl',
background_color = COLOR(RGB,1,1,1)).
```



Рис. 359

Параметры просмотра рисунка в браузере можно отрегулировать, выбрав правой кнопкой мыши опцию "Параметры" (рис. 360). Помимо изменения цвета фона (обычно он темный), здесь есть возможность указать один из трех типов Рендерера: OpenGL, DirectX или R98. Можно изменить частоту кадров и тип оформления.

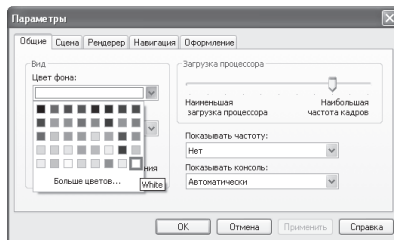


Рис. 360

5.5.6. Основные опции оператора plot3d

Самый распространенный вызов оператора построения трехмерных графиков имеет вид `plot3d(expr, x=a..b, y=c..d, opts)`, где `expr` — функция, зависящая от `x` и `y`, `a, b` и `c, d` — пределы изменения аргументов, `opts` — дополнительные опции. Другой способ — параметрический, имеет следующую форму: `plot3d([f, g, h], s=a..b, t=c..d, opts)`, где `f, g, h` — функции, зависящие от `s` и `t`. Опции `axes`, `axesfont`, `caption`, `color`, `caption`, `filled`, `font`, `labels`, `labelfont`, `linestyle`, `scaling`, `thickness`, `title`,

`titlefont` и `view` совпадают с аналогичными опциями оператора `plot`, с. 265. Другие опции:

- `ambientlight`=[*r*, *g*, *b*]. Установка интенсивности красного, зеленого и синего цвета (red, green, blue) внешнего освещения. Если приняты параметры [1,1,1], то собственный цвет объекта, заданный опцией `color`, сохраняется. Если одно из значений *r*, *g*, *b* равно 0 или близко к 0, то фигура будет черной. Опция `lightmodel` отменяет раскраску `ambientlight`.
- `contours`. Если поверхность строится в виде контуров (опция `style=contour`), то этот параметр задает их число. Можно даже оставить только один контур. Изображение поверхностей по их контурам позволяет выделить их особенности. На рис. 361 и 362 показана одна и та же поверхность (параболический гиперболоид, седло), построенная обычным образом: `plot3d(x^2-y^2, x=-3..3, y=-3..3, axes=NORMAL)` и с помощью контуров: `plot3d(x^2-y^2, x=-3..3, y=-3..3, axes=NORMAL, style=contour, contours=21)`.

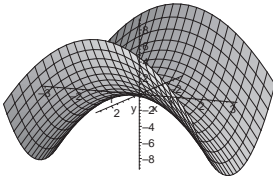


Рис. 361

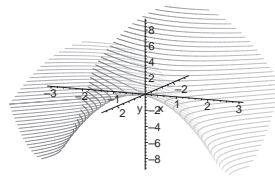


Рис. 362

- `coords`. Параметр, задающий систему координат. По умолчанию это декартова система (Cartesian). Есть также системы:

`bipolarcylindrical`, `bispherical`¹, `cardioid`, `cardioidcylindrical`, `casscylindrical`, `confocalellip`, `confocalparab`, `conical`, `cylindrical`, `ellcylindrical`, `ellipsoidal`, `hypercylindrical`, `invcasscylindrical`, `invellcylindrical`, `invoblspheroidal`, `invproospheroidal`, `logcoshcylindrical`, `logcylindrical`, `maxwellcylindrical`, `oblatespheroidal`, `paraboloidal`, `paracylindrical`, `prolatespheroidal`, `rosecylindrical`, `toroida`, `sixsphere`, `spherical`, `tangentcylindrical`, `tangentsphere`.

С помощью `addcoords` опции можно добавить свою систему: `addcoords(Zcyl, [z, r, phi], [r*cos(phi), r*sin(phi), z])`.

- `filled`. Если `filled=true`, то пространство между поверхностью и плоскостью $x-y$ окрашивается (заполняется): `plot3d(x^2+y^2, x=-3..3, y=-3..3, axes=NORMAL, filled=true)`

¹Бисферическая система координат используются в задаче о движении шара в электростатическом подвесе [40].

- **filledregions**. Условие для окраски полос между линиями уровня, если такая опция (`style=contour` в `plot3d`) включена. По умолчанию `filledregions=false`.
- **glossiness**. Глянец на поверхности. Параметр, принимающий значение от 0 (матовая поверхность) до 1 (глянцевая поверхность). В классической версии `swmaple.exe` этот параметр не работает (наоборот, если `glossiness=0`, то поверхность становится просто немного ярче, но отблики при `glossiness=1` не возникают).
- **labeldirections**. Направление текста меток осей. По умолчанию `labeldirections=[horizontal, horizontal, horizontal]`. Другой вариант — `vertical` (текст вдоль оси). Имеет смысл, если указаны наименования осей, например, `labels=["X", "Y", "Z1"]`.
- **light**. Направление и цвет освещения. Первые два параметра — углы (в градусах), три остальные — интенсивность (от 0 до 1) красного, зеленого и синего цветов (r,g,b). Например, при `light=[180,0, 1, 0, 0]` объект будет освещен снизу экрана красным цветом. Поворот осей мышью не меняет освещения, т. е. оси могут менять относительно экрана направление, а освещение остается прежним. Не работает вместе с опцией `lightmodel`.
- **numpoints**. Число точек, из которых состоит объект. По умолчанию `numpoints=625`.
- **orientation**. Направление осей относительно наблюдателя. Измеряется двумя углами в градусах. По умолчанию `orientation=[45,45]`. Пример см. на рис. 365, с. 293. Если в меню View интерфейса программы активизировать `Context Bar` и мышью выбрать нужный рисунок, то, меняя ориентацию осей, на панели можно подобрать нужное значение углов и занести их потом в параметр `orientation`. Эти же углы ориентации ϑ и φ видны на рис. 412, с. 354.
- **projection**. Вид проекции. По умолчанию проекция ортогональная: `projection = orthogonal`. Другие варианты: `normal` и `fisheye` ("рыбий глаз", или короткофокусный объектив). Этим константам соответствуют значения 1, 0,5 и 0.
- **shading**. Вид раскраски. Возможны следующие значения этого параметра: `xyz`, `xy`, `z`, `zgrayscale`, `zhue`, `none`.
- **style**. Стиль поверхности. По умолчанию `style=patch` — раскрашенная поверхность, покрытая сеткой. Возможны следующие варианты:

- `surface` или `patchnograd`. Гладкая поверхность. Сетка для наглядного отображения формы поверхности не отображается.
 - `surfacewireframe` ("веревочная сетка"). То же, что и `patch`.
 - `contour`. Контурное представление поверхности. Сетка на поверхности — это два ортогональных семейства кривых, а контур (`contour`) — одно из семейств. Для сферы это параллели.
 - `surfacecontour` или `patchcontour`. То же, что и `contour`, но с раскраской поверхности.
 - `wireframe` или `line`. Отображается только сетка на поверхности.
 - `wireframeopaque` или `hidden`. Отображается только сетка на поверхности, но сам объект непрозрачен, т. е. видна только одна сторона объекта.
 - `point`. Точки на поверхности (узлы сетки). Наиболее прозрачный вариант отображения тела.
- `symbol`. Вид символа. Варианты: `asterisk`, `box`, `circle`, `cross`, `diagonalcross`, `point`, `solidsphere`, `sphere`. Реально отображаются только `box`, `cross`, `diamond` и `circle`, причем `circle` больше похож на `diamond`. Пример: `plot3d([[2,3,4]], x=0..5, y=0..3, style=point, symbol=box, symbolsize=40)`. В программах `swmaple.exe` и `maplew.exe` изображения точек несколько отличаются.
 - `symbolsize`. Размер символа. По умолчанию `symbolsize=10`.
 - `tickmarks`. Параметр, указывающий число или положение подписанных отметок на осях. Например, запись `tickmarks=[10,4,5]` означает, что по осям x , y и z будет подписано, соответственно, 10, 4 и 5 меток. В действительности это выполняется приближенно, реальное число подписанных отметок может немного отличаться. В следующем примере (рис. 363) по оси x разметка выбирается автоматически `default` и зависит от длины оси; вдоль оси y подписаны значения 1, 2, 3, а вдоль оси z — вместо чисел в указанные места поставлены буквы А, В, С: `plot3d(exp(x-y), x=0..3, y=0..Pi, tickmarks=[default, [1,2,3],[2="А",8="В",19="С"]], style=surface, axes=boxed)`.
 - `transparency`. Степень прозрачности объекта (от 0 до 1). По умолчанию объект непрозрачный: `transparency=0`. Опция работает в программе `maplew.exe`.

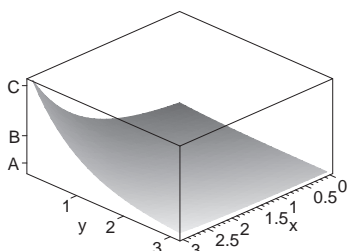


Рис. 363

5.5.7. Графические структуры

Простые графические объекты отображают операторы `PLOT` и `PLOT3D`. Таких объектов всего четыре: отрезки (`CURVES`), точки (`POINTS`), полигоны (`POLYGONS`) и текст (`TEXT`). Дополнительные пакеты при этом подгружать не требуется. Приведем примеры использования графических структур¹.

- `PLOT(CURVES([[1, 0], [1, 1]], LINESSTYLE(3)))`. Пунктирная линия, соединяющая точки (1,0) и (1,1). Параметр опции `LINESSTYLE` имеет значения от 1 до 7 и определяет стиль линии: `solid`, `dash`, `dot`, `dash-dot`, `long dash`, `space-dot` и `space-dash`². В одном операторе `CURVES` можно записать несколько отрезков, указав для каждого цвет. В следующем операторе указан вертикальный красный отрезок и горизонтальный синий:

```
> PLOT(CURVES([[1, 0], [1, 1]], [[0, 1], [1, 1]],
>           COLOR(RGB, 1, 0, 0, 0, 0, 1)));
```

Эти же цвета можно получить, используя режим цвета `HUE`: `COLOR(HUE, 1, 0.7)`. Здесь два отрезка, поэтому в режиме `HUE` две константы. Красному цвету (первый отрезок) соответствует 1, синему — 0.7. Можно выводить на экран и большее число отрезков, используя оператор создания последовательностей `seq`.

- `PLOT(POLYGONS([[1, 1], [1, 3], [3, 1]], THICKNESS(1)))`. Замкнутая ломаная (полигон), соединяющая точки с указанными координатами. В данном случае это равнобедренный прямоугольный треугольник. Задана необязательная опция — толщина линии. По умолчанию толщина задается самая малая, что соответствует

¹См. также примеры на с. 346, 356, 382, 404, 416, 445.

²Вариант с обозначением констант `LINESSTYLE(DASH)` и др. работает только в **Standard Worksheet**, в **Classic Worksheet** надо указывать значения констант, например `LINESSTYLE(3)`.

значению 0. Параметр толщины вычисляется по модулю 256, т. е. самая толстая линия задана значением 255. Цвет линии определяется так же, как и для CURVES.

- `PLOT(POINTS([1,2],SYMBOL(DIAMOND,100)))`. Изображение символа заданного размера (здесь 100pt) в заданной точке. Возможны еще такие символы: BOX, CROSS, CIRCLE, POINT.
- `PLOT(PLOT(ТЕХТ([1,0], "Москва", FONT(TIMES, BOLD, 8)))`. Надпись в точке с указанными координатами. Задан шрифт и размер шрифта.

5.5.8. Пример 3-D анимации в Maple

Рассмотрим задачу кинематики качения цилиндра по неподвижной цилиндрической поверхности — цилиндру большего радиуса (рис. 364). Система имеет одну степень свободы. Движение этой системы

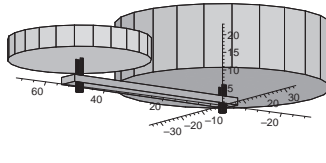


Рис. 364

можно изобразить и на плоском чертеже, но пространственная модель будет нагляднее, тем более что графикой в Maple легко управлять мышью, поворачивая объект относительно осей.

Система состоит из двух цилиндров радиуса R_1 и R_2 , водила (параллелепипеда) и двух осей (цилиндров). Оператором PLOT3D цилиндры группируются со своими осями в объекты Колесо1 и Колесо2. Здесь этот оператор фактически выполняет роль операторных скобок. Первоначально объекты создаются неподвижными. Все цилиндры размещены в начале координат. Первые два числа в списках координат это координаты x и y . Оси опущены на две единицы ниже плоскости xy , на что указывает третье число в списке $[0,0,-2]$, а цилиндры приподняты: $[0,0,5]$, $[0,0,7]$. Водило (параллелепипед) скользит по плоскости xy и имеет высоту по оси z равную 2. У параллелепипеда указаны координаты концов большой диагонали (рис. 365, 366). Далее в цикле по i формируем отдельные кадры анимации. Число кадров K определяет скорость и плавность движения. Объект Колесо1 неподвижен и сразу помещается первым аргументом оператора display в кадр $P[i]$. Водило и Колесо2 поворачиваются, поэтому к ним применяем оператор

поворота `rotate`. При этом формируются новые объекты P1 и P2. Углы поворота рассчитываются по простой формуле, следующей из того, что точка касания цилиндров является мгновенным центром скоростей для подвижного цилиндра: $\varphi_1 = \varphi(R_1 + R_2)/R_2$. Оператор `translate` смещает ось объекта P1 (подвижного колеса) в конец вращающегося валида. Координаты этой точки легко вычисляются: $x = OA \cos \varphi$, $y = OA \sin \varphi$. Задано произвольное колебательное движение: $\varphi = 0,3 \sin(2\pi i/K)$. Амплитуда колебаний равна 0,3.

Все кадры движения собираются в последовательность с помощью оператора `seq` и выводятся в промежуточную переменную P. Затем вся последовательность выводится на экран оператором `display`. При этом устанавливается режим замещения кадров (один кадр стирается, другой выводится) `insequence=true`.

Таким образом, в программе дважды используется оператор `display`. Ориентация осей определена параметром `orientation`.

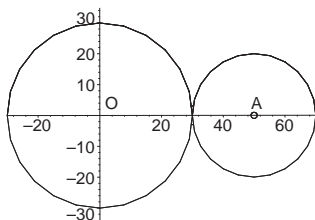


Рис. 365

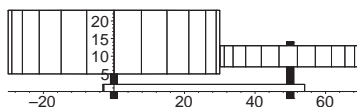


Рис. 366

Так, рисунок 365 построен оператором `display` при `orientation = [-90, 0]` а рисунок 366 — при `orientation = [-90, 90]`. Кроме того, выбран цвет заливки `color=white`.

Программа 2 (P2.mws)

```
> restart:
> with(plots):
> with(plottools):
> K:=48: #Количество кадров
> R1:=30: R2:=20: OA:=R1+R2:#Размеры
> Водило:=PLOT3D(cuboid([-3,-3,0],[OA+4,3,2])):
> Колесо1:=PLOT3D(cylinder([0,0,5],R1,18),
>                  cylinder([0,0,-2],1,16)):
> Колесо2:=PLOT3D(cylinder([0,0,7],R2,6),
>                  cylinder([0,0,-2],1,16)):
> for i from 0 to K do
>   Амплитуда:=0.3:
```

```

> phi:=sin(6.28*i/K)*Амплитуда:
> phi1:=phi*(R1+R2)/R2:
> P1:=rotate(Колесо2,0,0,-phi):
> P2:=rotate(Водило,0,0,-phi):
> x:= OA*cos(phi): y:= OA*sin(phi):
> P[i]:=display(Колесо1,translate(P1,x,y,0),P2);
> od:
  Изображение механизма в движении
> display(seq(P[i],i=0..K),orientation=[55,101],
> insequence=true,scaling=constrained,axes=normal);

```

5.6. Дифференциальные уравнения

В системе **Maple** можно находить как аналитические, так и численные решения обыкновенных дифференциальных уравнений и уравнений в частных производных, решать начальные и краевые задачи.

Производная в **Maple** обычно вводится оператором `diff`. Так, выражение `diff(y(x),x)` означает y'_x , а `diff(u(x,y,z),x$3,y,y,z)` — частную производную

$$\frac{\partial^6 u}{\partial x^3 \partial y^2 \partial z}. \quad (5.11)$$

Это же выражение можно ввести оператором

$$D[1,1,1,2,2,3](u)(x,y,z),$$

или, используя знак повторения `$`, `D[1$3,2$2,3](u)(x,y,z)`. При желании конвертацией `convert(%,diff)` можно вернуться к более привычному виду (5.11).

Еще один вариант записи производной: `diff(y(x),'$(x,2))`. Здесь имеется ввиду вторая производная по x .

Обыкновенное дифференциальное уравнение решают с помощью оператора `dsolve`. При этом никаких дополнительных пакетов подгружать не требуется, кроме, может быть, пакета для решения уравнений в частных производных `with(PDEtools)`. В этом пакете есть оператор `declare(y(x))` для улучшения вида полученных выражений. Функции $y(x)$ будут записываться просто как y , а производная `diff(y(x),x)` будет выглядеть на выводе результатов как y_x , а не $\frac{d}{dx}y(x)$. Это же относится и к старшим производным. Для того чтобы производная по x обозначалась штрихом, следует определить: `declare(y(x),prime=x)`. Для сложных уравнений компактный вид записи часто является существенным требованием (с. 383).

5.6.1. Оператор dsolve

Оператор `dsolve` может решать уравнения и системы уравнений, искать точные решения и численные. Можно ставить начальную, крайнюю задачу или искать общее решение.

Приведем примеры решений. Имеем дифференциальное уравнение первого порядка

```
> eq := diff(y(x), x, x) = 4*y(x) - x;
```

$$eq := \frac{d^2}{dx^2} y(x) = 4y(x) - x.$$

Найдем решение. Здесь в уравнении единственная функция, поэтому можно дополнительно не указывать, какую функцию ищем:

```
> dsolve(eq);
```

Получим решение $y(x) = e^{(-2x)}_C2 + e^{2x}_C1 + \frac{x}{4}$.

Если указывать неизвестную, то она записывается после уравнения: `dsolve(eq, y(x))`. Зададим начальные значения:

```
> dsolve({eq, D(y)(0)=0, y(0)=1/8}, y(x));
```

Производная записывается в операторном виде $D(y)$. Для уравнений более высокого порядка вторую производную в начальных условиях можно записать в виде $(D@@2)(y)(0)=b$ или $D(D(y))(0)=b$. Получим решение $y(x) = \frac{1}{8}e^{-2x} + \frac{x}{4}$. Решим это же уравнение в рядах¹. Ограничимся тремя членами ряда:

```
> Order:=3:
```

```
> dsolve({eq, D(y)(0)=0, y(0)=1/8}, y(x), series);
```

$$y(x) = \frac{1}{8} + \frac{1}{4}x^2 + O(x^3).$$

Для численного (`numeric`) решения существует много вариантов. Во-первых, можно задать метод решения (`rkf45`, `rosenbrock`, `bvp`, `rkf45_dae`, `rosenbrock_dae`, `dverk78`, `lsode`, `gear`, `taylorseries`, `mebdfi`, `classical`), во-вторых, существует несколько видов вывода (`output`) результатов. По умолчанию принимается метод Рунге–Кутты² `rkf45`.

```
> R:=dsolve({eq, D(y)(0)=0, y(0)=1}, y(x),
```

```
> numeric, output=operator);
```

Примеры численного решения дифференциальных уравнений см. на с. 375, 384. Еще один удобный способ решения дифференциальных уравнений в **Maple** — вызов интерактивного варианта оператора

```
> dsolve[interactive](diff(y(x), x)+4*y(x)+x);
```

¹См. пример на с. 382.

²*Carl David Tolme Runge* (1856–1927), *Martin Wilhelm Kutta* (1867–1944) — немецкие математики.

Из графической оболочки **Standard Worksheet** этот оператор доступен из меню по цепочке Tools->Assistants->ODE Analyzer. Открывается диалоговое окно, рис. 367:

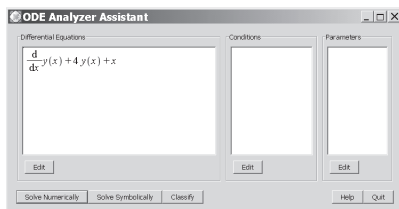


Рис. 367

Здесь можно проанализировать уравнение (фактически работает оператор `odeadvisor`, с. 302), решить аналитически или численно, отредактировать уравнение и начальные условия (рис. 368). Есть возможность выбора метода решения (результаты при этом могут отличаться в форме записи), времени, которое дается программе на «размышление», построить график.

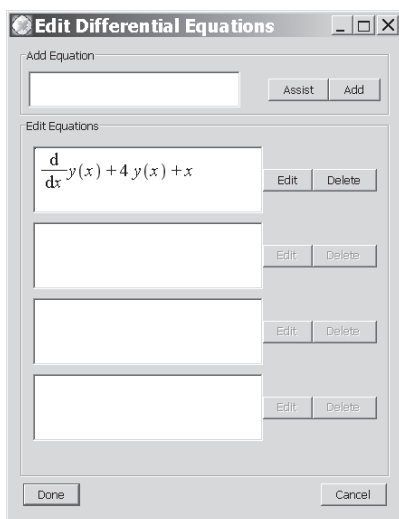


Рис. 368

Если вызвать оператор без аргумента `dsolve[interactive]()`, то сначала надо ввести уравнение, пользуясь удобной кнопкой `Assist` для записи производных или записывая производные вручную.

5.6.2. Пакет DEtools для решения обыкновенных дифференциальных уравнений

Опишем некоторые операторы пакета.

- `abelsol(eq, y(x))`. Решение уравнения Абеля¹

$$y' = f_0(x) + f_1(x)y + f_2(x)y^2 + f_3(x)y^3.$$

Заметим, что обычный оператор `dsolve` иногда справляется с задачей успешней.

- `autonomous(EQ, y, t)`. Проверка автономности дифференциального уравнения `EQ` для функции $y(t)$. В последнем аргументе указана независимая переменная, в данном случае `t`. Оператор работает и с системой уравнений, например `EQ:=diff(y(t), t)=x(t), diff(x(t), t$2)=y(t)+x(t)`. Система и функции заключаются при этом в фигурные скобки `autonomous({EQ}, {y, x}, t)`. Результат, возвращаемый оператором, `false` или `true`. В приведенном примере независимая переменная `t` не входит в уравнения, результатом будет `true`, т. е. система уравнений автономная.
- `bernoullisol(eq, y(x))`. Решение уравнения Бернулли²

$$y' + p(x)y = p(x)y^n.$$

- `chinisol(eq, y(x))`. Решение уравнения Чини для функции $y(x)$:

$$y' = f(x)y^n - g(x)y + h(x).$$

- `convertAlg(eq, y(x))`. Выделение коэффициентов из линейного дифференциального уравнения. Если применить оператор `convertAlg(eq, y(x))` к уравнению

$$2xy(x) + x \left(\frac{d}{dx} y(x) \right) + \left(\frac{d^2}{dx^2} y(x) \right) = 0, \quad (5.12)$$

то получим `[[2x, x, 1], 0]`, т. е. оператор действует аналогично `dcoeffs(eq, y(x))`, но применяется непосредственно к уравнению и выделяет также правую часть.

- `convertsys`. Оператор, приводящий систему уравнений к системе уравнений первого порядка. Пример. Имеем систему двух дифференциальных уравнений для функций $y(x), z(x)$:

$$\begin{aligned} y' &= y + zx, \\ z' &= y + x. \end{aligned} \quad (5.13)$$

¹ *Abel Niels Henrik* (1802–1827) — норвежский математик.

² *Daniel Bernoulli* (1700–1782) — швейцарский математик.

Зададим начальные условия $y(0) = 1$, $z(0) = 2$. К системе, которую запишем в виде

$$\text{ode} := \text{diff}(y(x), x) = y(x) + z(x) * x, \text{diff}(z(x), x) = x + y(x),$$

применим оператор, в котором первый аргумент — система уравнений, второй — список начальных условий (может быть пустым, если они не задаются), затем идет список переменных и независимая переменная x . Последний аргумент U (необязательный) — обозначение для вектора переменных:

`convertsys({ode}, {y(0)=1, z(0)=2}, {y(x), z(x)}, x, U).`

Получим результат, представленный списком

$$[[YP_1 = U_1 + U_2 x, YP_2 = x + U_1], [U_1 = y(x), U_2 = z(x)], 0, [1, 2]].$$

Сначала в квадратных скобках записана искомая система первого порядка. Под YP_1 понимается производная U_1' , YP_2 — это U_2' . Потом идет список новых обозначений для переменных как элементов введенного вектора U , начальное значение независимой переменной и начальные условия. Еще один пример. Дано уравнение

$$y'' = y/x.$$

Его можно записать в виде $\text{EQ} := (\text{D@@2})(y)(x) = y(x)/x$ или $\text{EQ} := \text{diff}(y(x), x\$2) = y(x)/x$. Применим оператор `convertsys(EQ, [], y(x), x)`. Получим

$$[[YP_1 = Y_2, YP_2 = \frac{Y_1}{x}], [Y_1 = y, Y_2 = y_x], \text{undefined}, []].$$

По умолчанию Y — обозначение для вектора переменных. Начальные значения здесь не заданы, поэтому скобки для них остались пустыми. Для более компактной записи функций и производных здесь применен оператор `declare(y(x))` из пакета `with(PDEtools)`.

- `diffop2de(N, y(x))`. Преобразование дифференциального оператора N в дифференциальное уравнение для функции $y(x)$. Для того чтобы задать оператор, необходимо с помощью переменной окружения¹ `_Envdiffopdomain` определить переменную, вводящую операцию дифференцирования. Пусть это будет литера d : `_Envdiffopdomain:=[dx, x]`. Если этого не сделать, то появится сообщение об ошибке: *Error, (in DEtools/diffop2de) differential algebra not specified*. Определим линейный оператор $N := dx^2 + 2x + x * dx$. Команда `diffop2de(N, y(x))=0` дает результат в виде линейного однородного дифференциального уравнения (5.12), с. 297. Обратное действие выполняет оператор `de2diffop`.

¹Переменная окружения в Maple начинается с символа подчеркивания.

- `dcoeffs(eq, y(x))`. Выделение коэффициентов из дифференциального выражения полиномиального вида. Если применить оператор `dcoeffs(lhs(eq), y(x))` к левой части (5.12), то получим $1, x, 2x$, т. е. коэффициенты при второй производной, первой и нулевой (самой функцией). Оператор `lhs` выделяет левую часть уравнения¹. Другой пример. Дано выражение

$$EQ:=\text{diff}(y(x), x^3) + a*\text{diff}(y(x), x)^2 + b*y(x) + c.$$

Применим оператор `dcoeffs(EQ, y(x))`. Получим: $1, a, b, c$.

- `de2diffop(L, y(x))`. Преобразование линейного однородного дифференциального выражения в операторную форму. Действие, обратное преобразованию `diffop2de(N, y(x))`. Пример. Дано выражение $L:=\text{diff}(y(x), x^3) + a*\text{diff}(y(x), x) + b*y(x)$. Применяем оператор `de2diffop(L, y(x))`. Получаем $Dx^3 + aDx + b$.
- `dfieldplot`. Построение поля направлений для дифференциального уравнения или системы уравнений. Например, для уравнения $y' = -x^2/y$ поле направлений (рис. 369) строится следующим оператором: `dfieldplot(diff(y(x), x)=-x^2/y(x), y(x), x=-1..1, y=-1..1)`.

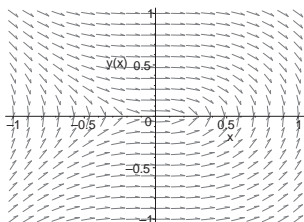


Рис. 369

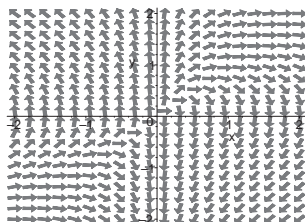


Рис. 370

Поле направлений подсказывает интегральную кривую уравнения. Для системы уравнений построение поля направлений (рис. 370) выглядит следующим образом: `dfieldplot([diff(x(t), t) = x(t)*y(t), diff(y(t), t) = y(t)-x(t)], [x(t), y(t)], t=0..1, x=-2..2, y=-2..2, arrows=LARGE)`.

В опциях оператора `dfieldplot` можно указать вид стрелок `arrows`. Возможны варианты `large, curve, comet, line, small, medium, none, mediumfill, smalltwo`. Допустимо писать эти значения и прописными, и строчными буквами, но не смешивать

¹ А оператор `rhs` — правую.

их. Можно указать цвет (например, `color=blue`), размер стрелок `size` (по умолчанию `size=1`).

- **DEnormal**. Приведение коэффициентов дифференциального уравнения к нормализованной форме — избавление от знаменателей, раскрытие скобок. Пример. Имеем дифференциальное уравнение

$$L := \frac{y'''}{4-x^2} + (1+x)^2 a y' + \frac{y}{x} = x.$$

Применяем оператор `DEnormal(L, x, y(x))`. Получаем

$$(-4ax - 8ax^2 - 3ax^3 + 2ax^4 + ax^5) y' - 4y + yx^2 - xy''' = -4x^2 + x^4.$$

Здесь для сокращения записи применен оператор `declare(prime=x, y(x))` из пакета `PDEtools`, обозначающий производную по x штрихом, а функцию $y(x)$ — просто y .

- **Dchangevar**. Замена переменной. Устаревший оператор, замененный на `dchange` из пакета `PDEtools`. Пример. Дано уравнение $dy(x)/dx = x^2/t$, записанное в виде `eq := diff(y(x), x) = x^2/t`. Замена `dchange(x=2*t, eq)` дает следующий результат:

$$\frac{1}{2} \left(\frac{d}{dt} y(t) \right) = 4t.$$

Для дифференциальных уравнений в частных производных применение этого оператора более эффективно. Так, имеем уравнение $f_x + g(x, y) + f_y = 0$, обозначенное в программе как `PDE`. Замена `dchange({x = u + v, y = u - v}, PDE)` дает

$$f_u + g(u, v) = 0.$$

Если же указать, что g — известная функция, `dchange({x = u + v, y = u - v}, PDE, known=g)`, то получим другой результат $f_u + g(u+v, u-v) = 0$. Для сокращения записи частных производных использован оператор `declare()`.

- **diff_table(y(x))**. Объявление сокращенного способа ввода производных. Если объявить `Y:=diff_table(y(x))`, то уравнение (5.12) можно записать в виде `Y[x, x] + x*Y[x] + 2*x*Y[[]]`. Аналогичное сокращение записи есть и для частных производных. Запишем: `V:=diff_table(v(x, y, z))`. Уравнение Лапласа¹ вводится в виде `V[x, x] + V[y, y] + V[z, z] = 0`.
- **exactsol**. Точное решение дифференциального уравнения первого порядка. Обращение к нему имеет вид `exactsol(ode, x(t))`.

¹*Pierre-Simon Laplace* (1749–1827) — французский математик, физик.



В действительности оператор `dsolve(ode, x(t))` справляется с этой задачей не хуже, а иногда и лучше. Так, применительно к уравнению $t^2 \cdot x(t) \cdot \text{diff}(x(t), t) + t \cdot x(t)^2 = 0$ оба оператора дают одинаковое решение, а оператор `dsolve(ode, x(t))` дает еще и тривиальное $x(t)=0$.

- `hamilton_eqs`. Оператор для вывода дифференциальных уравнений движения в форме Гамильтона. См. пример на с. 386.
- `intfactor(eq)`. Интегрирующий множитель. Практика показывает, что нахождение множителя в лучшем случае дает другую форму решения, зачастую менее компактную, чем просто из обращения к оператору `dsolve`.
- `initialdata(eq)`. Определение формы начальных данных для системы дифференциальных уравнений. Система должна быть разрешена относительно высших производных, иначе оператор дает ошибочное решение. Оператор применим и для уравнений в частных производных.

- `kovaciccsols(ode, y(x))`. Решение в форме Лиувилля уравнения второго порядка по алгоритму¹ J. Kovacic².

Решение Лиувилля предполагает, что в него входят только алгебраические функции, экспоненты и интегралы от алгебраических функций³.

- `linearsol(ode, y(x))`. Решение обыкновенного дифференциального уравнения первого порядка. От решения, полученного стандартным оператором `dsolve`, отличается тем, что оно заключено в скобки.
- `matrixDE(A, B, t)`. Решение системы дифференциальных уравнений в матричной форме: $\vec{X}'(t) = A(t)\vec{X}(t) + \vec{B}(t)$. По умолчанию \vec{B} — нулевой вектор. Например, решаем систему уравнений, записанную в матричной форме:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ t & -1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 2 \\ 1 \end{bmatrix}.$$

¹Kovacic J. An algorithm for solving second order linear homogeneous differential equations. J. Symb. Comp. 2, p. 3–43 (1986), (<http://vuz.exponenta.ru/alg.pdf>)

²Jerald J. Kovacic (1940–2009) — американский математик.

³Алгебраические функции $y(x)$ являются корнями уравнения $P(x, y) = 0$, где P — полином.

Записываем матрицу системы и вектор свободных членов

```
> A:=Matrix([[1,0],[t,-1]]); B:=Vector([2,1]);
```

Решаем систему, используя оператор

```
> s := matrixDE(A,B,t);
```

Получаем ответ также в матричной форме:

$$s := \left[\begin{array}{cc} 0 & 4e^t \\ e^{(-t)} & e^t(2t-1) \end{array} \right], [-2, 3-2t].$$

Первая матрица 2×2 должна быть умножена на вектор констант, вторая матрица добавляется к полученному результату. С помощью оператора транспонирования `Transpose` из пакета `LinearAlgebra` это можно сделать следующим образом:

```
> s[1].<<C1,C2>>+Transpose(convert(s[2],Matrix));
```

В результате имеем вектор решения системы

$$\left[\begin{array}{c} 4e^t C2 - 2 \\ e^{(-t)} C1 + e^t(2t-1)C2 + 3 - 2t \end{array} \right].$$

Обычный способ решения этой же системы имеет вид

```
> e1:=diff(x(t),t)=x(t)+2:
```

```
> e2:=diff(y(t),t)=t*x(t)-y(t)+1:
```

```
> dsolve({e1,e2},{x(t),y(t)});
```

$$\{x(t) = -2 + e^t_c2, y(t) = -2t + 3 + \frac{1}{2}te^t_c2 - \frac{1}{4}e^t_c2 + e^{(-t)}_c1\}.$$

Решения совпадают с точностью до константы $C2$.

- `muchange`. Если в дифференциальном уравнении произведена замена переменной оператором `dchange`, то эту же замену в интегрирующем множителе выполняет оператор `muchange`.
- `mutest`. Если с помощью `M:=intfactor(eq)` найден интегрирующий множитель `M` уравнения `eq`, то проверить его можно так: `mutest(M,eq)`. Должен получиться ноль.
- `ode_y1(eq)`. Получение дифференциального уравнения, которому удовлетворяет y' , где y — решение уравнения `eq`.
- `ode_int_y(eq)`. Получение дифференциального уравнения, которому удовлетворяет $\int y dx$, где y — решение уравнения `eq`.
- `odeadvisor(ODE)`. Классификация уравнения и подсказка метода решения. Различаются следующие типы уравнений первого порядка: `Abel`, `Abel2A`, `Abel2C`, `Bernoulli`, `Chini`, `Clairaut`, `dAlembert`, `exact`, `homogeneous`, `homogeneousB`, `homogeneousC`, `homogeneousD`, `homogeneousG`, `linear`,

patterns, quadrature, rational, Riccati, separable, sym_implicit.

Уравнения второго порядка: Bessel, Duffing, ellipsoidal, elliptic, Emden, erf, exact_linear, exact_nonlinear, Gegenbauer, Halm, Hermite, Jacobi, Lagerstrom, Laguerre, Lienard, Liouville, linear_ODEs, linear_sym, missing, Painleve, quadrature, reducible, sym_Fx, Titchmarsh, Van_der_Pol.

Уравнения высших порядков: quadrature, missing, exact_linear, exact_nonlinear, reducible, linear_ODEs.

Пример. Рассмотрим уравнение (4.82) на с. 217. Оператор `odeadvisor` даст следующее заключение `[[_2nd_order, _missing_x]]`, означающее, что это уравнение 2-го порядка и в нем нет переменной t (всегда обозначается как x) в явном виде. Если изменить это уравнение и добавить сомножителем t в первое слагаемое: $16t\ddot{x}\sqrt{x} + b\dot{x} + x^4 = 0$, то оператор `odeadvisor` даст еще более неинтересное заключение: `[NONE]`. Поэкспериментируем еще с этим уравнением. Уберем последнее слагаемое x^4 . Получим от оператора `odeadvisor` следующее заключение: `[[_2nd_order, _missing_x], [_2nd_order, _exact, _nonlinear]]`. Очевидно, замена переменной $x' = u$ понижает порядок этого уравнения до первого, решение которого легко получить. Именно поэтому в заключении стоит слово `exact`, означающее, что есть точное решение.

- `redode(f(x), y(x))`. Оператор возвращает дифференциальное уравнение, для которого $f(x)$ есть интегрирующий множитель. Проверку правильности нахождения уравнения можно выполнить оператором `intfactor`. По умолчанию определяется уравнение первого порядка. В общем случае для уравнения n -го порядка обращение к оператору имеет вид `redode(f(x), n, y(x))`.
- `riccatisol(ode, x(t))`. Решение уравнения Риккати¹. Заметим, что обычный оператор `dsolve(ode, x(t))` справляется с этой задачей не хуже. Общий вид уравнения Риккати: $x' = f(t) + g(t)x + h(t)x^2$. См. также тип `Riccati` из оператора `odeadvisor`.

¹Jacopo Francesco Riccati (1676–1754) — итальянский математик.

ПРОГРАММИРОВАНИЕ MAPLET

В этой главе кратко описано программирование маплетов — так мы будем называть программы с графической пользовательской оболочкой **Maplet**. Программирование маплетов возможно двумя способами. Первый способ — написание обычной программы в процессе **Maple**-сессии. В этом случае после отладки программы, записанной в **mws**-файл, например, **test.mws**, ее надо сохранить в виде **test.maplet**. Сделать это надо из меню **Maple: File->Save as**. Получится текстовый файл, который можно не только прочитать, но и отредактировать в любом текстовом редакторе. Другой способ — использование специальной программы визуального программирования **Maplet Builder**, встроенной в **Standard Worksheet**. Краткое описание **Maplet Builder** см. на с. 335.

Программирование маплетов имеет свою специфику. Несмотря на то что встроенная справка по **Maple** весьма обширна и содержит не только значительное число примеров, но и пошаговую инструкцию программирования, приведем описание некоторых основных элементов и их опций, необходимых для работы. Сразу заметим, что программирование пользовательского интерфейса в **Maple** не так развито, как, например, в **Borland Builder** или **Delphi**. В маплетах есть многочисленные повторения и ошибки. Так, например, система не переносит строчную русскую букву **я**, но допускает прописную **Я**. Причем в режиме программирования это не видно, но самостоятельного запуска маплета (т. е. запуска файла с расширением ***.maplet** вне сессии **Maple**) даже с **я** в комментариях не получается¹. Менее опасна для маплетов строчная буква **ю**. В процессе отладки она видна, а при самостоятельной работе маплета она неожиданно бесследно исчезает с экрана, коверкая при этом слова. Эти замечания должны объяснить читателю, почему автор в некоторых местах пишет, например, «Исследуемая опора», как на испорченной пишущей машинке, и опускает окончания слов, содержащих буквы **я**.

Все программы рассчитаны на версию **Maple 13**. Однако, почти без изменений они работоспособны и в других версиях **Maple**.

Тексты примеров из этой главы содержатся в архиве <http://vuz.exponenta.ru/maplets.rar>.

¹Подобные ошибки были в версии **Maple 6** за 2000 г. и ранних версиях **MATLAB**. Но они были быстро исправлены. В новой версии **Maple 15** за 2011 г. эта ошибка сохранена, но появление **я** в комментариях уже допускается.

6.1. Первая программа

Сначала приведем простейшую и традиционную программу, не выполняющую никаких действий, кроме отображения на экране слова "Hello!". Как минимум тут должно быть три команды. Во-первых, надо подгрузить специализированный пакет `Maplets[Elements]`, содержащий 63 оператора, один из которых — оператор `Maplet` — для создания маплета. В этом операторе есть обязательная пара круглых скобок и внутри нее — пара квадратных скобок, заключающая отображаемый элемент. В данном случае это слово "Hello!". Кавычки здесь ставить не обязательно. Если есть кавычки, тогда отображается *строка*. Без кавычек получится переменная `Hello`, которая будет отображаться, если она свободная, т. е. ей не было присвоено какое-либо значение. Значение, присвоенное переменной, например `Hello:=1`, приведет к ошибке. Число не может быть отображено в таком месте. Допустимо (но бессмысленно) присвоение `Hello:="Hello!"`, лучше сразу выводить *строку*. Полученный маплет присвоим переменной `mplt`, которую выведем на экран (фактически запустим) оператором `Display` пакета `Maplets`. Помимо уже упомянутого пакета `Elements`, в этом пакете есть вложенные пакеты `Tools`, `Utilities`, отдельный оператор `Display` и набор примеров `Examples`.

Пример 1

```
> with(Maplets[Elements]):  
> mplt:=Maplet(["Hello!"]):  
> Maplets[Display](mplt);
```



Рис. 371

Оператор вывода маплета `mplt` на экран может иметь другие формы: `with(Maplets): Display(mplt):` или `Maplets:-Display(mplt);`

Итак, получен простейший маплет. Он работает пока только из оболочки **Maple**. Можно сделать его более самостоятельным. К сожалению, **Maple** не создает исполняемые *.exe файлы, но можно получить в какой-то мере независимую программу, которая работает без запуска **Maple**. Для этого надо полученную программу сохранить, используя меню **Maple**. В интерфейсе имеем цепочку действий:

File→Save as →maplet.

Выберем имя маплета и получим файл с расширением *.maplet. Это текстовый файл, который можно смотреть и править в любом текстовом редакторе, например, WordPad или Блокнот. Отладку удобно производить в текстовом редакторе, контролирующем пары открывающих и закрывающих скобок¹. Запустить этот файл можно на любом компьютере, где установлена система **Maple**. По расширению *.maplet будет автоматически вызвана программа MapletViewer.exe размером 104 кб, расположенная в папке C:\ Program Files\Maple 13\bin.win. Если в программе ошибок нет, то через секунду-другую (не мгновенно, надо выждать!) после заставки MapletViewer.exe (рис. 372) появится окно маплета. Если в программе есть ошибки, то MapletViewer.exe, как правило, не указывает на них, а просто молча игнорирует маплет. Ошибки надо тестировать в среде **Maple**, запуская программу, создающую маплет.



Рис. 372

Усложним первый маплет, вводя постепенно необходимые элементы и меняя способы их отображения.

Способ изображения элементов маплета в окне программы совпадает с заданием матрицы. Так, если матрица

$$\begin{bmatrix} a1 & a2 & a3 \\ b4 & b5 & b6 \end{bmatrix}$$

задается в виде

> Matrix([[a1, a2, a3], [b4, b5, b6]]),
то маплет (рис. 373) имеет схожую структуру:

Пример 2

> with(Maplets[Elements]):

¹Например, WinEdt.

```
> mplt :=Maplet([[a1,a2,a3],[b4,b5,b6]])
> with(Maplets): Display(mplt):
```

Можно использовать явное обозначение строк:

```
> mplt := Maplet([BoxRow(a1,a2,a3),BoxRow(b4,b5,b6)]):
```

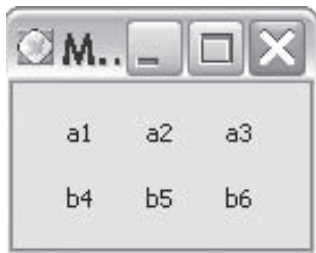


Рис. 373

Аналогично можно использовать и элементы-столбцы:

```
> mplt :=Maplet([BoxRow(BoxColumn(a1,b4),
>                        BoxColumn(a2,b5),
>                        BoxColumn(a3,b6))]):
```

Результат будет тот же (рис. 373). На первый взгляд было бы проще писать без элементов `BoxColumn` и `BoxRow`. Однако это не так. Эти элементы имеют опции, задание которых расширяет способы и качество изображения.

Опишем опции `BoxColumn` и `BoxRow` и опции других основных элементов для программирования маплетов. Опции всех элементов можно записывать в кавычках: `'border'=true`, обратных кавычках ``border`=true` или вообще без них: `border=true`. Значение опции, в данном случае `true`, записывается по-разному, в зависимости от типа. Логическая константа `true` может быть в виде `'true'` или `'true'`.

6.2. BoxColumn, BoxRow

Столбец и ряд. Эти элементы могут содержать в себе произвольное число элементов `BoxCell`, `BoxColumn`, `BoxRow` и имеют следующие опции:

- `background` — цвет фона за текстом. Например,

```
mplt := Maplet([BoxRow(background=red,a1,a2)]):
```

Текст будет на красном фоне. Цвет также можно задать в виде `background="#FF0000"`. Красный, зеленый и синий цвет в системе RGB задаются шестнадцатеричными числами. Синий цвет

соответствует константе "#0000FF", белый — "#FFFFFF", черный — "#000000".

- `border` — видимость границы. Логическая величина. Принимает значения `true` и `false` (по умолчанию).
- `caption` — текст заголовка на границе. Имеет смысл, когда `border=true`.

Пример 3

```
> with(Maplets[Elements]):
> mplt := Maplet([BoxRow(border=true,
> 'caption'="Кафедра МГУ", "Теории пластичности",
> "Теории упругости")]):
> with(Maplets):
> Display(mplt):
```

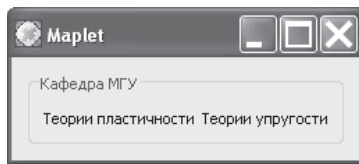


Рис. 374

- `font` — шрифт заголовка `caption`, если он существует. Например, `font=Font("helvetica",italic,20)`. Первый параметр — наименование шрифта. Всегда доступны также шрифты Times, Courier, Symbol. Имя может быть написано в произвольном регистре букв, Courier или courier. Эти шрифты относятся к семейству физических. Шрифты логического семейства: Serif, SansSerif, Monospaced, Dialog, DialogInput. Семейство шрифтов можно выбрать, не указывая конкретно название `Font(family="Monospaced")`. Будет выбран шрифт этого семейства, назначенный по умолчанию. Второй параметр — характер написания, в данном случае курсив. Для полужирного шрифта есть параметр `bold`. Равноправны записи `bold=true` и просто `bold`. Третий параметр — размер шрифта в пунктах (point).

Размер можно указывать более конкретно `size=20`. Одна или две характеристики шрифта из трех могут быть опущены.

- `halign` — способ горизонтального выравнивания. Три варианта: `left`, `center` или `right`. Для горизонтального выравнивания

можно использовать "горизонтальный клей" `HorizontalGlue()`, а для вертикального — `VerticalGlue()`¹. Раздвинем, например, надписи на рис. 374 и вставим сверху от них небольшой промежуток. Используем команду-повторитель `$25` в первом случае и `$3` — во втором (рис. 375):

Пример 4

```
> with(Maplets[Elements]):
> mplt :=Maplet([VerticalGlue()$3,
> BoxRow(border=true,'caption'="Кафедра МГУ",
> "Теории пластичности",
> HorizontalGlue()$25,"Теории упругости")]):
> with(Maplets):
> Display(mplt):
```



Рис. 375

- `hscroll` — возможность горизонтальной прокрутки содержимого специальным ползунком. Имеет три значения: `never`, `as_needed`, `always` (никогда, по необходимости, всегда). По умолчанию прокрутки нет: `hscroll=never`.
- `inset` — натуральное число, пространство вокруг элемента. Измеряется в пикселах. По умолчанию `inset=5`.
- `spacing` — натуральное число, промежуток между рядами (`BoxRow`) или столбцами (`BoxColumn`). Измеряется в пикселах. По умолчанию `spacing=5`.
- `valign` — способ вертикального выравнивания. Три варианта: `bottom` (внизу), `center` (в центре) или `top` (наверху).
- `visible` — видимость элемента. Логический параметр со значением `true` или `false`. По умолчанию принято `true`.

¹Эти опции появились в Maple 9.5.

- `vscroll` — возможность вертикальной прокрутки содержимого специальным ползунком. Имеет три значения: `never`, `as_needed`, `always`. По умолчанию прокрутки нет: `hscroll=never`.

6.3. Window

Окно. Как видно из рассмотренных примеров, маплет и без этого элемента создал окно. Он это делает в любом случае, в этом его основное предназначение. Однако элемент `Window` позволяет уточнять характеристики окна. Таких характеристик (опций) одиннадцать. Опишем их.

- `defaultbutton` — ссылка на кнопку (`Button`), которая сработает при нажатии клавиши `Enter`. Приведем пример (рис. 376). Окно содержит поле вывода `TextField` и две кнопки с именами `A` и `B`. На кнопках надписи `A1` и `B2` соответственно. Действие, которое будет вызвано при их нажатии, прописано сразу после подписи. В данном случае это присвоение (`SetOption`) полю вывода текстового элемента с именем `T` некоторых значений. Одно значение — свободная переменная `C216`, другое — переменная `a`, которой ранее была присвоена величина `11`, поэтому в поле вывода появится это число. Для `Window` указана конкретная кнопка с именем `B` и надписью `B2`, срабатывание которой происходит при нажатии `Enter`. Если эту опцию опустить, сработает первая из указанных кнопок, т. е. `A`.

Пример 5

```
> with(Maplets[Elements]):
> a:=11:
> b1:=Button["A"]("A1 ",SetOption('T'=C216)):
> b2:=Button["B"]("B2",SetOption('T'=a)):
> mplt:=Maplet(Window('defaultbutton'="B",
> [TextField['T'](),[b1,b2]])):
> with(Maplets): Display(mplt):
```



Рис. 376



Рис. 377

Каждой кнопке при необходимости можно легко сопоставить "горячую клавишу" (mnemonic key), так чтобы не мышью, а с клавиатуры можно было бы нажать на кнопку. Для этого в надписи на кнопке перед какой-либо буквой достаточно поставить амперсанд:

```
Button["A"]("&A1 ",SetOption('T'=C216)).
```

Соответствующая буква будет подчеркнута, а кнопка активируется комбинацией Alt-A. Надо, конечно, следить, чтобы эта комбинация не противоречила какой-либо системной. Аналогичное действие можно запрограммировать в элементах `CheckBox`, `CheckBoxMenuItem`, `Menu`, `MenuItem`, `RadioButton`, `RadioButtonMenuItem`, `ToggleButton`.

- `height`, `width` — высота и ширина окна в пикселах. По умолчанию размеры выбираются оптимальными, так, чтобы были видны все элементы, какие возможно (размеры некоторых элементов можно задать по ошибке нереальными). Если, например, в предыдущем примере специально задать размеры окна `100x100`: `Window(height=100,width=100, ...)`, то часть окна будет обрезана (рис. 377). Это некрасиво, но спасти ситуацию можно, просто раздвинув окно мышью.
- `menubar` — вызывает меню программы. Меню (`MenuBar`) сначала надо организовать. Пример меню с одной кнопкой (`Menu`), содержащей один пункт (`MenuItem`), см. на с. 427. Рассмотрим, как получить меню с двумя пунктами одной кнопки (рис. 378).

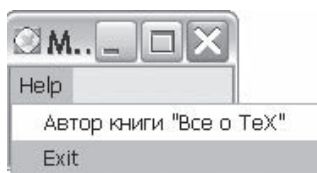


Рис. 378



Рис. 379

Пример 6

```
> with(Maplets[Elements]):
> a1:='onclick'=RunWindow('H'):
> a2:=Shutdown():
> b:=MenuItem("Автор книги Все о TeX ",a1):
> c:=MenuItem("Exit",a2):
> mplt:= Maplet(Window('menubar'='M1',[ "Окно" ]),
>               Window['H'](["Д.Кнут"]),
>               MenuBar['M1']( Menu("Help",b,c))):
```

```
> with(Maplets): Display(mplt):
```

Окно справки Window['Н'] и меню необходимо записать внутри определения маплета, до или после основного окна Window. Если требуется поставить кавычки внутри строки, то их надо записать с управляющим символом \ (backslash): \"Все о TeX\". Если требуется создать меню с двумя кнопками (рис. 379), достаточно записать, например

```
Menu("File",b1,c1),Menu("Help",b,c),
и описать соответствующие пункты b1, c1.
```

- **reference** — метка (имя) окна. Обычно она задается в квадратных необязательных скобках Window['Н'](["Д.Кнут"]), но может быть задана и с помощью этой характеристики: Window('reference'='Н',["Д.Кнут"]). Если же заданы две метки одновременно двумя разными способами, то преимущество имеет метка в квадратных скобках.
- **resizable** — опция, задающая возможность изменять мышью размер окна в процессе работы маплета. Имеет значения true или false. По умолчанию resizable= true.
- **title** — заголовок окна.
- **toolbar** — вызывает инструментальную панель программы ToolBar['name1'] по ее имени, например, 'name1'. Вызов аналогичен вызову меню: toolbar='name1'. Панель представляет организованный набор кнопок или групп кнопок, разделенных специальной линией ToolBarSeparator, и похожа на меню. Отличия: меню можно организовать как ниспадающее, и в панели четко видны кнопки. Панель (ToolBar) надо организовать в теле Maplets. Приведем пример. Назовем панель 'myB'. Поместим на нее две кнопки, разделенные линией (рис. 380). Нажатие одной кнопки приводит к вычислению синуса аргумента, помещенного в текстовое поле с именем 'T', другой — косинуса. Результат заносится в то же поле.

Пример 7

```
> with(Maplets[Elements]):
> ts:=ToolBarSeparator():
> ev1:=Evaluate('T' = 'sin(T)'):
> ev2:=Evaluate('T' = 'cos(T)'):
> bt1:=ToolBarButton("sin", 'onclick'=ev1):
> bt2:=ToolBarButton("cos", 'onclick'=ev2):
> mplt:=Maplet(ToolBar['myB'](bt1,ts,bt2),
```

```
> Window('toolbar'='myB', [TextField['T']()]):
> with(Maplets):
> Display(mplt):
```

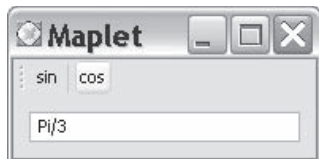


Рис. 380

При желании можно добавить кнопку выключения программы: `bt3:=ToolBarButton("Exit", Shutdown())`. Если не отделять ее линией от других, то аргумент `ToolBar` будет иметь вид `(bt1, ts, bt2, bt3)`. На кнопке будет надпись `Exit`.

- `xcoord`, `ycoord` — отступ от левого и верхнего края экрана. По умолчанию маплет центрируется. Если взять `Window(xcoord=0, ycoord=0, ...)`, то маплет будет помещен в левый верхний угол экрана.

Элемент `Window` сам может содержать в себе 15 видимых элементов: `Button`, `CheckBox`, `ComboBox`, `DropDownBox`, `Label`, `ListBox`, `MathMLEditor`, `MathMLViewer`, `Plotter`, `RadioButton`, `Slider`, `Table`, `TextBox`, `TextField`, `ToggleButton`. Все эти элементы записываются через запятую в тело `Window`. Например: `Window(Button(), TextField(), TextBox())`. Аргументы элементов здесь условно опущены. Некоторые элементы уже разбирались в предыдущих примерах. Рассмотрим теперь их работу подробнее и опишем остальные.

6.4. Label

Метка. Служит для вывода информации. Имеет следующие опции:

- `background` — цвет фона (см. с. 307).
- `caption` — текст метки. Пример: `Label(caption="МЭИ")`. Результат не отличается от простого обращения `Label("МЭИ")`.
- `enabled` — логический параметр со значением `true` или `false`. По умолчанию принято `true`. Если `enabled=false`, текст метки блокируется. Надпись имеет цвет близкий к фону. Если метка является изображением, то изображение мутнеет, подсказывая пользователю существование этой метки, но ее неработоспособности.

- `font` — шрифт (см. с. 308).
- `foreground` — цвет шрифта. Пример: `Label(background=blue, foreground=white, "TeX")` — белый текст по синему фону.
- `halign` — способ горизонтального выравнивания, или расположения текста внутри метки. Три варианта `left`, `center` или `right`. Имеет смысл, когда задана опция `width` и ширина поля текста метки больше текста. По умолчанию `halign=center`.
- `height` — высота метки. По умолчанию подбирается по размеру надписи или рисунка.
- `image` — этот параметр указывает на метку изображения.

Пример 8

```
> with(Maplets[Elements]):
> mplt:=Maplet(["Алупка", Label(image=a)], Image[a]
> (font=Font("TimeS", 40), "C:\\ris\\AL.jpg")):
> with(Maplets): Display(mplt):
```



Рис. 381

Можно и непосредственно вставить изображение в надпись (рис. 381):

```
> mplt := Maplet(["Алупка",
> Label(Image("C:\\ris\\AL.jpg"))]):
```

Результат будет тот же. Допустимы также файлы в формате `png` и `gif`, в том числе и `gif`-анимация.

- `reference` — см. с. 312.

- `tooltip` — подсказка (символ или строка). Текст в окне, всплывающем при наведении мыши на элемент. В большинстве систем программирования такая подсказка называется `hint`.
- `valign` — способ вертикального выравнивания, или расположения текста внутри метки. Три варианта: `bottom` (внизу), `center` (в центре) или `top` (наверху). Имеет смысл, когда задана опция `height`. По умолчанию `valign=center`.
- `visible` — видимость метки. Логический параметр со значением `true` или `false`. По умолчанию принято `true`. Если `visible=false`, метка исчезает, и место под нее не отводится.
- `width` — ширина метки в пикселах. По умолчанию подбирается по размеру надписи или рисунка.

6.5. Button

Управляющая кнопка. Имеет следующие опции: `background`, `caption`, `enabled`, `font`, `foreground`, `halign`, `height`, `image`, `reference`, `tooltip`, `valign`, `visible`, `width`, описанные в элементе `Label`, и еще одну опцию `onclick` — действие, выполняемое при нажатии кнопки. Работает аналогично той же опции элемента `ToolBarButton`, см. с. 312. Несколько действий, выполняемых при нажатии одной кнопки, могут быть объединены функциональной скобкой `Action`, см. с. 427, 452, 455, 475.

6.6. TextField

Задаёт строку ввода или вывода. Имеет следующие опции: `background`, `enabled`, `font`, `foreground`, `halign`, `reference`, `tooltip`, `visible`, `width`, описанные выше, и следующие:

- `editable` — опция, задающая возможность редактирования пользователем текста. Имеет значения `true` или `false`. По умолчанию `editable=true`.
- `onchange` — команда, группа команд в скобках `Action` или ссылка на команду. Всякое изменение текста приводит к выполнению указанной команды. В следующем примере это счетчик изменений. Число изменений первой строки подсчитывается во второй, как только во второй фиксируется курсор (рис. 382).

Пример 9

```
> with(Maplets[Elements]):
```

```

>      mplt := Maplet(Window([
>      TextField('onchange'=Evaluate('C' ='C'+1)),
>      TextField['C'](value=1)]):
>
>                                     )#Maplet
> Maplets[Display](mplt):

```

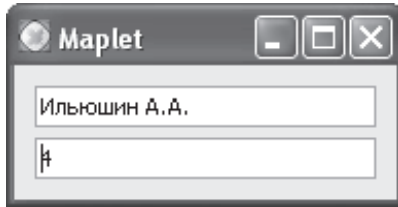


Рис. 382

Первое поле осталось безымянным, имя второго поля (оно же — значение 1) необходимо для того, чтобы менять его первоначальное значение 1 командой `Evaluate`. Допустима форма $C = C + 1$ (без кавычек).

- `popupmenu` — задает ссылку на меню, всплывающее при нажатии правой клавиши мыши.

Пример 10

```

> with(Maplets[Elements]):#Всплывающее меню
> mplt := Maplet(
>   Window([
>     TextField['T']('popupmenu'='P', 'value'="x")],
>   PopupMenu['P'](
>     Menu("Calculate",
>       MenuItem("diff", Evaluate('T'='diff(T,x)')),
>       MenuItem("int ", Evaluate('T'='int (T,x)')),
>       MenuItem("Exit", Shutdown()) )):
> Maplets[Display](mplt);

```

Эта программа создает маплет с одним полем вывода, в котором опцией `value` уже записано значение `x`. Всплывающее по указанию правой клавиши мыши меню содержит два пункта `Calculate` и `Exit`. Пункт `Calculate` имеет еще два подпункта: `diff` и `int`. Этим операциям подвергается содержимое текстового поля `T`. При этом поле по ходу вычислений можно подправлять, например продифференцировать,

получить результат (он появляется в том же поле) и дописать туда еще что-нибудь, а потом проинтегрировать. Следует однако помнить, что вычисление интегралов в аналитической форме случай не очевидный, поэтому не надо увлекаться сложными формулами. Компьютер может просто надолго зависнуть в ожидании аналитического результата, часто на нескольких строчках или даже страницах. Упрощенная структура этого примера следующая:

```
> mplt := Maplet(
>   Window([TextField()]),
>   PopupMenu(
>     Menu("Calculate",MenuItem("diff",...),
>       MenuItem("int",...)),
>     MenuItem("Exit",...)  ):

```

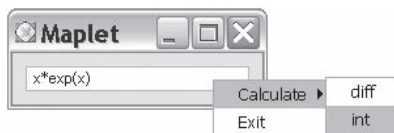


Рис. 383

- **value** — содержимое текстового поля. Допустима как форма `'value'=200`, так и с кавычками `'value'="200"`. Справочные материалы **Maple** рекомендуют использовать конвертацию чисел в текст `sprintf("%a", value)`. Однако опыт показывает, что это излишне, числовые данные легко воспринимаются в аргументе `value`.

6.7. CheckBox

Переключатель из независимых кнопок. Имеет следующие опции: `background`, `caption`, `enabled`, `font`, `foreground`, `image`, `reference`, `tooltip`, `value`, `visible`, описанные в элементах `Label` и `TextField`, и опцию `onchange`, задающую команду или группу команд в скобках `Action` или ссылку на команду. Всякое изменение состояния кнопки (всего два состояния) приводит к выполнению указанной команды. В следующем примере это счетчик изменений. Переключатель состоит из двух кнопок (рис. 384). Кнопки имеют логические значения, которые присваиваются их именам. Величины 'А' и 'В' логические. В зависимости от состояния кнопки имеют значения или `'true'`, или `'false'`. Эти значения передаются процедуре `Результат`, в которой указывается тип аргумента `boolean`. Указывать тип аргумента желательно там, где возможна ошибка при его вводе. Тогда появится

соответствующее предупреждение. Здесь же это не обязательно, так как заранее известно, что аргументами будут или 'А', или 'В', а они логические. Если кнопка нажата и помечена, то в соответствующем поле появляется надпись "Эйлер", если кнопка не выбрана (пустой квадратик), то после нажатия кнопки Run появляются годы жизни великого математика — "1707-1783". Все нажатия кнопок подсчитываются в переменной N.

Пример 11

```
> Результат:=proc(x:boolean)
>   if x then 'Эйлер' else '1707-1783' fi;
> end proc:
> with(Maplets[Elements]):
> f:=font=Font(18):
> N:='onchange'=Evaluate('T3'='T3'+1):
> maplet := Maplet(Window([
>   [CheckBox['A'](f,caption="левый",N),
>   CheckBox['B'](f,caption="правый",N)],
>   [TextField['T1'](f),TextField['T2'](f)],
>   TextField['T3'](f,value=1),
>   Button(f,"Run",'onclick'=Action(
>     Evaluate('T1'='Результат(A)'),
>     Evaluate('T2'='Результат(B)')))])):
> Maplets[Display](maplet);
```

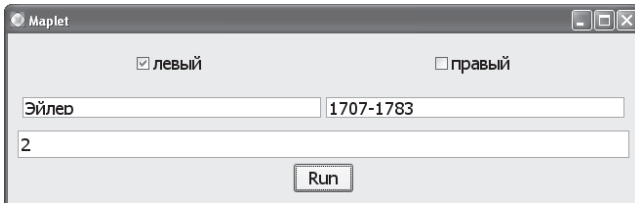


Рис. 384

6.8. TextBox

Если TextField задает только одну строку ввода или вывода, то элемент TextBox задает несколько строк. Имеет следующие опции: background, editable, enabled, font, foreground, height, onchange, popupmenu, reference, tooltip, value, visible,

`width`, описанные выше¹, и опцию `wrapped`, регулирующую перенос текста на следующую строку. Эта опция логического типа. По умолчанию `wrapped=true`, и перенос производится (рис. 385). Если `wrapped=false`, то строки записываются в том же виде, как они были даны. Поэтому часть текста может оказаться невидимой, что, однако, не мешает раздвинуть мышью окно или передвинуть горизонтальный движок окна (рис. 386).

Пример 12

```
> x:="На Ленинских Горах щебечут воробьи,  
> на Воробьевых - ласточки летают,\n  
> На купола Москвы отсюда посмотри,  
> и сердце в сладкой боли замирает..":  
> mplt:=Maplet(Window([TextBox(font=Font(20),  
>   wrapped=true,width=20,height=7,'value'=x)])):
```

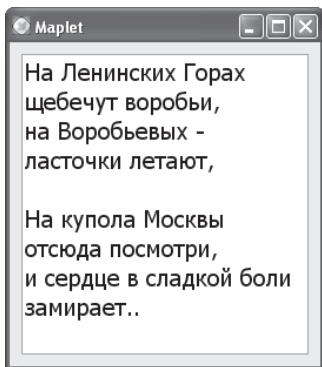


Рис. 385

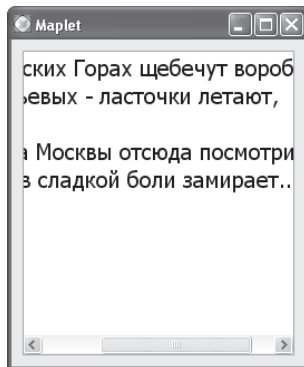


Рис. 386

6.9. ToggleButton

Кнопочный переключатель. Имеет уже известные опции `background`, `caption`, `enabled`, `font`, `foreground`, `image`, `onchange`, `reference`, `tooltip`, `value`, `visible` и опцию `group`. В следующем примере маплет содержит две зависимые (см. `RadioButton`) кнопки, расположенные столбцом (рис. 387). Если убрать из каждой кнопки опцию `'group'='G1'` и группу `ButtonGroup['G1']()`, то переключатель станет независимым.

¹Есть некоторая особенность опций `height`, `width`. Для них в качестве единиц измерения взяты размеры буквы, высота и ширина.

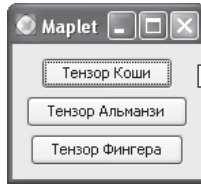


Рис. 387

Пример 13

```
> mplt := Maplet([
> ToggleButton['t1']("Тензор Коши", 'group'='G1'),
> ToggleButton['t2']("Тензор Альманзи", 'group'='G1'),
> ToggleButton['t3']("Тензор Фингера", 'group'='G1')],
> ButtonGroup['G1']()):
```

6.10. Table

Таблица. Все опции уже описаны: `background`, `font`, `foreground`, `height`, `reference`, `tooltip`, `visible`, `width`. Высота и ширина измеряются в пикселах. Особенность этого элемента в том, что он нормально работает только совместно с `BoxCell(..., 'as_needed')` или `GridRow` с той же опцией `as_needed`, указывающей на введение при необходимости полосы прокрутки для просмотра всей таблицы, если часть таблицы не видна. Иначе заголовок таблицы не будет виден.

Пример 14

```
> with(Maplets[Elements]):
> mplt:=Maplet([BoxCell(Table([1,2],[[q,mv],[F,ma]]),
> 'as_needed'))):
> Maplets[Display](mplt);
```

Здесь заголовком является строка `[1,2]` (рис. 388). Она обязательна, даже если и не будет видна.

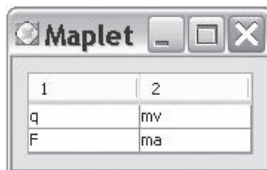


Рис. 388

Этот же результат можно получить, указывая явно, что является заголовком таблицы (`TableHeader`), что — рядом (`TableRow`), а что — ячейкой (`TableItem`). Содержимое ячейки можно записать в нее непосредственно (`TableItem(ma)`) либо с помощью опции `caption`:

Пример 15

```
> with(Maplets[Elements]):
> mplt := Maplet([BoxCell( Table(
> TableHeader(TableItem("1"),TableItem("2")),
> TableRow(TableItem('caption'=q),TableItem(mv)),
> TableRow(TableItem('caption'=F),TableItem(ma))),
> 'as_needed'))):
> Maplets[Display](mplt);
```

6.11. Slider

Регулятор-ползунок. Имеет опции `background`, `enabled`, `foreground`, `onchange`, `reference`, `tooltip`, `value`, `visible`, описанные выше для других элементов, и ряд специальных:

- `filled` — опция, задающая видимость прорези, в которой движется ползунок. Имеет значения `true` или `false`. По умолчанию `filled=true` и прорези не видно (рис. 389). При `filled=false` имеем вид на рисунке 390.

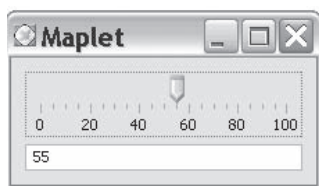


Рис. 389

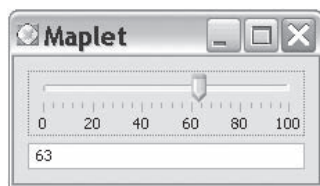


Рис. 390

- `lower` — наименьшее целое значение, принимаемое регулятором. По умолчанию равно 0. Допускаются отрицательные величины.
- `major ticks` — шаг для подписанных значений регулятора (главные значения). По умолчанию подписанных значений нет.
- `minor ticks` — шаг отметок на шкале (маленькие черточки между большими подписанными). По умолчанию таких отметок нет. Желательно брать значение `major ticks` кратным `minor ticks`. Наименьшее значение `minor ticks` равно 1.

- `orientation` — ориентация регулятора. Два варианта: `vertical`, `horizontal`. По умолчанию шкала горизонтальная.
- `showlabels` — опция, задающая подпись главных значений (большие метки). Имеет значения `true` или `false`. По умолчанию `showlabels = true`.
- `showticks` — видимость меток. По умолчанию `showticks = true`. При `showticks = false` видны только подписи главных значений.
- `snapticks` — опция, регулирующая точность выбора значений регулятора. По умолчанию `snapticks=true`, и значения получают кратными `minorticks`, регулятор нельзя поставить между отметок, он "прилипает" к ближайшей. При `snapticks=false` можно задать любое целое число от `lower` до `upper`.
- `upper` — наибольшее целое значение, принимаемое регулятором. По умолчанию 100. Допускаются отрицательные величины.

Запись некоторых опций можно упростить. Числа в аргументе `Slider`, разделенные двумя точками, принимаются за величины `lower` и `upper`. Отдельное число — `value`. По умолчанию `value=0`. Порядок указания опций безразличен. В следующем примере (рис. 389) задано `lower=0`, `upper=100`, `value=0`. Значение указателя помещается в поле вывода `TextField`, как только происходит его изменение (`onchange`):

Пример 16

```
> with(Maplets[Elements]):
> mplt:=Maplet(Window([
>   Slider[S1](0..100,
>     minorticks=5,majorticks=20,snapticks=false,
>     onchange=Evaluate(B2 =S1)),
>   TextField[B2]())):
> Maplets:-Display(mplt);
```

6.12. RadioButton

Переключатель на два зависимых положения (радиокнопка). Имеет опции `background`, `caption`, `enabled`, `font`, `foreground`, `group`, `image`, `reference`, `tooltip`, `value`, `visible`. Пример см. в программе 40, с. 424.

6.13. Plotter

Просмотр рисунков и графиков. Имеет описанные выше опции `background`, `height`, `reference`, `tooltip`, `value`, `visible`, `width`. Высота `height` и ширина `width` по умолчанию равны 400 пикселям. Опишем остальные девять опций, относящихся к управлению анимацией:

- `continuous` — заикливание проигрывателя анимации. Имеет значения `true` или `false`. По умолчанию `continuous = true`, и после завершения анимация начинается снова.
- `delay` — промежуток времени между кадрами в миллисекундах. По умолчанию `delay=100`.
- `frame_backwards` — пошаговая прокрутка анимации назад, если имеет значение `true`.
- `frame_forward` — пошаговая прокрутка анимации вперед, если имеет значение `true`.
- `pause` — пауза, если `pause=true`.
- `play` — пуск, если `play=true`. По умолчанию `play=false`.
- `'stop'` — остановка, если `'stop'=true`.
- `to_end` — демонстрация последнего кадра, если `to_end=true`. Анимация должна быть предварительно остановлена.
- `to_start` — демонстрация первого кадра, если `to_start=true`. Анимация должна быть предварительно остановлена.

Приведем программу анимации с двумя опциями `play` и `'stop'`. Опции анимации по определению не статичны, т. е. они созданы, чтобы их менять в процессе работы приложения. Для обеспечения возможности их изменения эти опции имеет смысл записывать в управляющие команды кнопок. С этой целью используем команду `SetOption`. Сама анимация, полученная оператором `animate`¹, записывается в переменную `g1`. Число кадров в операторе `animate` задается параметром `frames`. По умолчанию `frames=25`. В этом же операторе указываются пределы изменяемой величины `a=1..9` (управляющий параметр анимации). Изображение представляет собой график синусоиды с изменяемым сдвигом по оси *y* и переменным во времени периодом (рис. 391). Область, в которой строится график,

¹Анимацию можно получить и с помощью опции `insequence=true`, с. 354. Такой вариант более удобен для анимации механизмов и графических объектов.

обычно явно задается, например, $[\exp(\text{Pi} \cdot \cos(x+a)), x=0..9]$. По умолчанию заданы пределы от -10 до 10 . К двум указанным опциям легко добавить кнопки, управляющие остальными семью.

Отметим, что опция `'`stop`'` заключена сначала в левые одиночные кавычки¹, так как слово `stop` уже задействовано при работе Maple в режиме командной строки (синонимы `quit` и `done`).

Пример 17

```
> with(Maplets[Elements]):
> with(plots):
> g1:=animate(plot, [exp(Pi*cos(x+a))], a=1..9):
> mplt:=Maplet([
>     [
>     Plotter[P1](g1,height=200, continuous=false),
>     [
>     Button("play",SetOption(P1('play')=true)),
>     Button("stop",SetOption(P1('`stop`')=true))
>     ]
>     ]
> ):
> Maplets:-Display(mplt);
```

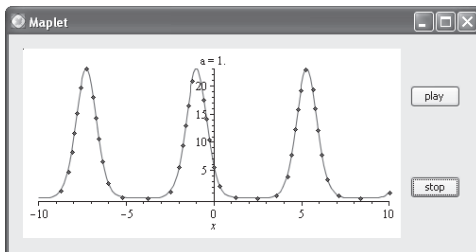


Рис. 391

При записи этого примера использована горизонтальная иерархия закрывающих и открывающих скобок.

См. также примеры использования `Plotter` на с. 424, 452, 344.

¹Или обратные кавычки. На клавиатуре на одной клавише с тильдой \sim .

6.14. MathMLViewer

Просмотр формул в формате MathML. Имеет описанные выше опции `background`, `foreground`, `height`, `reference`, `value`, `visible`, `width`, `wrapped`, а также следующие:

- `breakwidth` — ширина текста¹. Если `wrapped=true`, то этот параметр игнорируется. По умолчанию `breakwidth=400`.
- `fontsize` — размер шрифта. По умолчанию `fontsize=18`.

В следующем примере на экран выдается вторая производная функции $\arcsin(\ln(x))$.

Пример 18

```
> mplt := Maplet([MathMLViewer('value'=MathML[Export]
>                               (diff(arcsin(ln(x)), x$2)))]):
> Maplets:-Display(mplt);
```

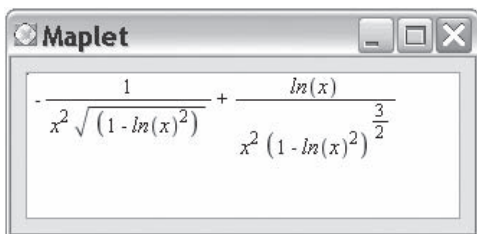


Рис. 392

Отметим, что есть еще элемент `MathMLEditor` — готовый редактор формул в формате MathML.

6.15. DropDownBox

Раскрывающийся вниз список выбора. Все его опции уже описаны: `background`, `enabled`, `font`, `foreground`, `halign`, `height`, `onchange`, `reference`, `tooltip`, `valign`, `value`, `visible`, `width`. Заметим только, что термин `onchange` здесь неудачен. Лучше было бы назвать его `oncheck`, так как в действительности он срабатывает при выборе, а не при изменении, который и сделать в заданном списке невозможно. Эта опция использована в следующем примере (рис. 393):

¹Эта опция, а также опции `visible` и `wrapped` появились в Maple 10.

Пример 19

```

> with(Maplets[Elements]):
> sp:=[20,24,21,54,55,81,87]:
> ev:=onchange=Evaluate('T'='D2'):
> mplt:=Maplet(["Выбор",DropDownBox['D2'](ev,sp)],
>             TextField['T'](6)):
> Maplets[Display](mplt);

```

Демонстрируется выбор одного из элементов списка и запись результата выбора в текстовое поле. Если список состоит из строковых значений `sp=["20", "24", "21", "54", "55", "81", "87"]`, то можно подсказать элемент, который будет стоять в еще не раскрывшемся списке (выбор по умолчанию): `(ev, "87", sp)`. Иначе там появляется первый элемент, т. е. 20. Для большей ясности структуры маплета список и действие задаются вне маплета и записываются во временные переменные `sp` и `ev`.



Рис. 393

6.16. ComboBox

То же, что и `DropDownBox`, но с возможностью редактировать список или дополнять его. Опции этих элементов совпадают, но теперь термин `onchange` вполне соответствует своему назначению. Пример можно взять предыдущий с простой заменой слова `DropDownBox` на `ComboBox` и поэкспериментировать с вводом.

6.17. ListBox

То же, что и `DropDownBox`, но список постоянно раскрыт. Приведем пример. Ставится задача выбора одной или нескольких

формул из предложенного списка и запись этих формул в файл "C:\выбор.txt". Выбор нескольких элементов выполняется как обычно — с помощью удерживаемой клавиши Ctrl, если нужно выбрать элементы в разных местах списка, или клавиши Shift, если выбираются последовательные элементы. Маплет состоит из трех элементов: текста t1, раскрытого списка lb и кнопки завершения работы b1. Оператором Shutdown кнопка Ok обрабатывает закрытие маплета с сохранением результата L1 элемента lb. Пусть из четырех предложенных формул пользователь выбрал формулу для количества движения $q = mv$ и выражение для момента инерции цилиндра $J = mR^2/2$ (рис. 394). Результат (с отброшенными квадратными скобками) представляет собой строку "q =mv, J = 1/2*m*R^2", в чем можно убедиться, выполнив команду проверки типа `whattype(op(res))`. Оператор `op` снимает с выражения скобки. Если есть необходимость разделить результаты на отдельные, надо применить команду `ListBoxSplit`. Получится последовательность (`exprseq`) из двух строк "q = mv", "J= 1/2*m*R^2". Этот результат и записывается в файл. Для этого командой `fopen` открываем файл для записи и одновременно назначаем ему имя `f`. Заметим, что вместо двойной обратной черты (backslash) в названии файла допустима дробная черта "C:/выбор2.txt". Для записи выбран формат вывода строки "% s."

Пример 20

```
> with(Maplets[Elements]):
> t1:="Выбор\n формулы":
> sp:=[q=mv, T=m*v*v/2, J=m*R*R/2, M=q1^2/8]:
> lb:=ListBox['L1'](sort(sp, length)):
> b1:=Button("Ok", Shutdown(['L1'])):
> maplet := Maplet([[t1, lb], b1]):
> res:= Maplets[Display](maplet);
> A:=Maplets[Tools][ListBoxSplit](res[1]);
> B:=convert(A, string):
> f:=fopen("C:\выбор.txt", WRITE):
> fprintf(f, "%s", B):
> fclose(f):
```

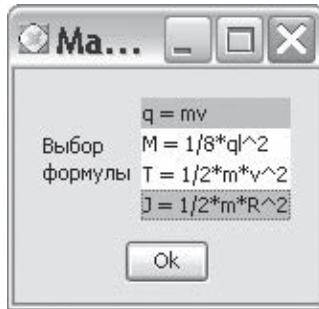


Рис. 394

Список в этой программе сортирован по длине элементов в порядке возрастания командой `sort` с опцией `length`. Это действие необязательное, а служит только для демонстрации этой удобной команды. Отметим также, что список состоял из алгебраических выражений. На входе и на выходе (в списке выбора) они не совпадают. Так произведение $R \cdot R$ превратилось в квадрат R^2 и т. д. **Maple** автоматически переводит алгебраические выражения в принятую форму.

Программу можно немного усложнить и ввести операцию выбора файла для записи или создания нового файла в стандартном окне проводника. Используем оператор `GetFile` из пакета `Maplets[Utilities]`. В начале программы загружаем это пакет `with(Maplets[Utilities])`, затем после обозначения клавиши `b1` и перед открытием маплета вставляем строку

```
> fz:=GetFile('title' = Файл,'directory'="C:/MSU",
>
> approvecheck=false):
```

Опция `approvecheck=false` указывает, что если в директории нет искомого файла, то можно его создать. Запись теперь будет иметь форму `f:=fopen(fz,WRITE)`. Оператор `GetFile` есть также в пакете `Maplets[Examples]`. Он немного проще: например, там не надо указывать опцию `approvecheck=false`.

6.18. Диалоги

Среди элементов маплетов отдельное место занимают диалоги. Всего их семь: `AlertDialog`, `ColorDialog`, `ConfirmDialog`, `FileDialog`, `InputDialog`, `MessageDialog`, `QuestionDialog`. Хотя назначение их ясно из названий, программирование диалогов не всегда очевидно, поэтому следует описать некоторые из них. Остальные работают аналогично.

- **MessageDialog** — сообщение, простейший диалог. По команде из основной программы (например, нажатие кнопки) выскакивает окно с сообщением и одной кнопкой ОК, закрывающей это сообщение. Если указать тип сообщения, например `type='error'`, то в левом углу появится еще и соответствующий значок. Всего четыре типа: `'error'` (красный крестик), `information` (буква i), `plain` (по умолчанию, без значка), `warning` (знак !). Тип `'error'` непременно ставится в обратных кавычках. Есть также уже описанные опции `caption`, `height`, `reference`, `resizable`, `title`, `width` и опция `onapprove`, указывающая на действие `Action` или команду.

В следующем примере окно сообщений запускается командой¹ `RunWindow`.

Пример 21

```
> with(Maplets[Elements]):
> b:=MessageDialog['M'](title="Подсказка",
>     "Маплеты: http://www.tspu.tula.ru"):
> mplt:= Maplet(Window(title="Нажми кнопку F1",
>     height=100, [Button("F1",RunDialog('M'))]),b):
> Maplets[Display](mplt);
```

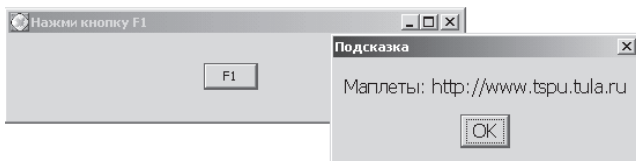


Рис. 395

Это простейший вариант такой программы. Можно немного усложнить, если ввести оператор `Action`. В этом случае перед второй строкой `b:=` надо сначала обозначить

```
> a:=Action['A'](RunDialog('M')):
    Описание маплета будет выглядеть так:
> mplt:= Maplet(Window(title="Нажми кнопку F1",
>     height=100, [Button("F1", onclick='A')]), a,b):
    Результат будет тем же.
```

¹Другие команды в маплетах: `CloseWindow`, `Evaluate`, `SetOption`, `Shutdown`.

- `ColorDialog` — выбор цвета. Приведем пример программы, в которой в поле ввода заносится некоторая функция и при нажатии кнопки `Plot` строится график (рис. 396). В программе предусмотрено изменение цвета кривой (кнопка `Color`).

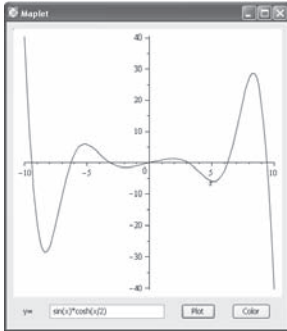


Рис. 396

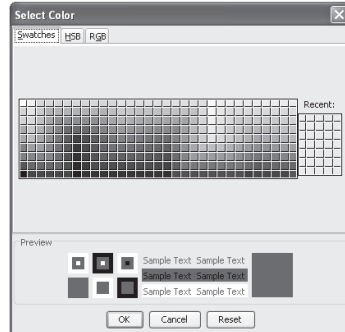


Рис. 397

Оператор `Get` передает содержимое поля ввода `'Y'` в переменную `'y'`. При этом оператор `parse` снимает кавычки с текста. Например, `"sin(x)"` превращается в `sin(x)`. Переменная `'y'` является локальной в процедуре **ВыборЦвета**, в которой выбирается цвет и происходит построение графика. В операторе `plot` не указан предел

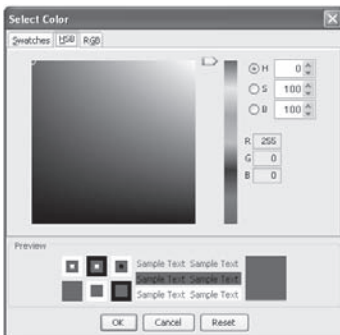


Рис. 398

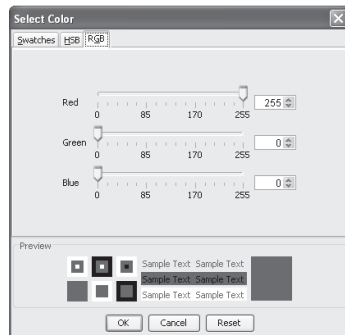


Рис. 399

изменения `x`, по умолчанию он принят от `-10` до `+10`. Цвет также выбран по умолчанию. Но здесь надо уточнить, в каких случаях что умалчивается. В первом случае, когда пользователь строит график,

не нажимает предварительно кнопку `Color`, переменная цвета будет иметь значение " ", не воспринимаемое оператором `plot`, поэтому условный оператор назначает значение `blue`. Во втором случае пользователь начинает выбирать цвет, но при выходе из процедуры не фиксирует свой выбор, а нажимает кнопку `Cancel` или `Reset`. Тогда срабатывает опция `value=red` оператора `ColorDialog`. В самом операторе выбора цвета встроены три способа окраски `Swatches` (образцы, рис. 397), `HSB` (Нюе — оттенок, Saturation — насыщенность, Brightness — яркость, рис. 398), `RGB` (Red, Green, Blue, рис. 399).

Пример 22

```
> ВыборЦвета:=proc() local цвет, y;
> use Maplelets[Tools] in
> y:=parse(Get('Y')); цвет:=Get(CD1);
> end use;
> if цвет="" then цвет:=blue fi; # Цвет по умолч.
> plot(y, color=цвет);
> end proc:
> with(Maplets[Elements]):
> y1:="y=", TextField['Y'](20);
> b1:=Button("Plot", Evaluate('function' =
>         'ВыборЦвета', 'target' = 'P1')):
> b2:=Button("Color", RunDialog('dialog'='CD1')):
> График:= Maplet(
>     Window([Plotter[P1](), [y1, b1, b2]]),
> ColorDialog[CD1](value=red)
>     ):
> Maplelets[Display](График);
```

- `InputDialog` — ввод данных. Вводить данные достаточно просто и в обычном поле ввода. Это продемонстрировано в маплетах на с. 313, 345 и др. Однако чтобы не загромождать окно многочисленными полями ввода или просто чтобы подчеркнуть важность каких-нибудь входных данных, ввод организуют в отдельное окно. В **Maple** это окно имеет поле ввода и две кнопки `OK` и `Cancel`. В следующем примере, демонстрирующем операцию возведения в квадрат, нажатие кнопки со знаком вопроса (рис. 400) открывает окно ввода. Маплет назван «Ввод», имеет три элемента: `a` — текстовое поле, `b` — кнопка и `c` — диалог ввода. Информация введенная в `c` (если нажата кнопка `OK`, т. е. режим `onapprove`) переносится в поле `a` с помощью процедуры `d`. Если закрыть диалог ввода клавишей `Cancel`, то поле `a`, имеющее имя `Y`, очищается. Это происходит

присвоением пустой строки `value = ""`. Оператор `parse` снимает со строки кавычки. Функцию (процедуру) `d` следует заменить на какую-нибудь свою, результат будет помещен по адресу, указанному в опции `target`.

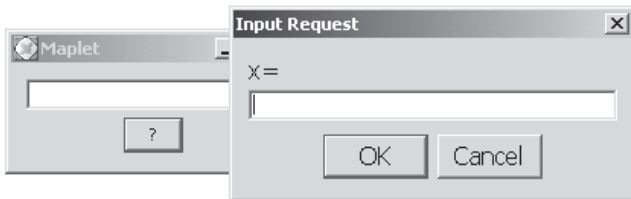


Рис. 400

Заметим, что в этом примере со всех имен сняты очищающие кавычки, т. е. вместо обычной записи `'target' = 'Y'` имеем `target = Y`, и вместо `'M'` просто `M` и т. д. Так можно делать, если нет риска конфликта между одноименными переменными. В какой-нибудь большой программе можно не заметить присвоения, например, `M:=11`, после чего называть этой же буквой окно диалога уже, конечно, будет нельзя, а в «очищенном» виде `'M'` — можно.

Пример 23

```
> with(Maplets[Elements]):
> with(Maplets[Tools]):
> a:=TextField[Y]():
> b:=Button("?", RunDialog(dialog = M)):
> d:=proc() parse(Get(M))^2; end:
> c:=InputDialog[M]("x=",
>     onapprove=Evaluate(function=d,target =Y),
>     oncancel=SetOption(target = Y,value = "")):
> Ввод:= Maplet(Window([a,b]),c):
> Maplets[Display](Ввод);
```

- `FileDialog` — аналог вызова проводника Windows. Получение имени файла. Имеет ранее описанные опции `height`, `onapprove`, `oncancel`, `reference`, `resizable`, `title`, `value`, `width`, а также:

– `approvecaption`. Просто текст, который можно поместить вместо слова `Open` в диалоге открытия файла;

- `closeonapprove`. По умолчанию `closeonapprove= true`, и диалог сам закрывается после выбора файла. Если выбрать `false`, то диалог надо закрывать отдельно;
- `filefilter`. Расширение файла. По умолчанию просматриваются все файлы, что эквивалентно `filefilter="*"`. Если требуются файлы с непустым расширением, то надо писать `filefilter="*. *"`;
- `directory`. Каталог, где происходит поиск файла;
- `filename`. Заранее выбранное имя открываемого файла;
- `fileselectionmode`. Три значения: `filesonly` (только файлы), `directoriesonly` (только каталоги), `filesanddirectories` (и файлы, и каталоги);
- `filterdescription`. Просто описание (в произвольной форме) искомым файлов. По умолчанию `All Files`. Эта надпись появляется в диалоге открытия в стандартном для Проводника месте.

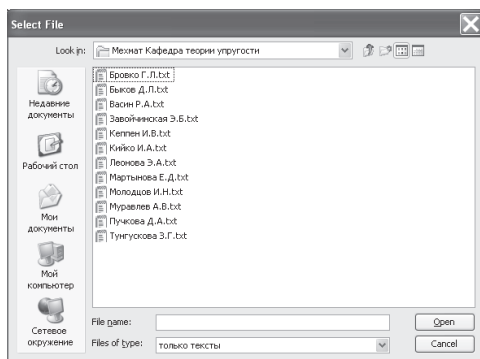


Рис. 401

В следующем примере продемонстрировано открытие файла, чтение файла и запись содержимого файла в поле `TextBox`. Открывает файл диалог `FileDialog` (рис. 401), читает и записывает прочтенное специальная процедура под именем `откр`. В конце этой процедуры (фактически — ее результат) можно было бы поставить просто `текст`, однако в этом случае будут видны окружающие кавычки и символы переноса строки `\n`. Поэтому использовано конвертирование в `symbol`. Шрифт в `TextBox` выбран покрупнее, без указания его семейства и стиля¹. Полностью шрифт задается так: `Font("Helvetica", italic, 12)`.

¹С. 265.

В данном примере результат не уместился в `TextBox`, в котором было прописано `height=6` — шесть строк, поэтому появилась полоса прокрутки (рис. 402). В операторе `fopen(Get(F),READ)` не указан тип файла `TEXT`, этот тип тут берется по умолчанию, а вот в операторе `readbytes` это надо было сделать обязательно. Кроме этого, тут же следует указать длину считываемого файла (в символах), иначе будет прочтен только один первый символ.

Пример 24

```
> with(Maplets[Elements]):
> with(Maplets[Tools]):
> a:=TextBox['Y'](height=6,font=Font(12)):
> b:=Button("File", RunDialog(dialog = 'F')):
> откр:=proc() local fd, текст;
>   fd:=fopen(Get(F),READ);
>   текст:=readbytes(fd,123,TEXT):
>   fclose(fd); convert(текст,symbol);
> end proc:
> c:=FileDialog['F']('filefilter' = "*.txt",
> filterdescription="только тексты",
> directory="C:\\Мехмат Кафедра теории упругости",
> onapprove=Evaluate(function=откр,target='Y'),
>                                     oncancel = Shutdown()):
> Чтение:= Maplet(Window([a,b]),c):
> Maplets[Display](Чтение);
```

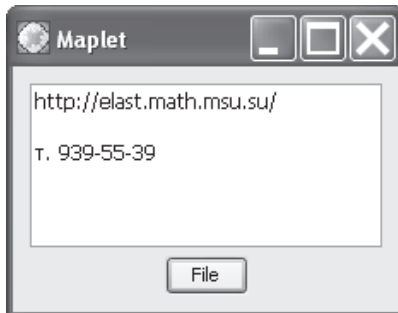


Рис. 402

- **AlertDialog** — диалог предупреждения (рис. 403). Имеет две кнопки OK и Cancel и опции `caption`, `height`, `onapprove`, `oncancel`, `reference`, `resizable`, `title`, `width`

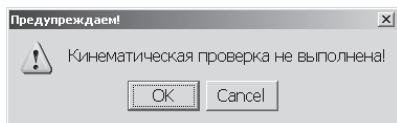


Рис. 403

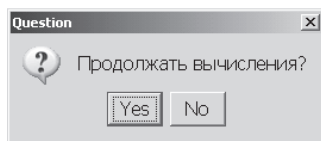


Рис. 404

- **ConfirmDialog** — диалог подтверждения. Имеет три кнопки Yes, No и Cancel и те же опции, что и **AlertDialog**, и еще опцию `type` с четырьмя значениями `'error'`, `plain`, `question`, `warning`, определяющими значок¹ в окне, и опцию `ondecline`. Последняя опция соответствует действию **Action** при нажатии кнопки No. Опция `onapprove`, как и раньше, вызывается при нажатии кнопки Yes, опция `oncancel` — кнопки Cancel.
- **QuestionDialog**. Вопрос. Имеет две кнопки Yes, No и значок вопроса (рис. 404). Опции: `caption`, `height`, `onapprove`, `ondecline`, `reference`, `resizable`, `title`, `width`.

6.19. Maplet Builder

Визуальное программирование широко распространено в большинстве алгоритмических языков. Не отстает здесь и **Maple**. С помощью **Maplet Builder** можно легко создавать маплеты, перетаскивая мышью на поле маплета элементы и назначая им свойства и действия.

Создадим в **Maplet Builder** простейший маплет, состоящий из кнопки и текстового поля. При нажатии кнопки в поле появляется слово Hello. Последовательно выполним следующие действия:

1. Запустим **Standard Worksheet** `maplew.exe`.
2. Из меню программы запустим **Maplet Builder**. Последовательность выбора команд меню: **Tools**→**Assistants**→**Maplet Builder**.
3. Перетаскиваем левой клавишей мыши с палитры **Body** (слева сверху²) на поле маплета элемент `Button1`. Процедура перетаскивания отличается от традиционного способа, ведущего начало от **Delphi**. Здесь отпустить клавишу мыши надо уже в точке назначения. Кнопка

¹с. 329.

²Всего семь палитр с элементами: **Body**, **Command**, **Layout**, **Other**, **ToolBar**, **Menu**, **Dialog**. Раскрыть палитру или закрыть (для экономии места на экране) можно, щелкнув по треугольнику палитры в левом углу.

автоматически центрируется и получает имя (и подпись) Button1. Первоначально поле маплета есть элемент VBoxColumn1.

4. С этой же палитры перетаскиваем элемент TextField. Он выстраивается к кнопке Button1 в столбец и получает имя TextField1. Меняем его в поле reference свойств (справа, см. рис. 405.)

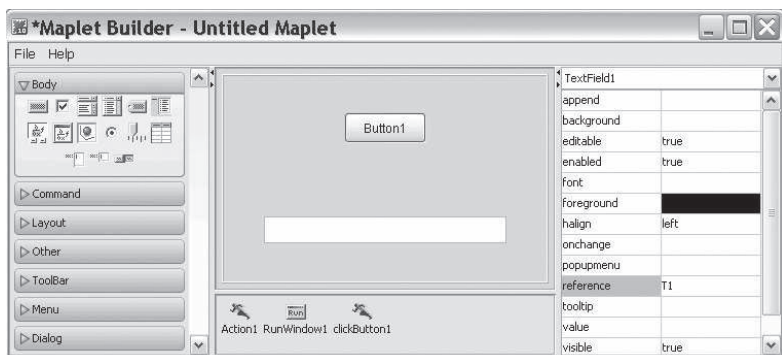


Рис. 405

5. Возвращаемся к кнопке Button1. Щелкаем по ней, активируя ее свойства справа экрана. Меняем в поле caption надпись на Ok. Имя (reference) кнопки оставляем прежним — Button1. Одновременно назначаем ей действие, щелкая по полю onclick. В выпадающем меню выбираем <Evaluate> (рис. 406).

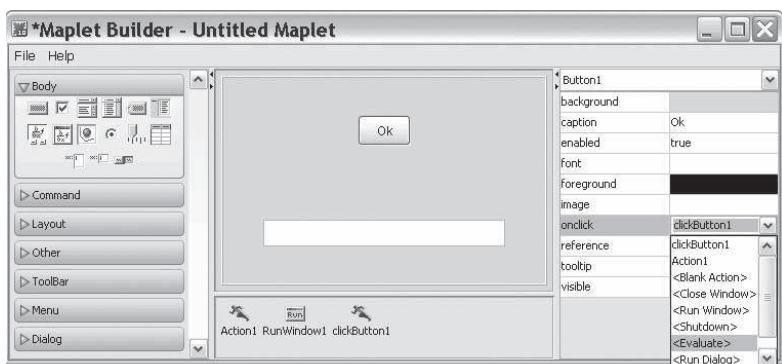


Рис. 406

6. С щелчком по полю <Evaluate> открывается окно (рис. 407), в котором уже автоматически записано имя элемента текстового поля T1

(других элементов, с которыми что-то можно сделать, в проекте пока нет). Прописываем в поле Expression слово Hello.

7. Сохраняем маплет: File→ Save (в меню Maplet Builder).

8. Запускаем маплет: File→ Run. Это действие можно и опустить, так как маплет, как почти самостоятельная единица, уже готов и его можно запустить, выйдя из Maplet Builder и Maple.

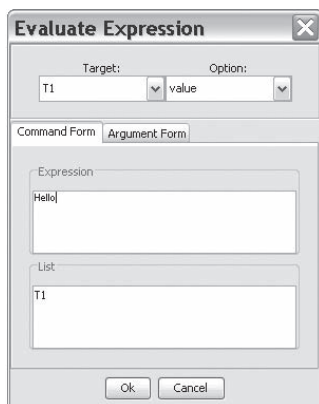


Рис. 407

Если маплеты, созданные вручную в классическом варианте **Classic Worksheet** с помощью исполняемого файла `swtmaple.exe`, не перенесли только букву **я** и пропускали букву **ю**, то Maplet Builder не воспринимает уже буквы **ю**, **я**, **Я**. При этом появляется надпись

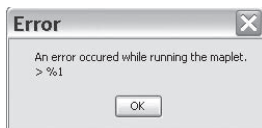


Рис. 408

Глава 7

MAPLE-ПРОГРАММЫ

Приведенные ниже программы можно условно разделить на вычислительные и алгоритмические. Первые, простейшие, помогают вычислить результат, руководствуясь известными формулами и уравнениями. В этих программах **Maple** выступает в роли большого (и удобного!) калькулятора, дающего возможность считать и при необходимости легко пересчитывать результат при других данных или при обнаружении ошибки. Как правило, задачи для этих программ принципиально несложные.

Сложные задачи, например статически неопределимые системы, могут быть решены по стандартной схеме. Однако из-за исключительно громоздкого математического аппарата, требующего повышенного внимания и большого напряжения, такие задачи решить несколько раз трудно, а это приходится делать при поиске почти неизбежных ошибок. В этом случае на помощь приходят алгоритмические программы, служащие в основном для проверки решения. В этих программах, как правило, используется план решения, отличный от того, который дается в процессе обучения (вычисления вручную). Безусловно, студент должен овладеть классическими методами решения, раскрывающими физическую суть задачи. Однако в современных условиях нельзя пренебрегать и так называемыми пользовательскими программами, работающими по принципу «черного ящика». Задаем информацию на входе программы, нажимаем ENTER и получаем результат. В **Maple** (как и в других математических пакетах) принцип «черного ящика» привычно используется, например, в операторе `solve` для решения алгебраических уравнений и `dsolve` для решения дифференциальных уравнений. Вероятно, придет время и будут созданы такого же рода операторы для расчета фермы, балки или других конструкций. А пока воспользуемся предлагаемыми программами. Тексты программ размещены по адресу <http://vuz.exponenta.ru/progr.rar>.

7.1. Рама

Приведем решение задачи 1 на с. 12 о нахождении реакций опор рамы. Программа начинается с необязательной команды `restart`, очищающей память программы и значения всех переменных, назначенные ранее. Это особенно важно при повторном выполнении программы или

при одновременной работе с несколькими документами, когда значения переменных в одной программе могут повлиять на работу другой. Если начинать программу с этой команды, то можно избежать многих недоразумений, которые часто относят к "ошибкам Maple".

Значения тригонометрических функций угла α обозначены как $\sin\alpha$ и $\cos\alpha$ (не путать с самими функциями $\sin(\alpha)$, $\cos(\alpha)$!), уравнения равновесия — $eq1$, $eq2$, $eq3$. Неизвестные реакции можно было бы обозначить как переменные с индексом, например $X[a]$, или, наоборот, не меняя регистра: x_a , для ускорения набора. В небольших программах это несущественно.

Решение системы уравнений выполняет оператор `solve`. Особенность этого оператора состоит в том, что он находит решения, но не присваивает их искомым переменным. Для этого используется оператор `assign`. Найденные реакции необходимы для проверки решения. В качестве проверки составляется еще одно уравнение равновесия — сумма моментов относительно точки K (рис. 5, с. 12).

Определить реакции опор в этой раме и построить эпюры моментов и перезывающих сил можно также по программе 41 (файл данных `p1.txt` в архиве <http://vuz.exponenta.ru/progr.rar>).

Программа 3 (P3.mws)

```
> restart;
  Дано
> sina:=0.6: cosa:=0.8: F:=10: M:=34: P:=2:
> eq1:=Xa-F*cosa+P: # Уравнение проекций на x
> eq2:=Ya+Yb+F*sina: # Уравнение проекций на y
> eq3:=2*Yb-2*F*cosa+8*F*sina-M-3*P:# Уравн.мом.отн.А
> r:=solve({eq1,eq2,eq3},{Xa,Yb,Ya});
> assign(r);
      r := { Yb = 4., Xa = 6., Ya = -10.}
  Проверка
> -Ya*8-6*Yb+3*Xa-5*F*cosa-M;
      0.
```

7.2. Составная конструкция

В программе решена задача 7, с. 23. Записаны значения нагрузок, уравнения равновесия отдельных частей (правые части по умолчанию равны нулю). Массив нагрузок удобно вводить списком: $F:=2, 4, 2$. Решение системы 11 уравнений выполняет оператор `solve`. Для выполнения проверки необходимо, чтобы решение было бы присвоено неизвестным. Автоматически это в Maple не происходит. Если

неизвестная x одна в уравнении $2x+5=0$, то это делается просто: $x:=\text{solve}(2x+5, x)$. В общем случае решение присваивается неизвестным с помощью оператора `assign`. В том случае, когда в системе уравнений все коэффициенты численные, нет необходимости указывать неизвестные в операторе `solve` — они выделяются автоматически. В нашем случае можно избежать длинного перечисления неизвестных и просто написать $R:=\text{solve}(\{eq[i]\$i=1..11\})$. Это уберегает от ошибок ввода. Очень часто пользователь не может получить решение, казалось бы, очевидной системы только потому, что неправильно указаны неизвестные: например, неизвестная введена не в том регистре или одна из букв заменена на схожую из другого алфавита.

Программа 4 (P4.mws)

```
> restart;
> F:=2,4,2: M:=3:#Силы и момент
  Равновесие AD
> eq[1]:=X2+Xa:
> eq[2]:=Y2+Ya:
> eq[3]:=8*Ya-3*Xa-M:
  Равновесие DC
> eq[4]:=X1+F[1]-S+Xc:
> eq[5]:=Y1-F[2]:
> eq[6]:=4*Xc-2*S-8*F[2]:
  Равновесие BD
> eq[7]:=X3+S-F[3]:
> eq[8]:=Y3+Yb:
> eq[9]:=-4*X3-2*S+2*F[3]:
  Равновесие оси
> eq[10]:=X1+X2+X3:
> eq[11]:=Y1+Y2+Y3:
  Решение
> R:=solve({eq[i]\$i=1..11},
  {X1,X2,X3,Y1,Y2,Y3,Xa,Ya,S,Yb,Xc}); assign(R);
  R:={ Y1 = 4, X3 = 0, Ya = -3, Yb = 7,
  Xa = -9, Xc = 9, X1 = -9, S = 2, Y3 = -7, Y2 = 3, X2 = 9}
  Проверка
> -Xa*3-Ya*3+F[2]*3-F[3]*2+Xc*4-Yb*11-M;
```


7.3. Система с односторонними связями

Рассмотрим решение задачи 8, с. 25. В программе записаны уравнения равновесия левой и правой частей рамы и найдено решение в зависимости от усилия натяжения нити S_1 . Построен график зависимости усилия натяжения нити S_2 от S_1 . Очевидно, решение справедливо для положительных усилий S_1 и S_2 . С помощью оператора `minimize` найдено минимальное значение реакции в шарнире. Из приведенного графика ясно, что минимум попадает в допустимую область изменения S_1 и S_2 .

Оператор `minimize` с опцией `location` не только определяет минимальное значение, но и дает само значение S_1 , соответствующее этому минимуму, т. е. как раз то, что в этой задаче необходимо. Для проверки находим S_1 стандартными методами. Из уравнения $dR_C/dS_1 = 0$ определяем S_1 и вычисляем все реакции конструкции.

На графике изображены зависимости от S_1 усилия S_2 и реакции опоры R_C (1.35), (1.36), с. 27. Парабола $R_C(S_1)$ достигает минимума при $S_1 = 25/9$ кН. В пересечении этих графиков искать особый смысл не надо — просто при таком значении S_1 реакция R_C равна усилию S_2 .

Две кривые на графике отличаются толщиной линии и стилем. Толщина задается опцией `thickness`, а параметр стиля — опцией `linestyle`, имеющей значения `solid`, `dot`, `dash`, `dashdot`, `longdash`, `spacedash`, `spacedot`. Стиль линии можно задать и числом от 1 до 7. Например, сплошная линия соответствует `linestyle=1`.

Отметим особенность оператора `minimize`. Если не указать в нашем случае команду упрощения `simplify`, то вместо ответа 2 получим результат в виде формулы $\sqrt{900}/15$. Оператор `minimize` дает ответ в виде списка из двух элементов, поэтому команда `simplify` применена только ко второму элементу списка. Извлечь из ответа значение S_1 можно, последовательно применяя оператор снятия скобок `or`.

Программа 5 (P5.mws)

```

> restart;
Дано
> F:=4: M:=2: a:=arctan(3/4):
Уравнения равновесия
> eq[1]:=Xc+S2+S1*cos(a)-F:
> eq[2]:=Ya+Yc-S1*sin(a):
> eq[3]:=-Ya*6-F*2+S2*3+S1*sin(a)*4:
> eq[4]:=Xb-Xc-S2-S1*cos(a):
> eq[6]:=Yb-Yc+S1*sin(a):
> eq[5]:=Xc*5+S2*2+S1*cos(a)*2-M:

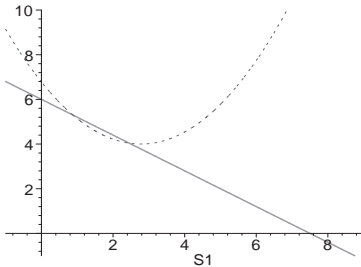
```

Решение

```

> R:=solve({seq(eq[i], i=1..6)}, {S2, Xc, Yc, Ya, Xb, Yb});
> assign(R):
R := {Xb = 4, Yb = -5/3, S2 = 6 - 4*S1/5, Xc = -2, Ya = 5/3, Yc = -5/3 + 3*S1/5}
> Rc:=Xc^2+Yc^2:# Реакция шарнира
    Минимум реакции
> simplify(minimize(sqrt(Rc), location)[2]);
      {{{S1 = 25/9}, 2}}
    Графики зависимостей усилий от натяжения S1
> plot([Rc, S2], S1=-1..9, -1..10,
>      thickness=[2, 4], linestyle=[dash, solid]);

```



```

> S1:=solve(diff(Rc, S1), S1);
      S1 := 25/9
> s2=S2, xc=Xc, yc=Yc, ya=Ya, xb=Xb, yb=Yb; #Ответы
      s2 = 34/9, xc = -2, yc = 0, ya = 5/3, xb = 4, yb = -5/3

```

7.4. Ферма. Маклет

Рассмотрим решение задачи 11 о ферме на с. 32. Пронумеруем узлы и стержни. Номера узлов отметим рамкой (рис. 409). Опоры моделируем стержнями, подвижную опору A — стержнем 9, неподвижную B — двумя стержнями стержнем 10 и 8. Все стержни фермы (включая опорные) условно представляем векторами, направление которых выбираем произвольно. Эти векторы не связаны с усилиями в стержнях, и решение не зависит от выбора их направления. В программу вводятся концы стержней в виде векторов-строк N_{beg} , N_{end} .

В программе используется метод вырезания узлов¹. Опорные стержни-векторы должны быть направлены от фермы к неподвижным узлам.

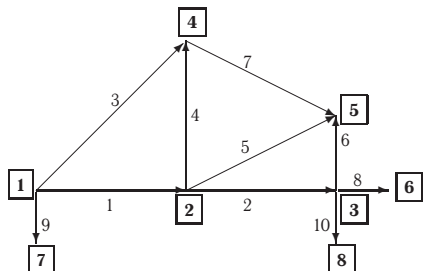


Рис. 409

Кроме того, опорные шарниры нумеруются последними. Последние условия существенные и невыполнение их часто является причиной ошибок ввода. Связано это с условным оператором $\text{Nend}[i] \leq N-K$, работающем в цикле по числу стержней. Усилия во всех стержнях, кроме опорных, входят в систему дважды —

в уравнение равновесия узла-начала и в уравнение узла-конца стержня. Соответствующие направляющие косинусы имеют противоположные знаки. Разрешающая матрица размером $M \times M$ системы содержит направляющие косинусы стержней.

Вводится вектор свободных членов системы \mathbf{V} . В него заносится информация о нагрузках. Горизонтальные и вертикальные нагрузки образуют отдельные списки \mathbf{F}_x и \mathbf{F}_y . Даже тогда, когда нагрузка единственная, как в данном случае, ее значение оформляется в виде двух списков длиной 1. Длина списка (число векторов сил) вычисляется оператором `ops`. Списки \mathbf{F}_x и \mathbf{F}_y должны иметь одинаковую длину.

Решение системы получается с помощью оператора `LinearSolve` пакета `LinearAlgebra`. Но этот способ решения системы линейных уравнений в `Maple` не единственный. Можно, например, найти обратную матрицу G^{-1} , а затем умножить ее на вектор свободных членов: $\mathbf{S} := -G^{-1} \cdot \mathbf{V}$. Здесь точка — знак умножения матрицы на матрицу или матрицы на вектор. При этом не требуется загружать пакет `LinearSolve`. Обратная матрица в `Maple` может быть также найдена с помощью простого деления: $1/G$.

Программу оформим в виде маплета. Общая структура программы следующая:

```
FermaModule:=module()
  Frm:=proc()
    end proc;

  runFerma:=proc()
    end proc;
end module:
FermaModule:-runFerma():
```

¹ Алгоритм Горячева В. Н., Резунова А. В. (ВГАСУ, Воронеж).

В целом программа записана в модуль `FermaModule` с двумя процедурами `Frm` и `runFerma`. В первой производятся вычисления, во второй организован интерфейс программы (рис. 410). Программа

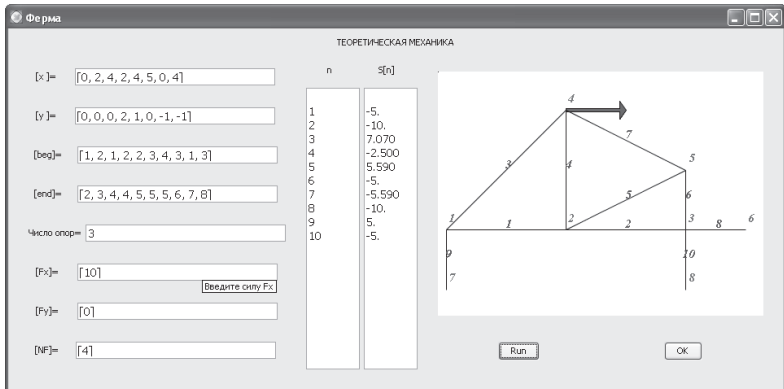


Рис. 410. P6.maplet

запускается вызовом процедуры `runFerma` из модуля `FermaModule`. Опишем структуру каждой процедуры. Схема процедуры вычислений `Frm` имеет вид

```
Frm:=proc()
use Maplets[Tools] in
```

- ввод данных оператором `Get`;
- изображение фермы `Ris`, состоящее из стержней $R[i]$, $i=1..M$, нагрузок iF , составленное из стрелок `arrow`, шарниров `Шарнир[i]`, $i=1..N$;
- заполнение матрицы направляющих косинусов `G`;
- заполнение вектора правых частей `B` (нечетные строки `B` — горизонтальная нагрузка, четные — вертикальная);
- получение решения `S` системы оператором `LinearSolve`;
- занесение решения в таблицу оператором `Set`;

```
end use;
Ris:
end proc;
```

Последним в процедуре вызван рисунок `Ris`, поэтому при обращении `display(Frm())` в процедуре интерфейса на дисплее появится изображение фермы¹.

Опишем структуру процедуры `runFerma`, задающей интерфейс программы:

```
runFerma:=proc()
  переменные; константы;
  use Maplelets:-Elements in
  обозначения элементов интерфейса (a,c,T,p,mplt);
  mplt:= Maplelet(Evaluate(),Window());
  Maplelets:-Display(mplt);
  end use;
end proc;
```

Программа 6 (P6.mws)

```
> restart;with(LinearAlgebra):
> with(plots): with(plottools):
> FermaModule:=module()
> export Frm,runFerma:
  Счет и изображение фермы
> Frm:=proc()
> local F,Nbeg,Nend,K,x,y,fd,N,m,M,Шрифт,i,Шарнир,
> Симв,Стержень,Вид,nF,R,Ris,iF,G,Lx,Ly,L,S,B:
> use Maplelets[Tools] in
> x:=Get('Tx'::list); # Координаты
> y:=Get('Ty'::list); # Координаты
> F[1]:=Get('TFx'::list); # Fx
> F[2]:=Get('TFy'::list); # Fy
> nF:=Get('TnF'::list); # Нагруженные узлы
> Nbeg:=Get('TNbeg'::list); # Начало стержн
> Nend:=Get('TNend'::list); # Конец стержн
> K:=Get('TNopr'::integer); # Число опор
> m:=nops(F[1]); # Число нагрузок
> N:=nops(x); # Число узлов
> M:=nops(Nend); # Число стержней
> Шрифт:=FONT(TIMES,BOLD,12):
```

¹Рекомендации по выводу на экран двух изображений см. на с. 423.

```

> for i to M do
>   R[i]:=PLOT(CURVES([[x[Nbeg[i]],y[Nbeg[i]]],
>                                     [x[Nend[i]],y[Nend[i]]]])):
> od:
> Симв:=proc(x) convert(x,symbol) end:
> for i to N do
>   Шарнир[i]:=PLOT(ТЕХТ([x[i]+0.1,y[i]+0.2],Симв(i)),
>                                     Шрифт,COLOR(HUE,1)):
> od:
> for i to M do
>   Стержень[i]:=PLOT(ТЕХТ(
> [(x[Nbeg[i]]+x[Nend[i]])/2+0.05,
> (y[Nbeg[i]]+y[Nend[i]])/2+0.1],
> Симв(i)),Шрифт,COLOR(HUE,0.7)):
> od:
  Параметры стрелки на рисунке
> Вид:=0.05,0.4,0.2,color=red:
  Изображение нагрузок
> iF:=seq(arrow([x[nF[i]],y[nF[i]]],
>               evalm([F[1][i],F[2][i]]*0.1),Вид),i=1..m):
  Изображение фермы
> Ris:=display(iF,seq(R[i],i=1..M),
>               seq(Шарнир[i],i=1..N),
>               seq(Стержень[i],i=1..M),
>               scaling=constrained,axes=none);
  Заполнение матрицы
> G:=Matrix(M,M):
> for i to M do
>   Lx:=x[Nend[i]]-x[Nbeg[i]]:
>   Ly:=y[Nend[i]]-y[Nbeg[i]]:
>   L[i]:=sqrt(Lx^2+Ly^2):
>   G[2*Nbeg[i]-1,i]:=Lx/L[i]:
>   G[2*Nbeg[i],i]:=Ly/L[i]:
>   if Nend[i]<=N-K then G[2*Nend[i]-1,i]:=-Lx/L[i]:
>                       G[2*Nend[i],i]:=-Ly/L[i]:fi:
> od:
  Прав. часть системы. Вектор нагрузок
> B:=Vector(M):

```

```

> for i to m do
>   B[2*nF[i]-1]:=F[1][i]: B[2*nF[i]]:=F[2][i]:
> od:
> S := LinearSolve(Matrix(M,G),-B):
> Set (t1=" \n",t2=" \n"):
> for i to M do
>   Set(t1(appendline)=convert(i,string)):
>   Set(t2(appendline)=convert(evalf(S[i],4),string)):
> od:
> end use:
> Ris:
> end proc:

```

Процедура ввода и вывода (интерфейс программы)

```

> runFerma:=proc()
> local a,c1,c2,p1,mplt,z,T1,T2;
> use Maplets:-Elements in
> шрифт:='font'=Font(14):
> z:='width'=5,'height'=25,'editable'='false',шрифт:
> T1:=TextBox['t1'](z):
> T2:=TextBox['t2'](z):
> a[1]:=["[x]=",TextField['Tx'](шрифт,
>   'value'=[0,2,4,2,4,5,0,4])]:
> a[2]:=["[y]=",TextField['Ty'](шрифт,
>   'value'=[0,0,0,2,1,0,-1,-1])]:
> a[3]:=["[beg]=",TextField['TNbeg'](шрифт,
>   'value'=[1,2,1,2,2,3,4,3,1,3])]:
> a[4]:=["[end]=",TextField['TNend'](шрифт,
>   'value'=[2,3,4,4,5,5,5,6,7,8])]:
> a[5]:=["[Опоры]=",TextField['TNopr'](шрифт,
>   'value'=3,'tooltip'="Число опорных узлов")]:
> a[6]:=["[Fx]=",TextField['TFx'](шрифт,
>   'value'=[10],'tooltip'="Введите силу Fx")]:
> a[7]:=["[Fy]=",TextField['TFy'](шрифт,
>   'value'=[0],'tooltip'="Введите силу Fy")]:
> a[8]:=["[NF]=",TextField['TNf'](шрифт,
>   'value'=[4],'tooltip'="Загруженные узлы")]:
> c1:=Button("Run",Evaluate('F'='display(Frm())')):

```

```

> c2:=Button("OK",Shutdown(['F'])):
> p1:=Plotter['F']():
> mplt:= Maplet(
>     Evaluate(function='Frm()'),
>     Window('title'="Ферма",
>         ["ТЕОРЕТИЧЕСКАЯ МЕХАНИКА",
>         [BoxColumn( seq(a[i],i=1..8)),
>         [ ["n","S[n]"], [T1,T2] ],[ [p1],[c1,c2] ]]])
>     ):#Maplet
> Maplets:-Display(mplt):
> end use;
> end proc;
> end module:
> FermaModule:-runFerma():

```

Приведем решение задачи 14 на с. 39. Подвижная опора C моделируется вертикальным стержнем. Его длина не влияет на решение. Число опорных узлов, равновесие которых не рассматривается (значение K), здесь четыре.

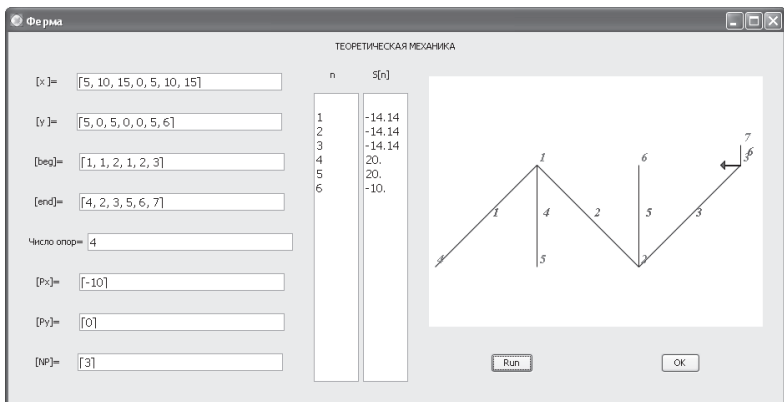


Рис. 411. 14.maplet

Опыт работы с этим маплетом показывает, что ввод данных является его слабым звеном. Ошибка при вводе, например, в числе введенных координат, приводит к появлению сообщения *invalid subscript selector*, а сообщение *Matrix index out of range* появляется, если занижено число опор. Другой недостаток — невозможность

сохранить введенные данные. Но тут есть простой способ — ввести данные не в маплете, а в его тексте, открыв файл `R6.maplet`¹ программой Блокнот, как обычный текстовый файл. Последний недостаток может быть поводом для самостоятельного совершенствования программы. Для этого в **Maple** есть разнообразные средства. Прочитать данные о высотах, записанных в файл `l.txt` на диске `C` в виде строки `0 0 0 2 1 0 -1 -1` во временную переменную `Y0`, можно оператором `Y0:=op(readdata("C:\\1.txt",float,9))`, а вставить эти данные в поле ввода (с возможностью дальнейшей коррекции в интерактивном режиме) надо так: `a[2]:=["[Y]=",TextField['Ty']('value'=Y0)]`, см. с. 347. Для записи данных есть оператор `writedata`.

7.5. Полка

Решим задачу 17 на с. 46. Сначала зададим размеры полки и положения опор. Каждая из опор в этой задаче представляет собой один или несколько опорных стержней. Сферический шарнир A фактически состоит из трех взаимно перпендикулярных стержней. Стержни введем как единичные векторы с направлениями $S[k]$, $k = 1, \dots, 6$ и точками приложения, радиус-векторы которых обозначим $\text{Коорд}[k]$, $k = 1, \dots, 6$. Начало координат может быть в произвольном месте, поместим его в опору A . Для того чтобы вектор $\text{Vector}([-a, -b, -c])$, лежащий на большой диагонали (подпорка), сделать единичным, применим операцию нормировки `Normalize` с опцией `Euclidean`. Значение этой константы равно 2. По умолчанию выбирается норма 1 или `infinity`, в которой $|\vec{V}| = \max(|v_k|)$, $k = 1, 2, 3$.

Матрицу системы уравнений равновесия формируем из двух частей. Первые три строки соответствуют уравнениям проекций. Направляющие косинусы усилий (проекции единичных направляющих векторов) записаны в матрицу $A1$. В матрицу $A2$ записаны проекции векторных произведений $\text{Коорд}[i]$ и $S[i]$, задающих векторы моментов $\vec{R}_i \times \vec{S}_i$. Слияние матриц² по вертикали (строки одной под строками другой) проводим простой операцией $\langle A1, A2 \rangle$. Заметим, что если потребуется соединить матрицы по горизонтали, надо записать $\langle A1 | A2 \rangle$.

Система уравнений решается методом обратной матрицы $-A^{-1} \cdot B$, где точкой в **Maple** в данном случае обозначено произведение матрицы на вектор свободных членов. Операция

¹Этот файл хранится в архиве программ на сайте автора по адресу <http://vuz.exponenta.ru/progr.rar>.

²Различные действия с матрицами и векторами можно проводить с применением пакета `ArrayTools`. В частности, слияние матриц выполняет оператор `Concatenate` (конкатенация).

транспонирования применена к ответу только для более удобного вывода результатов — в строчку. Для сокращения записи изменено обозначение операции `Transpose` на `Tr` с помощью команды `macro`.

Для контроля ввода данных дадим изображение полки и всех сил, приложенных к ней. Параллелепипед задает процедура `suboid`. Два обязательных аргумента — координаты концов большой диагонали. В общем параллелепипед непрозрачный, из-за чего усилий внутри него можно не увидеть. Прозрачность придает опция `style=line`. Орты неизвестных изображаем командой `arrow` пакета `plottools` (в программе он был задан после `plots`, имеющего такую же команду `arrow`, но по другому заданную). Здесь требуется вектор-строка, а использованная ранее команда `Vector` дает столбец. Применяем операцию `Transpose`, точнее, ее сокращение `Tr`. Масштабные коэффициенты 2 и 1.5 при `F[k]` и `S[k]` использованы только для корректировки внешнего вида рисунка и в общем необязательны.

Программа 7 (P7.mws)

```
> restart: with(plots): with(plottools):#Полка
> with(LinearAlgebra):
  Размеры объекта (м)
> a:=2: b:=6: c:=3:
  Нагрузки (кН)
> F[1]:=Vector([0,3,0]):
> F[2]:=Vector([0,0,-12]):
  Координаты точек приложения нагрузок
> Точки1[1]:=Vector([a,b,0]):
> Точки1[2]:=Vector([a/2,b/2,0]):
> N:=add(F[i],i=1..2):
> M:=add(CrossProduct(Точки1[i],F[i]),i=1..2):
  Вектор правых частей системы
> V:=<N,M>:
  Направления усилий (№ 1, 2, 3 – реакция шарнира А,
  № 4 – вертикальный опорный стержень,
  № 5 – горизонтальный опорный стержень,
  № 6 – подпорка)
> S[1]:=Vector([1,0,0]):
> S[2]:=Vector([0,1,0]):
> S[3]:=Vector([0,0,1]):
> S[4]:=Vector([0,0,1]):
> S[5]:=Vector([-1,0,0]):
> S[6]:=Normalize(Vector([-a,-b,-c]),Euclidean):
```

```

> A1:=Matrix([seq(S[i],i=1..6)]):
  Координаты точек приложения
> Коорд[1]:=Vector([0,0,0]):
> Коорд[2]:=Vector([0,0,0]):
> Коорд[3]:=Vector([0,0,0]):
> Коорд[4]:=Vector([a,0,0]):
> Коорд[5]:=Vector([0,b,0]):
> Коорд[6]:=Vector([a,b,0]):
> A2:=Matrix([seq(CrossProduct(
>
  Коорд[i],S[i]),i=1..6)]):
> A:=<A1,A2>;

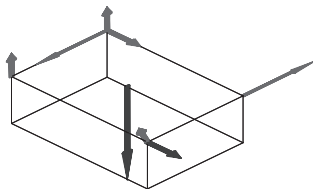
```

$$A := \begin{bmatrix} 1 & 0 & 0 & 0 & -1 & \frac{-2}{7} \\ 0 & 1 & 0 & 0 & 0 & \frac{-6}{7} \\ 0 & 0 & 1 & 1 & 0 & \frac{-3}{7} \\ 0 & 0 & 0 & 0 & 0 & \frac{-18}{7} \\ 0 & 0 & 0 & -2 & 0 & \frac{6}{7} \\ 0 & 0 & 0 & 0 & 6 & 0 \end{bmatrix}$$

```

> Вид:=0.1,0.2,0.3,cylindrical_arrow:
> macro(Tr = Transpose):
> Nagr:=seq(arrow(Tr(Точки1[k]),Tr(F[k]/2),
>
  Вид,color=blue),k=1..2):
> Sterz:=seq(arrow(Tr(Коорд[k]),Tr(S[k]*1.5),
>
  Вид,color=red),k=1..6):
> display(Nagr,Sterz,cuboid([0,0,0],[a,b,-c]),
>
  shading=none,style=line);
> Tr(-A^(-1).B);#Ответ

```



$$\begin{bmatrix} -5 & -15 & 6 & 0 & -1 & -14 \end{bmatrix}$$

7.6. Статические инварианты

Возможности Maple и пакета LinearAlgebra наиболее ярко проявляются в решении пространственных задач статики. Ввод данных о нагрузках в виде векторов Vector позволяет задействовать удобные операторы: CrossProduct — для векторного произведения и DotProduct — для скалярного. Помимо вычислений, данные о векторах можно легко использовать для визуализации условия. Это можно выполнить различными способами. В Maple есть пакет plots и пакет plottools для графического отображения вектора в виде стрелки arrow. Выбираем первый пакет, тем более что в нем есть также совершенно необходимый оператор display для вывода рисунка на экран¹. При определении модулей векторов можно использовать оператор нормы вектора в форме VectorNorm(R, Euclidean) или Norm(R, 2). В конце программы (сверх поставленной задачи по определению инвариантов) выводятся уравнения центральной винтовой оси. Записываются левые части уравнений:

$$5 - 3y - 4z = 0, \quad 3x - 12/5 = 0, \quad -16/5 + 4x = 0.$$

Из трех уравнений оси независимы два (прямая — пересечение двух плоскостей). Здесь два последних уравнения отличаются множителем.

Изображение сил и осей координат дает оператор display. Так как задача пространственная, то необходимо подобрать угол обзора. За это отвечает опция orientation=[ϑ , φ]. Здесь ϑ и φ , углы в градусах, указывают положение наблюдателя. При $\vartheta=90$ наблюдатель видит рисунок с конца оси y , при $\varphi=90$ наблюдатель находится где-то на плоскости xy , а при $\varphi=0$ видит все сверху — с конца оси z .

В программу введены данные задачи 18, с. 47.

В старых версиях Maple вместо пакета LinearAlgebra применялся пакет линейной алгебры linalg. Для сохранения преемственности этот пакет сохранен и последних версиях Maple.

Программа 8 (P8.mws)

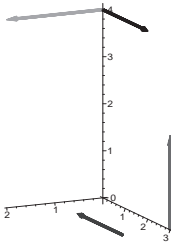
```
> restart:#Инварианты приведения системы сил
> with(LinearAlgebra):
> with(plots): N:=4:
  Проекция векторов сил
> F[1]:=Vector([0,0,1]):
```

¹Пакет plottools создает объекты для изображения (линии, фигуры) и используется вместе с пакетом plots, содержащим оператор display. При этом надо следить за порядком их подключения. При двойном толковании одноименных операторов принимается вариант последнего подключенного пакета.

```

> F[2]:=Vector([0,-3,0]):
> F[3]:=Vector([0,5,0]):
> F[4]:=Vector([2,0,0]):
  Координаты точек приложения
> A[1]:=Vector([0,3,0]):
> A[2]:=Vector([1,3,0]):
> A[3]:=Vector([0,0,4]):
> A[4]:=A[3]:
> # Два одинаковых вектора можно ввести и так:
> #           A[3],A[4]:=Vector([0,0,4])$2
  Главный вектор
> R:=add(F[i],i=1..N):
> R0:=Norm(R,Euclidean);
  Главный момент
> M:=add(CrossProduct(A[i],F[i]),i=1..N):
> M0:=Norm(M,Euclidean):
  Скалярный инвариант
> Inv:=DotProduct(R,M);
  Минимальный момент
> Mmin:=Inv/R0:
  Шаг винта
> p:=Mmin/R0:
> V:=Vector([x,y,z]):
  Уравнение центральной винтовой оси
> AX:=evalm(M-CrossProduct(V,R)-R*p);
           R0 := 3
           Inv := -21
  AX :=  $\left[ -\frac{37}{3} - y + 2z, \frac{38}{3} - 2z + x, -\frac{2}{3} - 2x + 2y \right]$ 
> Цвет:=[red,blue,black,green]:
> Стрелка:=width=0.05,head_length=0.2,
>
           color=Цвет[i],length=2:
> Вид:=scaling=constrained,
>
           axes=normal,orientation=[65,76],
>
           axesfont=[HELVETICA,6],tickmarks=[3,3,4]:
> display(seq(arrow(A[i],F[i],Стрелка),i=1..N),Вид);

```



Для большего эффекта рисунок можно оживить, заставив его, например, поворачиваться вокруг оси z на 90 градусов. В цикле по j создадим 90 кадров, отличающиеся параметром в угле поворота $65+j$. Кадры получим оператором `display` (не забыв поставить после него двоеточие вместо точки с запятой) и обозначим их $P[j]$, $j = 1, \dots, 90$:

```
> for j to 90 do
> Вид:=axes=normal,orientation=[65+j,76]:
> P[j]:=display(seq(arrow(A[i],F[i]),i=1..N),Вид):
> od:
> display(seq(P[i],i=1..90),insequence=true);
```

Для сокращения программы некоторые необязательные опции здесь опущены. Затем, с помощью того же оператора `display`, но с опцией `insequence=true`, последовательно стирающей одно изображение и выводящей другое, отображаем получившееся видео на экран. Далее, когда программа выдаст изображение, надо активизировать его мышкой и в интерфейсе **Maple** появится панель магнитофонного типа (рис. 412). Если нажать клавишу "Пуск", рисунок начнет вращаться.




Рис. 412

7.7. Набор стандартных процедур для рисунков

Многие программы, используемые для решения задач механики или для иллюстрации этих решений, содержат часто повторяющиеся процедуры, программировать которые всякий раз заново нет смысла.

В **Maple** предусмотрена возможность записи таких программ в специальный файл с расширением `*.m`. Для использования этих программ достаточно вызвать соответствующий файл. Это существенно упрощает программирование и увеличивает его надежность. В программе 9 собраны часто используемые процедуры для вычислений и рисования.

Процедура **Опора** изображает неподвижную шарнирную опору  в точке с координатами $x[i]$, $y[i]$. Радиус окружности в шарнире равен R . Переменные массивы x , y являются глобальными, поэтому в обращении к процедуре достаточно указать номер точки, в который шарнир помещается, и радиус R в зависимости от масштаба рисунка. Пользовательские процедуры названы по-русски для того, чтобы отличить их от процедур, операторов и команд **Maple**.

Процедура **Окружн** изображает окружность в точке i радиуса R , процедура **Линия** рисует отрезок от точки i к точке j . Толщину и цвет линий можно определить опциями `thickness` и `color`.

В программе содержатся три вычислительные процедуры. В процедуре **Координата** определяется точка k пересечения двух окружностей радиусов ik и jk с центрами в точках i и j (рис. 413). Из двух

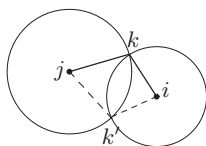


Рис. 413

точек пересечений выбирается та, которая лежит справа от направленного отрезка $i \rightarrow j$. Аналогично в процедуре **Координата2** вычисляются координаты внутренней точки k на отрезке ij по длинам ik и jk . Очевидно, **Координата2** есть частный случай процедуры **Координата**, когда окружности пересекаются в одной точке, однако этот частный

случай очень неустойчив по отношению к размерам. Если сумма радиусов при таком подходе окажется немного меньше расстояния между точками, программа выдаст ошибку. Именно поэтому и создана процедура **Координата2**. Процедура **Pif** является простым аналогом **Norm(*, 2)** и не требует загрузки пакета **LinearAlgebra**. Форма программирования процедуры `:= (x, y) -> вместо := proc (x, y)` использована только для иллюстрации возможностей **Maple**. Запись полученных процедур и часто используемой числовой константы π (не путать с символьной константой Pi) производится оператором `save` в файл "Podpr1.m". Имя файла произвольное, но русские буквы тут не допускаются. Можно указать путь к файлу, например "C:\Podpr1.m". Расширение `m` файла также менять нельзя. Абсолютные адреса предпочтительней, так как не требуют привязки программы к месту ее запуска.

Начиная с **Maple 11** в файл `*.m` с процедурами надо подгружать необходимые пакеты, в данном случае `plottools` и `plots`. В старых версиях это можно было делать только в вызывающей программе.

Программа 9 (P9.mws)

```

> restart:with(plottools):with(plots):pi:=evalf(Pi):
  Процедура изображения неподвижной опоры в точке i
> Опора:=proc(i,R) local x0,x1,y0,h,N:
>   x0:=x[i]-R*0.7:x1:=x[i]+R*0.7:
>   y0:=y[i]-R*0.7:h:=3*R:N:=2:
>   display(PLOT(circle([x[i],y[i]],R,color=blue)),
>   PLOT(CURVES([[x0,y0],[x[i]-h,y[i]-h]],
>               [[x1,y0],[x[i]+h,y[i]-h]],
>               [[x0-h,y[i]-h],[x1+h,y[i]-h]],
>               seq([[x0-h*(-j/N+1),y[i]-h],
>                   [x0-h*(-j/N+1)-h/4,y[i]-h*1.3]],j=0..2*N+1))))):
> end:
  Процедура изображения окружности с центром в точке i
> Окружн:=proc(i,R) global CLR:
>   PLOT(circle([x[i],y[i]],R,color=CLR))
> end:
  Цвет окружности по умолчанию
> CLR:=blue:
  Процедура изображения линии от точки i к j
> Линия:=proc(i,j) global x,y:
>   PLOT(CURVES([[x[i],y[i]],x[j],y[j]]]))):
> end:
  Процедура определения координат точки K
  по координатам i и j и длинам ik и jk
> Координата:=proc(i,j,k,ik,jk)
>   local L0,sina,cosa,cosb,sinb: global x,y:
>   L0:=Pif(x[i]-x[j],y[i]-y[j]):
>   sina:=(y[i]-y[j])/L0:cosa:=(x[i]-x[j])/L0:
>   cosb:=(jk^2+L0^2-ik^2)/(2*jk*L0):
>   sinb:=sqrt(1-cosb^2):
>   x[k]:=x[j]+jk*(cosa*cosb-sina*sinb):
>   y[k]:=y[j]+jk*(sina*cosb+cosa*sinb):
> end:
  Процедура определения координат точки k на отрезке ij
  по координатам концов i и j и длинам ik и jk

```



```

> Координата2:=proc(i,j,k,ik,jk)
>   global x,y:
>   x[k]:=x[i]+ik/(ik+jk)*(x[j]-x[i]):
>   y[k]:=y[i]+ik/(ik+jk)*(y[j]-y[i]):
> end:
  Теорема Пифагора
> Pif:=(x,y)-> sqrt(x^2+y^2):
  Запись процедур на диск
> save CLR,Pif,Координата,Координата2,pi,
>   Линия,Окружн,Опора,"C:\\Podpr1.m";

```

7.8. Кинематика точки

Решим задачу 19, с. 51. Для нахождения скорости и ускорения точки применяем оператор дифференцирования `diff`. Траекторию строим по ее параметрическому представлению оператором `plot`. Для наглядности средствами **Maple** на экране компьютера получаем изображение движущейся точки вместе с изменением векторов скорости и ускорения. Для этого формируем массив изображений — отдельных кадров движения¹.

Программа 10 состоит из двух основных частей.

В первой части программы строим траекторию движения точки и изображаем векторы скорости и ускорения. Количество кадров задаем числом N . В цикле от 1 до N в массиве `p[i]` создаем N кадров, показывающих положение вектора скорости \vec{v} и вектора ускорения \vec{a} на траектории; \vec{r} — радиус-вектор точки. Вторая производная в операторе дифференцирования `diff` обозначена значком `$2`. Оператор `map` применяет операцию дифференцирования к каждому элементу вектора и создает соответствующий вектор скорости или ускорения. Коэффициенты `mv` и `ma` введены для масштабирования изображений векторов и подбираются вручную.

Перед построением траектории производим очистку переменной времени `t:=t'`.

Для того чтобы при выводе изображения оператором `display` изображения не накладывались друг на друга, а замещались, создавая тем самым эффект движения, используем опцию `insequence=true`. График параметрически заданной функции (траектории) строим оператором `plot`. Время меняем от 0 до T . Совмещенный вывод изображения неподвижной траектории и движущихся стрелок осуществляем оператором `display`.

¹ Аналогичная программа приведена в книге [42].

Во второй части программы вычисляем тангенциальное ускорение, нормальное ускорение и радиус кривизны траектории.

Используем пакет линейной алгебры `linalg`. Сначала находим модуль вектора скорости `norm(v, 2)`. Цифра 2 означает, что норма вычисляется по формуле $v = \sqrt{v_x^2 + v_y^2}$.

Тангенциальное ускорение вычисляем с помощью скалярного произведения `dotprod`: $|a_\tau| = \vec{a} \cdot \vec{v}/v$. Выведем аналогичную формулу для вычисления нормального ускорения. Умножаем векторно равенство $\vec{a} = \vec{a}_\tau + \vec{a}_n$ на \vec{v} :

$$\vec{v} \times \vec{a} = \vec{v} \times \vec{a}_\tau + \vec{v} \times \vec{a}_n.$$

Учитывая, что $\vec{v} \times \vec{a}_\tau = 0$ и $|\vec{v} \times \vec{a}_n| = va_n$, получаем $|\vec{v} \times \vec{a}| = va_n$. Использован оператор векторного произведения `crossprod`.

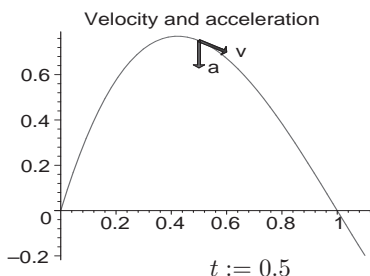
Программа 10 (P10.mws)

```
> restart: # Кинематика точки
      Количество кадров анимации и масштабные коэффициенты
> N:=50: ka:=0.05: kv:=0.1:
      Закон движения
> x:=t: y:=2*t*(t-1)*(t-2):
> T:=1.1: # Время движения
> r:=vector([x,y,0]):# Векторы скорости и ускорения
> v:=map(diff,r,t): a:=map(diff,r,t$2):
> with(plots):
> pv:=0.01,0.04,0.11: # Параметры стрелок на рисунке
      v := [1, 2(t-1)(t-2)+2t(t-2)+2t(t-1), 0]
      a := [0, 12t-12, 0]
> with(plottools):
> for i to N do
>   t:=i/N*T:
>   r1:=vector([r[1],r[2]]):
>   v1:=vector([v[1],v[2]]):
>   a1:=vector([a[1],a[2]]):
>   acc:=arrow(r1,evalm(a1*ka),pv,color=red):
>   vel:=arrow(r1,evalm(v1*kv),pv,color=blue):
>   txa:=TEXT([r[1]+a[1]*ka,r[2]+a[2]*ka],"a"):
>   txv:=TEXT([r[1]+v[1]*kv,r[2]+v[2]*kv],"v"):
>   p[i]:=display(vel,acc,txa,txv):
> end:
```

```

> t:='t':
> g1:=display(seq(p[i],i=1..N),insequence=true):
> g2:=plot([x,y,t=0..T]):
> display(g1,g2,scaling=constrained,
> title="Скорость и ускорение точки");
> with(linalg): V:=norm(v,2): #Модуль скорости
> at:=dotprod(v,a)/V: #Касательное ускорение
> a_:=norm(a,2): #Модуль ускорения
> an:=norm(crossprod(v,a),2)/V:#Нормальное ускорение
> t:=0.5; #Заданное время

```



```

> `x`=x, `y`=y, `v`=V, `at`=at,
> `an`=an, `a`=a_, `R`=V^2/an;

```

```

x = 0.5, y = 0.750, v = 1.118033989, at = 2.683281572,
an = 5.366563145, a = 6.0, R = 0.2329237479

```

7.9. Движение точки в полярных координатах

В программе 11 вычисляются скорость и ускорение точки, движение которой задано в полярных координатах. Решена задача 20 на с. 53. Компоненты скорости объединены в двумерную структуру `vector`, к которой можно применить оператор вычисления длины `norm(v,2)` из пакета `linalg`¹.

Программа 11 (P11.mws)

```

> restart:
> ro:=t*sin(3*t): f:=t^3: # Закон движения
> vr:=diff(ro,t); ft:=diff(f,t); vf:=ro*ft:

```

¹Более современная версия этого пакета `LinearAlgebra`. В этом пакете используются обозначения `Vector` и `Norm(v,2)`.

```

> v:=vector([vr,vf]):
> with(linalg):
> V:=norm(v,2): # Модуль скорости
> ar:=diff(ro,t$2)-ro*ft^2:
> af:=ro*diff(f,t$2)+2*vr*ft:
> a:=sqrt(ar^2+af^2):#Модуль ускорения
> at:=(vr*ar+vf*af)/V:
> an:=sqrt(a^2-at^2):#Нормальное ускорение
> t:=1.:
> Vel=V;Acc=a;At=at;An=an;
      vr := sin(3t) + 3t cos(3t)
      ft := 3t^2
      Vel = 2.860361579
      Acc = 18.22015177
      At = 5.999854577
      An = 17.20394361

```

7.10. Многозвенный механизм

7.10.1. Расчет угловых скоростей

Для решения задачи 21, с. 60, используем метод кинематических графов, с. 55. Получение двух уравнений для скоростей в проекциях на оси оформим в виде процедуры. Граф может иметь разный размер (число дуг, [29]). Поэтому аргументы процедуры зададим четырьмя списками. Первый список — вершины графа, или узловые точки, между которыми в графе указываются расстояния (второй аргумент L), направления переходов (углы в радианах, отсчитываемые против часовой стрелки от положительного направления оси x) — третий аргумент f . Используется правая система координат. Последний аргумент — угловые скорости. Результат работы процедуры — два уравнения вида (2.8), с. 55, получается в виде списка. Список сформирован оператором `seq`. Суммирование по дугам графа (их на одну меньше числа вершин) выполняется оператором `add`. Верхний предел суммирования подсчитан командой `ops(f)` по числу элементов в третьем аргументе f . Заметим, что в первом аргументе S процедуры только начало и конец списка используются для процедуры. Остальные даны как комментарий, для придания внешней схожести процедуры и принятого обозначения графа. Последний элемент списка в **Maple** имеет условный номер (-1) . Если в первом уравнении слагаемые содержат $\sin(\varphi_i)$, то во втором должны присутствовать $\cos(\varphi_i)$. Оба эти случая обобщаются

выражением $\sin(\varphi_i - (j-1)\pi/2)$. При $j = 1$ имеем первое уравнение, при $j = 2$ — второе.

Проекция скорости на ось x обозначена вторым индексом 1, на ось y — индексом 2. Таким образом, $v[A, 1] = v_{Ax}$, $v[A, 2] = v_{Ay}$.

Процедуру **Граф** потребуется вызывать еще неоднократно — многие задачи кинематики и динамики мы будем решать аналитическим методом. Поэтому оператором **save** запишем ее на диск в файл **graph.m**.

Программа 12 (P12.mws)

```
> restart;# Кинематический граф
  (список вершин, длины, углы, угловые скорости)
> Граф:=proc(S,L,f,w)
> [seq(v[S[-1],j]=v[S[1],j]-
> add(L[i]*sin(f[i]-Pi/2*(j-1))*w[i],
>                                     i=1..nops(f)),j=1..2)]
> end:
> save Граф, "C:\\graph.m"; # Сохраняем на диске
```

Следующая программа использует процедуру **Граф** для расчета угловых скоростей в задаче 21, с. 60. Пронумеруем стержни механизма (рис. 97): OA — № 1, AB — № 2, DE — № 3, FE — № 4, CB — № 5.

Система уравнений сформирована из отдельных уравнений графов, но ее можно записать также в виде $\{op(Gr1), op(Gr2)\}$.

Программа 13 (P13.mws)

```
> restart;# Решение задачи 21
> read "C:\\graph.m";
  Неподвижные точки (нулевые скорости)
> v[O,1],v[O,2],v[F,1],v[F,2],v[C,1],v[C,2]:=0$6:
  Заданная угловая скорость OA
> w1:=12:
  Размеры
> BC:=5: AD:=3: DE:=6: EF:=6: OA:=2: AB:=9:
>                                     alpha:=arccos(4/5):
> Gr1:=Граф([C,B,A,O],[BC,AB,OA],[Pi+alpha,Pi/2,Pi],):
>                                     [w5,w2,w1]):
> eq1:=Gr1[1]; eq2:=Gr1[2];
  eq1 := 0 = 3 w5 - 9 w2
```

```

eq2 := 0 = -24 - 4 w5
> Gr2 := Граф ([ F, E, D, A, O ], [ EF, DE, AD, OA ],
>
[ Pi/2, 0, Pi/2, Pi ], [ w4, w3, w2, w1 ] ):
> eq3 := Gr2[1]; eq4 := Gr2[2];
> solve ( { eq1, eq2, eq3, eq4 }, { w2, w3, w4, w5 } );
eq3 := 0 = -6 w4 - 3 w2
eq4 := 0 = -24 + 6 w3
{ w3 = 4, w5 = -6, w2 = -2, w4 = 1 }

```

7.10.2. Анимация

Следующая программа непосредственно не решает задачи кинематики механизма (задача 21), а только иллюстрирует его движение. Для решения задачи надо четко представлять объект исследования:

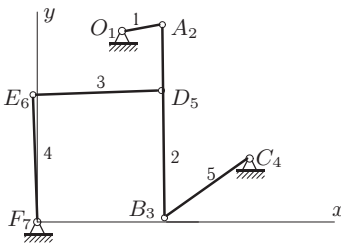


Рис. 414

как он движется, какие у него особенности. Система **Maple** хорошо приспособлена для этого. Проиллюстрируем движение механизма на рис. 97, с. 60. Выберем систему координат, пронумеруем шарниры, обозначив номера нижними индексами, и вычислим координаты опорных шарниров (рис. 414). Номера в программе присвоим списком, используя

оператор повторения $\$$. Запись $i\$\{i=1..7$ означает список 1, 2, 3, 4, 5, 6, 7. Вместо букв O и D, которые являются служебными в **Maple** (остаток ряда и производная-оператор), пришлось использовать обозначения O1 и D1. При необходимости можно, конечно, снять "служебную" защиту с любого имени, например, `unprotect(O)`. Также списком присвоим координаты шарниров. Безусловно, в программе можно было бы использовать непосредственную нумерацию точек, т. е. $x[2]$ вместо $x[A]$ и др., но для наглядности программы лучше обозначать точки буквами, как в условии. В цикле по i создаются кадры изображения $P[i]$. Равномерное вращение ведущего звена OA обеспечивается приращением угла поворота на постоянную величину $df:=2*\pi/N$ в каждом цикле. За счет этого меняется координата точки A, затем координаты точки B, определяемые с помощью процедуры Координата (с. 356), и, наконец, координаты точек D и E.

Число кадров N надо выбирать так, чтобы движение было достаточно плавным, но не слишком большим, потому что формирование геометрических образов в **Maple** происходит не очень быстро, а размер



GIF-анимации может получиться значительным: 1,5Мб при $N=96$ в данной задаче¹. Оценить время счета можно с помощью оператора `time()`. Для этого в начале программы (сразу после `restart`) надо засечь время `t0:=time()`, а в конце программы вычислить время в секундах, как разность конечного и начального: `time()-t0`.

Оператор `display` в программе работает дважды. Первый раз он записывает содержимое кадра в элементы массива `P[i]`, поэтому после оператора и после конца цикла надо непременно поставить двоеточие, чтобы не забыть экран множеством ненужных чисел и выражений, соответствующих графическим образам. Второй раз оператор `display` выводит на экран последовательность созданных образов — один вместо другого. Стирать один кадр и выдавать следующий призывает опция `insequence=true`.

Программа 14 (P14.mws)

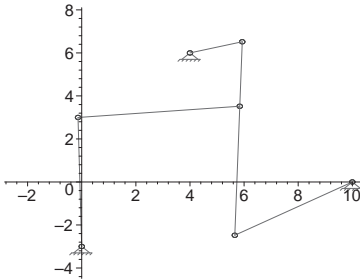
```
> restart:
> read "C:\\Podpr1.m":# Программа 9, с. 356
> with(plots):
> O1,A,B,C,D1,E,F:=i$1=1..7:# Номера точек
> x[O1],x[C],x[F]:=6,0,10: # Координаты опор
> y[O1],y[C],y[F]:=0,6,0:
> OA,AB,CB,AD,BD,DE,FE:=2,9,5,3,6,6,6:#Длины стержней
> N:=24: # Число кадров
> df:=2*pi/N:# Шаг угла
> for k to N do
> f:=df*k:
> x[2]:=x[1]+OA*cos(f):
> y[2]:=y[1]+OA*sin(f):
> Координата(A,C,B,AB,CB): #x[B],y[B]-? См. прогр.9
> Координата2(A,B,D1,AD,BD):#x[D1],y[D1]-? Прогр.9
> Координата(D1,F,E,DE,FE): #x[E],y[E]-?
> M:=2,3,5,6:
> K:=4,1,7:
> Шарниры:=seq(Окружн(M[i],0.1),i=1..4):#См. прогр.9
```

¹Любителям анимации рекомендуем освоить несложный, похожий на Си, язык Action Script. Результат в формате swf получается значительно лучшего, чем в Maple, качества и существенно (в десятки и сотни раз) меньшего размера. Примеры, выполненные студентами МЭИ(ТУ) во время прохождения производственной практики, можно посмотреть на сайте <http://vuz.exponenta.ru/> в разделе «Механические модели».

```

> Опоры:=seq(Опора(К[i],0.1),i=1..3): #См. прогр. 9
> Стержни:=Линия(О1,А),Линия(А,В),Линия(В,С),
> Линия(Д1,Е),Линия(Е,Ф),thickness=2: # См. прогр. 9
> Р[k]:=display(Стержни,Шарниры,Опоры):
> od:
> display(seq(Р[i],i=1..N),insequence=true);

```



Для вывода движущегося объекта оператор `display` работает в программе с опцией `insequence=true`. По умолчанию `insequence=false`. Интересно посмотреть, что получится в этом случае. Запустим измененную таким образом программу. То, что получается, видно на рисунке 415.

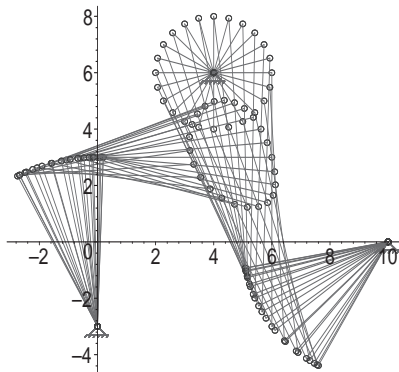


Рис. 415

Незакрашенные части плоскости естественным образом указывают на зоны безопасности — в этих местах можно свободно располагать какие-либо объекты без риска соударения с частями движущегося механизма.

Следующие две программы дают решения задач 22 и 23, с. 62, 65.

Программа 15 (P15.mws)

```
> restart; # Решение задачи 22
> read "C:\\graph.m";
  Неподвижные точки (нулевые скорости)
> v[0,1],v[0,2],v[P,1],v[P,2]:=0$4:
  Угловая скорость (в рад/с) и радиус цилиндра (в см)
> w1:=2: R:=4:
> Gr1:=Граф([P,D1,A,O],[R*sqrt(2),8,4],
>           [Pi/4,0,-Pi/2],[w1,w5,w4]):
> eq1:=Gr1[1]; eq2:=Gr1[2];
> Gr2:=Граф([P,E,C,B,O],[R,4*sqrt(2),8,8],
>           [Pi/2,Pi/4,0,-Pi/2],[w1,w3,w2,w4]):
> eq3:=Gr2[1]; eq4:=Gr2[2];
> solve({eq1,eq2,eq3,eq4},{w2,w3,w4,w5}); # Решение
      eq1 := 0 = -8 + 4 w4
      eq2 := 0 = 8 + 8 w5
      eq3 := 0 = -8 - 4 w3 - 4 w4
      eq4 := 0 = 4 w3 + 8 w2
      {w4 = 2, w5 = -1, w3 = 2, w2 = -1}
```

Программа 16 (P16.mws)

```
> restart; # Решение задачи 23
> read "C:\\graph.m";
  Неподвижные точки, шарнир и МЦС (нулевые скорости)
> v[0,1],v[0,2],v[P,1],v[P,2]:=0$4:
  Угловая скорость
> w1:=20:
  Размеры
> R:=3: OA:=6: AB:=4*sqrt(2): DC:=4: AC:=10:
> Gr1:=Граф([O,A,B,P],[OA,AB,R*sqrt(2)],
```

```

> [-Pi/2, 3*Pi/4, 5*Pi/4], [w1, w2, w5] ):
> eq1:=Gr1[1]; eq2:=Gr1[2];
      eq1 := 0 = 120 - 4 w2 + 3 w5
      eq2 := 0 = -4 w2 - 3 w5
> Gr2:=Граф([P, D1, C, A, O], [R*sqrt(2), DC, AC, OA],
> [3*Pi/4, -Pi/2, 0, Pi/2], [w5, w3, w4, w1] ):
> eq3:=Gr2[1]; eq4:=Gr2[2];
> solve({eq1, eq2, eq3, eq4}, {w2, w3, w4, w5});
      eq3 := 0 = -120 - 3 w5 + 4 w3
      eq4 := 0 = -3 w5 + 10 w4
      {w2 = 15, w3 = 15, w4 = -6, w5 = -20}

```

7.11. Уравнения трех угловых ускорений

Программа 17 состоит из двух частей. Первая часть расчетная. В программе использованы данные задачи 26 на с. 71. В массивы x , y заносятся координаты шарниров, в массивы ω и ϵ — данные об угловой скорости и угловом ускорении ведущего звена. Нумерация узлов и стержней соответствует рисунку 115 на с. 73.

С помощью команды `sum` (можно также использовать команду `add`) организуются расчетные уравнения трех угловых скоростей (2.22). Решение, полученное оператором `solve`, заносится во временную переменную R только для того, чтобы потом оператором `assign` присвоить найденные значения угловых скоростей переменным $\omega[2]$ и $\omega[3]$. Они потребуются при определении ускорений. Структура уравнений трех угловых ускорений такая же, как и для скоростей.

Во второй части программы для контроля правильности ввода данных создается изображение механизма в заданный момент времени. Используются процедуры рисования **Линия** и **Опора** из программы 9. Файл `Podpr1.m`, содержащий эти процедуры, считывается оператором `read` в начале программы. В процедурах **Линия** и **Опора** массивы координат x и y являются глобальными. Заголовок рисунка "Four-link chain" (четырёхзвенник) записан в параметре `title` на английском языке. Заголовок можно писать и на русском, однако при конвертации документа в формат LaTeX или HTML здесь могут возникнуть проблемы со шрифтами.

Программа 17 (P17.mws)

```

> restart; read "C:\\Podpr1.m"; # Программа 9, с. 356
> with(plots):
      Угловая скорость, угловое ускорение

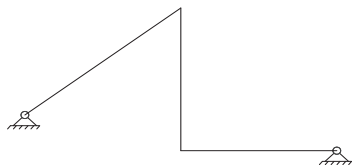
```

```

> omega[1]:=4: epsilon[1]:=0:
  Координаты
> x[1]:=0: y[1]:=0:
> x[2]:=4: y[2]:=3:
> x[3]:=4: y[3]:=-1:
> x[4]:=8: y[4]:=-1:
  Уравнения трех угловых скоростей
> eq1:=sum(omega[i]*(x[i]-x[i+1]),i=1..3):
> eq2:=sum(omega[i]*(y[i]-y[i+1]),i=1..3):
> R:=solve({eq1,eq2},{omega[2],omega[3]}):assign(R):
> omega2=omega[2],omega3=omega[3];
       $\omega_2 = 3, \omega_3 = -4$ 
  Уравнения трех угловых ускорений
> eq3:=sum(epsilon[i]*(x[i]-x[i+1]),i=1..3)
>      -sum(omega[i]^2*(y[i]-y[i+1]),i=1..3):
> eq4:=sum(epsilon[i]*(y[i]-y[i+1]),i=1..3)
>      +sum(omega[i]^2*(x[i]-x[i+1]),i=1..3):
> solve({eq3,eq4},{epsilon[2],epsilon[3]});
       $\{\epsilon_2 = 32, \epsilon_3 = 3\}$ 
> display(Линия(1,2),Линия(2,3),Линия(3,4),
>          Опора(1,0.1),Опора(4,0.1),axes=NONE,
>          scaling=constrained, title='Four-link chain');

```

Four-link chain



7.12. Механизм с муфтой

Вычисление скоростей и ускорений точек механизмов, содержащих муфты, представляет собой трудную задачу. Решение, основанное на теоремах о сложении скоростей и ускорений точек при плоском движении, дано на с. 74. Приведем здесь две программы расчета кинематики механизмов методом прямого дифференцирования координат точек.

Здесь наиболее полно проявляются преимущества системы компьютерной математики, позволяющей выполнять аналитические преобразования и дифференцирование. Рассмотрим сначала задачу 27, с. 74.

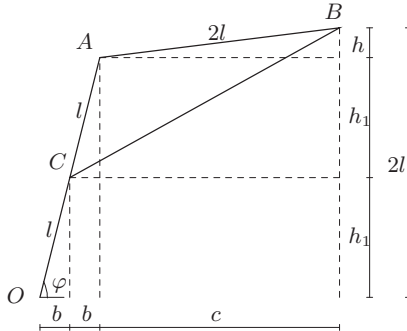


Рис. 416

Произвольное положение механизма определяется углом $\varphi(t)$ наклона стержня OA (рис. 416). При $\varphi(t) = \pi/2$ получаем расчетное положение механизма. Найдем зависимость расстояния $S = CB$ от времени, выразив это расстояние через угол $\varphi(t)$. По условию $OC = CA = l$, $AB = 2l$:

$$S = CB = \sqrt{(b+c)^2 + (2l-h_1)^2},$$

где $h_1 = l \sin \varphi$, $b = l \cos \varphi$, $c = \sqrt{4l^2 - h^2}$, $h = 2l - 2h_1$.

Дифференцируем по времени зависимость $S(t)$ в системе **Maple** и при $\varphi = \pi/2$, $\dot{\varphi} = \omega$ получаем относительную скорость $v_{от} = (2/\sqrt{5})\omega l$. Дифференцируя еще раз, получаем ускорение $a_{от} = 6\sqrt{5}\omega^2 l/25$. Оба метода дают одни и те же ответы.

Для того чтобы не путать длину l с единицей (они на экране практически неразличимы), сменим обозначение длины на L . В программе требуется указать тип этой переменной: `assume(L>0)`. Иначе **Maple** не будет упрощать выражения, и в промежуточных и конечных результатах будут появляться выражения типа $\sqrt{L^2}$, и никакие операторы упрощения `simplify` тут не помогут. В лучшем случае вместо $\sqrt{L^2}$ будет записано `Lcsgn(L)`, где `csgn` — функция знака. Переменные, указанные в операторе `assume`, выводятся в листинг по умолчанию со специальным значком \sim , что загромождает вывод и затрудняет чтение. Самый удобный (но не единственный) способ борьбы с этой проблемой — использование оператора `interface(showassumed=0)`.

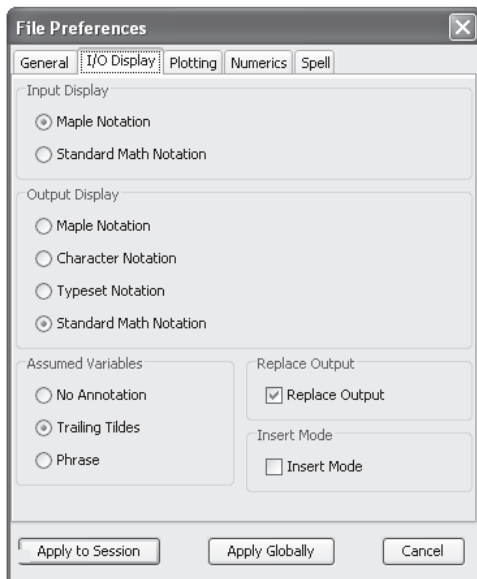


Рис. 417

Другой способ — зайти в меню: File -> Preference->I/O Display (рис. 417) и для Assumed Variables выбрать режим No Annotation. По умолчанию там установлен режим Trailing Tildes установки тильды над переменной с оговоренными свойствами. Можно также поставить режим Phrase текстового сопровождения (*with assumption on...*) переменной из оператора `assume`.

Выражения для скорости и ускорения при произвольном угле весьма громоздкие и не приводятся. Отметим особый эффект при вычислении ускорения. Вычисление ускорения обычным методом предполагает достаточно прочные знания и опыт определения величин и направлений ускорений, входящих в теорему Кориолиса. Такое решение громоздкое, а результаты недостоверны. Предлагаемый подход — надежный альтернативный способ проверки. Достаточно всего только добавить оператор повторения `$2` при вычислении производной.

Программа 18 (P18.mws)

```
> restart;      # Задача 27. Механизм с муфтой
> assume(L>0):# Предполагаем, что L>0
> interface(showassumed=0):
> f:=phi(t):   # Угол
```

```
> h:=2*L*(1-sin(f)): # Размеры
> b:=L*cos(f):
> c:=sqrt(4*L^2-h^2):
> S:=sqrt((b+c)^2+(2*L-L*sin(f))^2); #S=S(t)=BC
```

$$S := \sqrt{(L \cos(\phi(t)) + 2 \sqrt{L^2 - L^2 (1 - \sin(\phi(t)))^2} + (2L - L \sin(\phi(t)))^2}$$

Скорость и ускорение

```
> v:=subs(diff(phi(t),t)=w,diff(S,t)):
> a:=subs(diff(phi(t),t)=w,diff(S,t$2)):
> V=simplify(subs(f=Pi/2,v)):
> A=simplify(subs(f=Pi/2,a)):
```

$$V = -\frac{2\sqrt{5}Lw}{5}$$

$$A = \frac{6\sqrt{5}Lw^2}{25}$$

В следующей программе аналогичным методом решается задача 28 на с. 78. Сначала в программе определяются координаты шарнира B . Так как угол меняется в зависимости от времени, координаты также зависят от времени. Расстояние S от муфты D до шарнира O вычисляем из решения системы (2.32), с. 83. Дифференцируя полученную зависимость, находим скорость муфты и ускорение относительно стержня. Особенность решения — в громоздкости формул (они не приводятся, так как занимают несколько страниц) и в необходимости выбора нужного решения из четырех возможных. Число решений складывается из двух решений задачи о пересечении окружности при определении координат точки B и двух решений системы (2.32). Номер 4 в определении $S[4]$ выбран экспериментально и контролируется значением $S[4]$ при $f=Pi/4$. Муфта должна находиться примерно на середине кривошипа. Решение получается несмотря на то, что некоторые промежуточные результаты имеют неявный вид типа *RootOf*.

Ответ, полученный в программе, совпадает с графическим и аналитическим решениями.

Программа 19 (P19.mws)

```
> restart; # Задача 28. Механизм с муфтой
> CO:=62: BD:=99.32: #Размеры
> BC:=43: AB:=111: OA:=40:
> omega:=3: # Угловая скорость
> f:=F(t): # Угол
```

```

> xa:=OA*cos(f)+CO: #Координаты шарнира A
> ya:=OA*sin(f):
> eq1:=(xb-xa)^2+(yb-ya)^2=AB^2:
> eq2:=xb^2+yb^2=BC^2:
> R:=evalf(solve({eq1,eq2},{xb,yb})): assign(R);
> x:=S*cos(f)+CO: #Координаты муфты
> y:=S*sin(f):
> eq3:=(xb-x)^2+(yb-y)^2=BD^2:
> S:=solve(eq3,S):
> Vot:=diff(S[4],t): #Скорость
> Aot:=diff(S[4],t$2):#Ускорение
> evalf(subs(f=Pi/4,S[4]));# S=OD
19.99027571
> Z1:=subs(diff(f,t)=omega,Vot):
> Z2:=subs(diff(f,t)=omega,Aot):
> evalf(subs(f=Pi/4,Z1));# Подставляем заданный угол
-58.40091671
> evalf(subs(f=Pi/4,Z2));
-464.2036469

```

7.13. Кинетическая энергия

По заданным массам, радиусам и найденным кинематическим соотношениям в программе определяются приведенные массы механической системы — коэффициенты при $v^2/2$ в выражении для кинетической энергии. В качестве примера использованы данные задачи 29 на с. 87. Для упрощения угловые скорости обозначены буквой w . В языке Maple различается регистр букв, что удобно для ввода различных радиусов одного блока. Меньший радиус rc не совпадает с большим — Rc . Массы даны в килограммах, радиусы и радиусы инерции — в сантиметрах. В программе вычисляются модули скоростей и угловых ускорений с помощью метода МЦС, описанного на с. 87. В данном случае этот метод удобнее, чем метод графов.

Программа 20 (P20.mws)

```

> restart;# Приведенная масса, задача 29
Дано:
> ib:=3: rb:=4: Rb:=5:
> ic:=6: rc:=2: Rc:=8:
> ma:=7: mb:=25: mc:=8: md:=16: me:=50:

```

Угловые скорости, $v=va$

$$> \text{wb}:=v/Rb: \quad \text{wc}:=\text{wb}*\text{rb}/2/Rc:$$

Скорости

$$> \text{vc}:=\text{wc}*Rc: \quad \text{vd}:=\text{wc}*(Rc+\text{rc}): \quad \text{ve}:=\text{vc}:$$

Моменты инерции

$$> \text{Jb}:=\text{mb}*\text{ib}^2: \quad \text{Jc}:=\text{mc}*\text{ic}^2:$$

Кинетические энергии

$$> \text{Ta}:=\text{ma}*v^2/2;$$

$$> \text{Tb}:=\text{Jb}*\text{wb}^2/2;$$

$$> \text{Tc}:=\text{Jc}*\text{wc}^2/2+\text{mc}*\text{vc}^2/2;$$

$$> \text{Td}:=3/4*\text{md}*\text{vd}^2;$$

$$> \text{Te}:=\text{me}*\text{ve}^2/2;$$

$$ve := \frac{4v}{3}$$

$$vd := \frac{v}{2}$$

$$Ta := \frac{5v^2}{2}$$

$$Tb := \frac{9v^2}{2}$$

$$Tc := \frac{13v^2}{2}$$

$$Td := \frac{3v^2}{2}$$

$$Te := 16v^2$$

Приведенная масса системы

$$> \text{mu} := (\text{Ta}+\text{Tb}+\text{Tc}+\text{Td}+\text{Te}) / (v^2/2);$$

$$\mu := 32$$

7.14. Уравнение Лагранжа 2-го рода.

Одна степень свободы

План составления уравнения движения механической системы с одной степенью свободы дан на с. 91. Для вывода кинематических зависимостей используем метод кинематических графов, с. 55, и процедуру **Граф**, с. 361, реализующую этот метод. Рассмотрим для примера решение задачи 37 на с. 101. Запрограммируем два графа (рис. 150)

$$K \xrightarrow{\frac{2}{\pi/2}} A \xrightarrow{\frac{2}{\varphi}} P \xrightarrow{\frac{3}{\varphi}} B \xrightarrow{\frac{3}{0}} C,$$

$$A \xrightarrow[\varphi]{0} B \xrightarrow[0]{3} C.$$

Здесь над стрелкой графа указан номер тела. Спарник AB обозначен номером 0. Этим графам соответствуют четыре уравнения для неизвестных скоростей v_{Kx} , v_{Ax} и угловых скоростей ω_{2z} и ω_{3z} . Нулевые скорости v_{Cx} , v_{Cy} , v_{Ky} , v_{Ay} заданы с помощью команды повторения $\$$. Эта же команда использована и в обращении к процедуре для сокращения записи: $[r\$,4]$ вместо $[r, r, r, r]$. Решение системы четырех кинематических уравнений (оператор `solve`) дает выражения всех необходимых величин через обобщенную координату φ , обозначенную как $w0$. Далее записываем кинетическую энергию системы. Обозначаем частные производные, входящие в уравнение Лагранжа: $z1 = \frac{\partial T}{\partial \dot{\varphi}}$, $z2 =$

$= \frac{\partial T}{\partial \dot{\varphi}}$. Для вычисления полной производной по времени необходимо сделать замену переменной Φ на функцию времени $\Phi := \text{phi}(t)$. После этого простым дифференцированием по времени по найденным производным $z1$, $z2$ и обобщенной силе Q получается уравнение Лагранжа¹. Заметим, что деление $M*w2/w0$ не приводит к автоматическому сокращению на $\dot{\varphi}$. Требуется применить один из операторов упрощения `simplify`, `combine` или оператор `expand`. Для более компактной записи в выражении для кинетической энергии делаем упрощения тригонометрических выражений, указав параметр `trig` в команде `combine`. Общие множители можно вынести оператором `factor`. С помощью оператора `collect` собираем в окончательном выражении слагаемые, содержащие $\dot{\varphi}$. Команда `declare`² пакета дифференциальных уравнений в частных производных `PDEtools` используется для того, чтобы производные функции обозначались при выводе на экран нижним индексом. Если производные по t нужно обозначить штрихом, используют команду `declare(prime=t)`.

Уравнение получено, поставленная задача решена, но, имея такой мощный аппарат, как **Maple**, трудно удержаться от поиска решения. Самый эффективным было бы получение аналитического решения (**Maple** в основном для этого и предназначен), но в данном случае такого нет. Решение ищем численно. В операторе `dsolve` указываем тип решения: `type=numeric`. Начальные данные выбираем произвольно. На экран

¹Замена переменных использована здесь потому, что в **Maple** нельзя дифференцировать по функции или производной функции. В программе 26 на с. 383 введена процедура универсального дифференцирования **Дифф**, у которой таких ограничений нет.

²Не путать с одноименной командой из пакета `codegen`, предназначенного для конвертации программ в другие языки: C, Fortran, MathML.

выводим зависимость угла и угловой скорости от времени. Первую кривую выводим немного утолщенной: `thickness=2`.

Заметим, что для получения уравнений Лагранжа по заданной функции Лагранжа $L = T - \Pi$ в **Maple** есть специальная функция `EulerLagrange` из пакета `VariationalCalculus`¹.

Программа 21 (P21.mws)

```
> restart; # Задача 37.
    Считываем процедуру Граф, с. 361
> read "C:\\graph.m";
    Неподвижные точки (нулевые скорости)
> v[C,1], v[C,2], v[A,2], v[K,2] := 0$4:
> Gr1 := Граф([K, A, P, B, C], [r$4], [Pi/2, Phi, Phi, 0],
>
    [w2, w2, w3, w3]):
> eq1 := Gr1[1]; # Уравнение в проекции на x
> eq2 := Gr1[2]; # Уравнение в проекции на y
    eq1 := 0 = vK,1 - r w2 - r sin(Φ) w2 - r sin(Φ) w3
    eq2 := 0 = r cos(Φ) w2 + r cos(Φ) w3 + r w3
> Gr2 := Граф([A, B, C], [2*r, r], [Phi, 0], [w0, w3]):
> eq3 := Gr2[1]; # Уравнение в проекции на x
> eq4 := Gr2[2]; # Уравнение в проекции на y
    eq3 := 0 = vA,1 - 2 r sin(Φ) w0
    eq4 := 0 = 2 r cos(Φ) w0 + r w3
> S := solve({eq1, eq2, eq3, eq4}, {w2, w3, v[K,1], v[A,1]});
> assign(S):
    S := {w3 = -2 cos(Φ) w0, w2 = 2 cos(Φ) w0 + 2 w0,
    vK,1 = 2 r cos(Φ) w0 + 2 r w0 + 2 r sin(Φ) w0, vA,1 = 2 r sin(Φ) w0}
> J2 := m[2]*r^2/2: # Момент инерции
> T1 := m[1]*v[K,1]^2/2: # Кинетическая энергия
> T2 := m[2]*v[A,1]^2/2 + J2*w2^2/2:
> T := factor(combine(T1+T2, trig));
    T :=  $\frac{1}{2}r^2 w0^2(8 m_1 + 8 m_1 \cos(\Phi) + 4 m_1 \sin(2\Phi) + 8 m_1 \sin(\Phi) + 5 m_2$ 
     $- m_2 \cos(2\Phi) + 4 m_2 \cos(\Phi))$ 
```

¹Автор благодарит студента МЭИ А. Боброва за идею использования этого пакета.

```

> z1:=diff(T,w0):
> z2:=diff(T,Phi):
> with(PDEtools):
> declare(pfi(t)):
> Phi:=phi(t);
> w0:=diff(Phi,t);

```

pfi(t) will now be displayed as pfi

$$\Phi := \phi(t)$$

$$w0 := \phi_t$$

```

> Q:=-simplify(M*w2/w0);# Обобщенная сила

```

```

> Уравн:=collect(diff(z1,t)-z2,w0)=Q;

```

$$Q := -2 M (\cos(\phi(t)) + 1)$$

$$\begin{aligned} \text{Уравн} := & r^2(-4 m_1 \sin(\phi(t)) + 4 m_1 \cos(2 \phi(t)) + 4 m_1 \cos(\phi(t)) + \\ & m_2 \sin(2 \phi(t)) - 2 m_2 \sin(\phi(t))) \phi_t^2 + r^2 \phi_{t,t} (8 m_1 + 8 m_1 \cos(\phi(t)) \\ & + 4 m_1 \sin(2 \phi(t)) + 8 m_1 \sin(\phi(t)) + 5 m_2 - m_2 \cos(2 \phi(t)) \\ & + 4 m_2 \cos(\phi(t))) = -2 M (\cos(\phi(t)) + 1) \end{aligned}$$

```

> r:=1: m[1]:=1: m[2]:=2: M:=1:

```

```

> НачУсл:=phi(0)=Pi/2,D(phi)(0)=0:

```

```

> Sol:=dsolve({Уравн,НачУсл},phi(t),type=numeric,

```

```

>                                     output=operator):

```

```

> assign(Sol):

```

```

> with(plots):

```

```

> with(plottools):

```

```

> График1:=odeplot(Sol,[t,phi(t)],0..9,thickness=2):

```

```

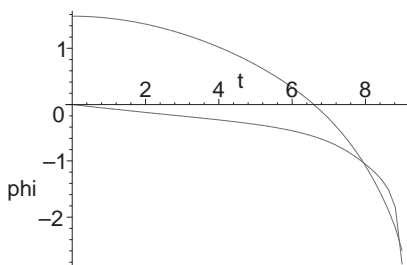
> График2:=odeplot(Sol,[t,w0],0..9):

```

```

> display(График1,График2);

```



7.15. Уравнение Лагранжа 2-го рода. Две степени свободы

В следующей программе вычисляются ускорения простой механической системы с двумя степенями свободы в случае, когда кинетическая энергия зависит только от обобщенных скоростей и не зависит от обобщенных координат. Система уравнений движения при этом получается с постоянными коэффициентами и имеет простое решение.

Для получения искомого ускорения в программу необходимо ввести все кинематические соотношения, выражения для кинетической энергии. Эта часть работы выполняется вручную. Вычисление обобщенных сил и решение системы берет на себя программа. В качестве примера рассмотрена задача 41 на с. 108.

Массы тел в программе введены как переменные с индексом только для того, чтобы вывод результатов имел привычный вид (нижний индекс A, B или C). Если не обращать на это внимание, то можно использовать переменные $m[A]$, $m[B]$, $m[C]$. Уравнения Лагранжа здесь условные в том смысле, что дифференцирования по времени частных производных $\partial T/\partial w_B$ и $\partial T/\partial w_C$ можно не производить, формально заменяя скорости на ускорения. Поэтому в начале решения v_1 и v_2 — скорости, а в конце это уже ускорения, с оговоркой на условность замены. В программе 25, с. 379, дано решение подобной задачи в общем случае, без всяких условностей, но эта задача немного сложнее, так как кинетическая энергия зависит не только от обобщенных скоростей, но и от координат, и при дифференцировании $\partial T/\partial w_B$ и $\partial T/\partial w_C$ по времени простой заменой скоростей на ускорения здесь обойтись нельзя.

Программа 22 (P22.mws)

```
> restart;# Задача 41
> JB:=m[B]*Rb^2/2: JC:=m[C]*Rc^2/2:# Момент инерции
> wB:=v1/Rb: wC:=(v2+v1)/Rc:      # Угловая скорость
  Кинетическая энергия
> T:=m[A]*v1^2/2+JB*wB^2/2+JC*wC^2/2+m[C]*v2^2/2:
> N:=-m[A]*g*v1-m[C]*g*v2: # Мощность
> Q1:=diff(N,v1): # Обобщенная сила 1
> Q2:=diff(N,v2): # Обобщенная сила 2
  Уравнения Лагранжа
> u1:=collect(diff(T,v1),v1)=Q1;
> u2:=collect(diff(T,v2),v1)=Q2;
> solve({u1,u2},{v1,v2});
```

$$\begin{aligned}
 u1 &:= (m_A + \frac{1}{2} m_B + \frac{1}{2} m_C) v1 + \frac{1}{2} m_C v2 = -m_A g \\
 u2 &:= \frac{1}{2} m_C v1 + \frac{3}{2} m_C v2 = -m_C g \\
 \{v2 = -\frac{2g(m_A + m_B + m_C)}{6m_A + 3m_B + 2m_C}, v1 = -\frac{2g(3m_A - m_C)}{6m_A + 3m_B + 2m_C}\}
 \end{aligned}$$

Решим задачу 42, с. 110, на составление уравнения движения системы с двумя степенями свободы. Эта задача не может быть решена по алгоритму предыдущей программы. Здесь кинетическая энергия зависит от обобщенной координаты и уравнения будут сложнее. Вначале составляем кинематические графы. Используем подпрограмму **Граф**, с. 361, которая автоматически генерирует по два уравнения для скоростей в каждом графе, независимо от его длины. Графы в программе несколько отличаются от графов в решении на с. 110. Их три, а не четыре, но скорости получают те же значения. Трем графам соответствует шесть уравнений, однако для решения задачи требуется выразить через обобщенные скорости только пять неизвестных. В действительности требуются только четыре уравнения $G[1][1]$, $G[2][1]$, $G[2][2]$, $G[3][2]$ для неизвестных $v[A, 1]$, $v[A, 2]$, $w1$, $w2$, или пять (вместе с уравнением $G[1][1]$), если искать вертикальную скорость $v[B, 2]$ точки B , которая, очевидно, равна нулю. Однако выискивать в системе шести уравнений нужные уравнения не требуется — зависимые уравнения (таких здесь два) **Maple** выкинет самостоятельно.

Основная трудность при составлении уравнений Лагранжа 2-го рода в системе **Maple** возникает при дифференцировании по обобщенной координате, являющейся функцией времени, или по скорости.

Так, если, например, задано выражение $y:=x(t)+1$, то операция $\text{diff}(y, x(t))$ вызывает сообщение об ошибке: *Error, invalid input: diff received x(t), which is not valid for its 2nd argument*, указывающее, что выражение $x(t)$ не может быть вторым аргументом оператора diff . Обойти это ограничение просто. Достаточно оператором subs^1 заменить переменную, по которой надо дифференцировать на временную, продифференцировать по этой переменной, а потом выполнить обратную подстановку. Это и реализовано в процедуре **Дифф**. Русские буквы в названии использованы для того, чтобы избежать путаницы между своими и встроенными в **Maple** операторами и процедурами. Эта

¹Оператор подстановки широко используется в **Maple** для преобразования выражений. В системе помощи из меню Help по цепочке Introduction → Programming → Operations → Substitution можно найти различные варианты подстановки: algsubs , applyrule , eval , evalindent , patmatch , powsubs , subsindent , subsup , trigsub .

процедура еще будет неоднократно использована, поэтому ее удобно записать в отдельный файл, а потом вызывать командой `read`.

Программа 23 (P23.mws)

```
> restart; # Оператор универсального дифференцирования
> Дифф:=proc(y, x):
>   diff(subs(x=z, y), z): subs(z=x, %):
> end proc:
> save Дифф, "C:\\diff.m"; # Сохраняем на диске
```

Для линеаризации уравнений использована процедура `Тейлор`, которая рассматривает уравнения как функцию обобщенных координат и скоростей и раскладывает их в ряд Тейлора¹, используя удобный оператор `mtaylor`. Параметр `Order` определяет число удерживаемых членов (по умолчанию 6)². При этом обобщенные координаты и скорости приходится заменять временными переменными `V[i]`, иначе в ряд по функциям разложить нельзя. Линеаризация будет еще неоднократно применяться в следующих задачах, поэтому имеет смысл командой `save` записать оператор `Тейлор` в отдельный файл, а потом при необходимости вызывать оператором `read` (с. 387, 389).

Программа 24 (P24.mws)

```
> restart; # Оператор линеаризации
> Тейлор:=proc(u, var) local V, N;
>   N:=1..nops(var);
>   subs(seq(var[i]=V[i], i=N), u);
>   mtaylor(%, [seq(V[i], i=N)]);
>   subs(seq(V[i]=var[i], i=N), %);
> end proc:
> save Тейлор, "C:\\teylor.m"; # Сохраняем на диске
```

¹*Taylor Brook* (1685–1731) — английский математик и философ.

²Для функции одной переменной есть оператор `taylor`, однако этот оператор выдает еще и порядок остаточного члена, иногда мешающий расчетам. Например,

```
> taylor(sin(x), x);
```

$$x - \frac{1}{6}x^3 + \frac{1}{120}x^5 + O(x^6)$$

Уравнения Лагранжа Eq1, Eq2 выводятся на экран в преобразованной форме. Оператор `collect(*, w, factor)` собирает слагаемые с угловой скоростью w , раскладывая при этом коэффициенты при степенях w на множители. Вместо w почти с таким же успехом можно выбрать vs .

После линеаризации получается система (3.30). Чтобы получить частоту колебаний, надо исключить \dot{s} . Сделать это можно разными способами. В данном случае решаем систему линейных уравнений относительно ускорений \ddot{s} и $\ddot{\varphi}$. Одно из решений будет иметь зависимость $\ddot{\varphi}$ от φ . Какое решение — первое или второе — угадать трудно. Здесь **Maple** непредсказуема. Запускаем программу пошагово и видим, что это Реш[2]. Далее оператором `rhs` выделяем правую часть полученного выражения, делим на φ и получаем (с обратным знаком) квадрат собственной частоты.

В конце программы анализируется предельное свойство полученного решения (частоты малых колебаний) при $m_1 \rightarrow \infty$, что соответствует анализу колебательных качений цилиндра по внутренней поверхности неподвижного цилиндра.

Программа 25 (P25.mws)

```
> restart; #Уравнение движения. Две степени свободы
> with(PDEtools): declare(s(t), phi(t)):
> read "C:\\diff.m"; # Программа 23, с. 378
> read "C:\\teylor.m"; # Программа 24, с. 378
> read "C:\\graph.m"; # Программа 12, с. 361
  Обозначения
> f:=phi(t): vs:=diff(s(t),t): w:=diff(f,t):
  Известны скорости
> v[P,1]:=0: v[P,2]:=0: v[B,1]:=vs:
> G[1]:=Граф([P,B],[R],[Pi/2],[w1]):
> G[2]:=Граф([A,B],[R-r],[f+Pi/2],[w]):
> G[3]:=Граф([A,K,B],[r,R],[f-Pi/2,f+Pi/2],[w2,w1]):
> Реш:=solve({seq(op(G[i]),i=1..3)},
>             {v[B,2],v[A,1],v[A,2],w1,w2});
> assign(Реш);
```

$$\text{Решение} := \{v_{B,2} = 0, w1 = -\frac{s_t}{R}, w2 = \frac{-\phi_t R + \phi_t r - s_t}{r},$$

$$v_{A,1} = s_t + \cos(\phi)\phi_t R - \cos(\phi)\phi_t r, v_{A,2} = \sin(\phi)\phi_t R - \sin(\phi)\phi_t r\}$$

```
> Va:=simplify(sqrt(v[A,1]^2+v[A,2]^2)):
> J1:=m[1]*R^2: J2:=m[2]*r^2/2:
```



```
> T:=collect(1/2*m[1]*vs^2+1/2*J1*w1^2+1/2*m[2]*Va^2+
> 1/2*J2*w2^2,w,factor);
```

$$T := \frac{3}{4}m_2(-R+r)^2\phi_t^2 - \frac{1}{2}s_t m_2(2\cos(\phi) + 1)(-R+r)\phi_t + \frac{1}{4}s_t^2(4m_1 + 3m_2)$$

```
> P:=m[2]*g*(R-(R-r)*cos(f)): #Потенциальная энергия
> L:=T-P: #Кинетический потенциал
Уравнения Лагранжа
```

```
> Eq1:=collect(diff(Дифф(L,vs),t)-Дифф(L,s(t)),
> diff(w,t),factor);
```

$$Eq1 := -\frac{1}{2}m_2(2\cos(\phi) + 1)(-R+r)\phi_{t,t} - m_2\sin(\phi)\phi_t^2 R + \\ m_2\sin(\phi)\phi_t^2 r + 2s_{t,t}m_1 + \frac{3}{2}s_{t,t}m_2$$

```
> Eq2:=factor(diff(Дифф(L,w),t)-Дифф(L,f));
```

$$Eq2 := \frac{1}{2}m_2(-R+r)(3\phi_{t,t}r - s_{t,t} \\ - 2\cos(\phi)s_{t,t} - 2g\sin(\phi) + 2s_t\sin(\phi)\phi_t - 3\phi_{t,t}R)$$

```
> Order:=2:
```

```
> U1:=Тейлор(Eq1,[diff(s(t),t,t),diff(w,t),phi(t)]);
```

```
> U2:=Тейлор(Eq2,[diff(s(t),t,t),diff(w,t),phi(t)]);
```

$$U1 := (2m_1 + \frac{3}{2}m_2)s_{t,t} - \frac{3}{2}m_2(-R+r)\phi_{t,t}$$

$$U2 := -\frac{3}{2}m_2(-R+r)s_{t,t} + \frac{1}{2}m_2(-R+r)(3r - 3R)\phi_{t,t} - \\ m_2(-R+r)g\phi$$

```
> Реш:=solve({U1,U2},{diff(w,t),diff(s(t),t,t)});
```

$$\text{Реш} := \{s_{t,t} = \frac{1}{2} \frac{m_2 g \phi}{m_1}, \phi_{t,t} = \frac{1}{6} \frac{g \phi (4m_1 + 3m_2)}{m_1(-R+r)}\}$$

```
> -rhs(Реш[2])/f);
```

$$-\frac{1}{6} \frac{g(4m_1 + 3m_2)}{(-R+r)m_1}$$

```
> k:=sqrt(%); # Частота колебаний
```

$$k := \frac{1}{6} \sqrt{-\frac{6g(4m_1 + 3m_2)}{(-R+r)m_1}}$$

```
> limit(k,m[2]=infinity);
```

$$\frac{1}{3} \sqrt{-\frac{6g}{-R+r}}$$

7.16. Функция Рауса

Для задач, имеющих циклические координаты, введение в решение функции Рауса позволяет упростить систему уравнений движения, понизив ее порядок. Решим в качестве примера задачу 43 на с. 113.

Выражения для кинетической и потенциальной энергии находятся вручную, обычным образом, через выбранные обобщенные координаты и скорости. Программа выполняет рутинную часть работы по составлению уравнений движения. Обобщенные координаты задачи — $x(t)$ и $\varphi(t)$. В самом начале программы к ним применяется оператор `declare` из пакета `PDEtools`. Это действие не обязательное и сделано для более привычного и компактного вида производных при выводе результатов на экран или печать.

Функция Рауса R , полученная в результате всех преобразований и подстановок, получается весьма громоздкой. Упростить ее внешний вид позволяют два оператора `simplify` и `collect`, примененные последовательно. Оператор `simplify` приводит подобные члены, сокращает множители, сворачивает выражения типа $\sin^2 \varphi + \cos^2 \varphi$ и др. Оператор `collect` собирает коэффициенты при степенях своего аргумента (здесь — при $\dot{\varphi}$), а очень удобная опция `factor` автоматически раскладывает полученные коэффициенты на скобки, или, по крайней мере, выносит из них общие множители.

С получением функции Рауса основная задача программы оказывается выполненной. Однако было бы интересно составить и проинтегрировать при каких-нибудь начальных условиях уравнение Рауса, тем более что в **Maple** это не очевидная задача и часто вызывает затруднения. Оператор `dsolve` является встроенным в **Maple** и не требует вызова какого-либо специального пакета. Полученное дифференциальное уравнение нелинейное, для его решения используем численный метод, указав опцию `type=numeric` и вид вывода полученного решения `output=operator`. Особенность решаемой задачи состоит в том, что решение для $\varphi(t)$ необходимо использовать далее для нахождения циклической координаты $x(t)$ с помощью интеграла (3.35), с. 115. Начальные значения в задаче Коши выбираем произвольными. Для позиционной координаты $\varphi(t)$ начальные условия вставляются в виде опции непосредственно в оператор `dsolve`, а начальные значения для циклической координаты $x(t)$ (здесь они выбраны нулевыми) используются при определении констант интегрирования. Чтобы к решению дифференциального уравнения можно было бы обращаться как к функции, например вызывать значение `x(1.2)`, применяется операция назначения решения `assign(Sol)`. Графики решений (обобщенные координаты $x(t)$ и $\varphi(t)$ и угловая скорость $\omega(t)$), построенные оператором `odeplot`, выводятся все вместе оператором `display`. При этом некоторые кривые решений получаются не очень четкими. Увеличивая

число `numpoints` выводимых точек, кривые можно сделать более гладкими. По умолчанию `numpoints` равно 50.

Имея решение, легко построить анимированную модель, наглядно иллюстрирующую характер явления. Отдельные кадры движения создаются в цикле по `k`. Затем они последовательно (оператор повторения `seq`) выдаются на экран оператором `display`. Каждый кадр состоит из изображений трех объектов — цилиндра, точки и бруска. Используются графические структуры, дублирующие в какой-то степени объекты пакета `plottools`, с. 279. Графические структуры выводятся на экран оператором `PLOT` и не требуют подключения специального графического пакета. Соответствующую справку можно получить кнопкой `F1` по цепочке `Graphics... ->2-D... -> structure1`. Есть всего четыре объекта 2-D графики: `CURVES`, `POINTS`, `POLYGONS` и `TEXT`. Опции для графических структур также пишутся заглавными буквами: `LINestyle`, `SCALING`, `THICKNESS` и др. Объекты можно раскрасить. Цвет, задаваемый в режиме `HUE`, определяется дробной частью аргумента, так, например, `COLOR(HUE, 0.3)` — зеленый, `COLOR(HUE, 0.7)` — синий, `COLOR(HUE, 0.0)`, `COLOR(HUE, 0.99)` — красный. Цветовой режим `RGB` задается перемешиванием трех цветов: красного (`Red`), зеленого (`Green`) и синего (`Blue`). Каждый цвет задается отдельным числом от 0.0 до 1.0. Например: `COLOR(RGB, 0.0, 0.0, 1.0)` — синий цвет, `COLOR(RGB, 1.0, 1.0, 1.0)` — белый, `COLOR(RGB, 1.0, 1.0, 0.0)` — желтый.

Интересно поэкспериментировать с полученным решением. Можно посмотреть на характер решения при различных начальных значениях, величинах момента и масс. Уменьшая, например, массу `m3`, заметим неадекватное поведение решения. Наблюдается необъяснимый на первый взгляд скачкообразный рост угла φ . Объяснение этого факта дает аналитическое решение. Получим решение уравнения Рауса в рядах, задав опцию `type='series'`. Величину массы `m3` оставим параметром решения. Зададим начальный угол $-\pi/2$:

```
> Order:=3:
```

```
> dsolve(Уравн, НачУсл, phi(t),
```

```
>                                     type='series', output=operator);
```

Параметр `Order` задает точность (число членов ряда). Получим

$$\phi = -\frac{\pi}{2} - \frac{4(10 + m_3)}{m_3(m_3 + 30)} t^2 + O(t^3).$$

Масса `m3` содержится в знаменателе, что и объясняет замеченный эффект. Увеличение числа членов ряда не меняет этой особенности.

¹Краткое описание см. на с. 291.

Программа 26 (P26.mws)

```

> restart;# Задача 43
> with(PDEtools): declare(x(t),phi(t)):
> read "C:\\diff.m";
  Обозначения
> f:=phi(t): vx:=diff(x(t),t): w:=diff(f,t):
  Массы и момент инерции
> m1:=4*m: m2:=m: m3:=3*m: J3:=m3*r^2/2:
  Координаты точки m2
> x2:=x(t)-r*f+r*cos(f):
> y2:=r*sin(f):
  Скорости
> vx2:=diff(x2,t):
> vy2:=diff(y2,t):
  Кинетическая энергия
> T:=m1*vx^2/2 + m2*(vx2^2+vy2^2)/2 +
>
  J3*w^2/2 + m3*(vx-r*w)^2/2:
  Потенциальная энергия
> P:=M*f+m2*g*y2:
  Функция Лагранжа
> L:=T-P:
  Обобщенный импульс
> px:=Дифф(L, vx):
  Обобщенная циклическая скорость
> X1:=solve(px=C, vx):
  Функция Рауса
> R:=collect(simplify(C*X1-subst(vx=X1,L)),w,factor);
  
$$R := -\frac{1}{16} m r^2 (35 + 8 \sin(\phi) + \cos(\phi)^2) \phi_t^2 + \frac{1}{8} r C (4 + \sin(\phi)) \phi_t$$

  + 
$$\frac{1}{16} \frac{C^2 + 16 m^2 g r \sin(\phi) + 16 M \phi m}{m}$$

  Уравнение Рауса
> Уравн:=simplify(diff(Дифф(R,w),t) - Дифф(R,f));

```

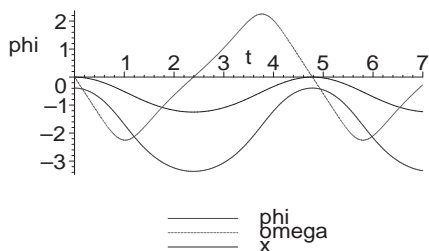
```

Уравн := - 1/2 m r^2 cos(phi) phi^2 + 1/8 m r^2 cos(phi) phi^2 sin(phi) - 35/8 m r^2 phi_t, t
- m r^2 phi_t, t sin(phi) - 1/8 m r^2 phi_t, t cos(phi)^2 - m g r cos(phi) - M
    Момент (Нм), масса (кг), ускорение (м/с2), радиус (м)
> M:=4: m:=2: g:=9.81: r:=1:
    Начальный угол и начальная угловая скорость
> f0:=-Pi/8: f10:=0:
> НачУсл:=phi(0)=f0, D(phi)(0)=f10:
    Решение задачи Коши отдельно для угла phi(t)
> Sol:=dsolve({Уравн, НачУсл}, phi(t),
>
>                                     type=numeric, output=operator):
> assign(Sol):
> with(plots): with(plottools):#Графические пакеты
> T2:=7:#Выбираем некоторый интервал времени
    Зависимость угла f от времени t
> Рис1:=odeplot(Sol, [t, f], 0..T2, color=blue,
    Зависимость угловой скорости w от времени t
>
>                                     thickness=2, legend="phi", numpoints=200):
> Рис2:=odeplot(Sol, [t, w], 0..T2, color=red,
>
>                                     linestyle=3, legend="omega", numpoints=200):
> X_:=int(diff(R, C), t)+C2;
    Определяем константы по начальным условиям для x
> eq1:=simplify(subs(f=f0, t=0, X_))=0:
> eq2:=subs(f=f0, w=f10, diff(R, C)):
> Решение:=solve({eq1, eq2}, {C, C2}):
> assign(Решение);
    Решение отдельно для координаты x
> Рис3:=plot(X_, t=0..T2, color=black, legend="x"):
> display(Рис1, Рис2, Рис3);

```

$$X_- := -\frac{1}{8} \cos(\phi) + \frac{\phi}{2} + \frac{Ct}{16} + C2$$

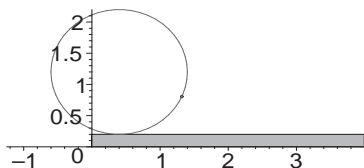
$$\text{Решение} := \{C = 0, C2 = \frac{1}{8} \cos(\frac{\pi}{8}) + \frac{\pi}{16}\}$$



```

> a:=4:b:=0.2:#Размеры бруска
> ЧислоКадров:=48:
> Шаг:=4.8/ЧислоКадров:
> for k to ЧислоКадров do
>   t:=k*Шаг:
>   F:=phi(t): x0:=evalf(X_):
>   Цилиндр:=circle([x0-r*F, r+b], r, color=red):
>   Брусок:=POLYGONS([[x0, b], [x0, 0], [x0+a, 0],
>                       [x0+a, b]]), STYLE(PATCH), COLOR(HUE, 0.3):
>   Точка:=POINTS([x0-r*F+r*cos(F), r+r*sin(F)+b],
>                  SYMBOL(CIRCLE), COLOR(RGB, 0, 0, 1)):
>   Кадр[k]:=PLOT(Цилиндр, Брусок, Точка):
> od:
> display(seq(Кадр[i], i=1..ЧислоКадров),
>          insequence=true, scaling=constrained);

```



7.17. Уравнения Гамильтона

В Maple имеется оператор `hamilton_eqs` для получения уравнений движения в форме Гамильтона по заданной функции Гамильтона H . Решим задачу 44, с. 116. Выразим кинетическую энергию цилиндра $T1$ через угловую скорость w . Скорость точки A найдем с помощью программы **Граф**. Эта программа составляет два уравнения $G[1]$, $G[2]$, из которых оператор `solve` определяет компоненты скорости $v_{Ax} = v[A, 1]$, $v_{Ay} = v[A, 2]$. При вычислении обобщенной силы использован оператор `expand`, вынуждающий сократить угловую скорость w в числителе и знаменателе. Как ни странно, Maple самостоятельно не производит эту очевидную операцию. Вместо `expand` в данном случае с таким же успехом можно применить `simplify` или `combine`. Потенциальную энергию в данном случае удобнее вычислять как интеграл от обобщенной силы.

В операторе `hamilton_eqs` обобщенный импульс обозначается $p1$, обобщенная координата — $q1$. Оператор можно использовать и для задач с большим числом степеней свободы. В этом случае импульсы и координаты нумеруются просто: $p1$, $p2$, $p3$, и т. д. $q1$, $q2$, $q3$, и т. д.

Программа 27 (P27.mws)

```
> restart; # Задача 44
> read "C:\\graph.m"; # Программа 12, с. 361
> T1:=(3/4)*m[1]*R^2*w^2:
> v[P,1],v[P,2]:=0$2:
> G:=Граф([P,O,A],[R,R],[Pi/2,q1],[w,w]):
> Решение:=solve({G[1],G[2]},{v[A,1],v[A,2]});
> assign(Решение);
> Va:=sqrt(v[A,1]^2+v[A,2]^2):
> T2:=m[2]*Va^2/2:
> T:=simplify(T1+T2): # Кинетическая энергия
> Q:=expand((M*w-m[2]*g*v[A,2]+F*v[A,2])/w):
> P:=-int(Q,q1): # Потенциальная энергия
    Решение := {vA,1 = -Rw - R sin(q1)w, vA,2 = R cos(q1)w}
> L:=T-P; # Кинетический потенциал
L :=  $\frac{1}{4}R^2w^2(3m_1 + 4m_2 + 4m_2 \sin(q1)) + Mq1 - m_2gR \sin(q1) + FR \sin(q1)$ 
> eq:=p1=diff(L,w):
> w:=solve(eq,w):
> H:=p1*w-L: # Функция Гамильтона
> with(DEtools):with(PDEtools):
```

```
> declare(q1(t), p1(t));
> Z:=hamilton_eqs(H); # Канонические уравнения
```

$$Z := [p1_t = \frac{4p1^2 m_2 \cos(q1)}{R^2(3m_1 + 4m_2 + 4m_2 \sin(q1))^2} + M - m_2 g R \cos(q1) + F R \cos(q1),$$

$$q1_t = \frac{2p1}{R^2(3m_1 + 4m_2 + 4m_2 \sin(q1))}], [p1, q1]$$

7.18. Паллограф

На с. 123 приведена задача 46 о малых колебаниях паллографа (прибора для записи вибраций) и два способа ее решения — обычный, на составление уравнения Лагранжа, и способ с множителями Лагранжа. Здесь даны две программы для каждого способа в отдельности. Сначала решим задачу по способу 1. Угловую скорость стержня 2 и скорость точки C вычислим по программе **Граф**. Угол ψ , необходимый для графов, выразим из треугольника OAB с использованием обратной тригонометрической функции. В общем, это нежелательная операция, так как \arccos функция неоднозначная, но в данном случае в предположении малости угла φ ошибок не будет. Однако с помощью оператора **assume** пришлось указать область изменения угла $0 < \varphi < \pi/6$. Кроме того, в решении появляется извлечение корня $\sqrt{c^2}$. Поэтому в операторе **assume** указано, что c — положительная величина. В этом случае $\sqrt{c^2} = c$ и выражения упрощаются. Но все равно, окончательное уравнение Лагранжа получается очень громоздким (на несколько страниц), и здесь бессмысленно его приводить. А вот линейный вариант его будет достаточно простым.

В аргументе процедуры **Тейлор** записан список переменных, по которым производится линеаризация. Последовательность здесь существенна. Сначала надо записывать производные функции в убывающем порядке, а потом уже саму функцию $\varphi(t)$. Связано это с тем, что в таком порядке будет работать оператор подстановки **subs**. А этот оператор спешит сделать указанную замену, поэтому если заменить функцию константой (а именно так и делается в процедуре **Тейлор**), то **Maple** тут же ее продифференцирует, получит, естественно, ноль, и подставлять производные будет уже некуда.

Программа 28 (P28.mws)

```
> restart; # Задача 46. Паллограф
> read "C:\\graph.m"; # Программа 12, с. 361
> read "C:\\diff.m"; # Программа 23, с. 378
> read "C:\\teylor.m"; # Программа 24, с. 378
```

```

> assume(phi(t)<Pi/6,phi(t)>0,c>0);
> interface(showassumed=0):
> with(PDEtools): # Для оператора declare
> declare(phi(t)): # Компактная запись функц.и произв.
> v[0,1],v[0,2],v[B,1]:=0$3:# Условия на скорости
> psi(t):=arccos((a/c)*sin(phi(t))):
> w1:=diff(phi(t),t): # Угловая скорость 1
> G1:=Граф([0,A,B],[a,c],[Pi/2+phi(t),psi(t)],
> [w1,w2]):
> w2:=simplify(solve(G1[1],w2)): # Угловая скорость 2
> G2:=Граф([0,A,C],[a,b],[Pi/2+phi(t),Pi+psi(t)],
> [w1,w2]):
> Решение:=solve({G2[1],G2[2]},{v[C,1],v[C,2]}):
> assign(Решение):
> Vc:=sqrt(v[C,1]^2+v[C,2]^2): # Скорость точки C
> T:=simplify(m*Vc^2/2): # Кинетическая энергия
> Q:=expand((-m*g*v[C,2])/w1): # Обобщенная сила
    Уравнение Лагранжа 2-го рода
> eq:=diff(Дифф(T,diff(phi(t),t)),t)-Дифф(T,phi(t))-Q:
> Order:=2:
    Линеаризация
> Eq:=collect(Тейлор(eq,[diff(phi(t),t$2),
> diff(phi(t),t),phi(t)]),phi(t),factor);
    
$$Eq := \frac{m a g (-c^2 + b a) \phi}{c^2} + \frac{m a^2 (b + c)^2 \phi_{t,t}}{c^2}$$

> sqrt(-solve(Eq,diff(phi(t),t$2))/phi(t)); # Частота
    
$$\sqrt{\frac{g(-c^2 + ba)}{a(b^2 + c^2 + 2cb)}}$$


```

Рассмотрим программу расчета частоты малых колебаний паллографа по способу 2 с использованием множителей Лагранжа, с. 125. Программа в целом очень похожа на предыдущую. Для определения скорости с помощью процедуры Граф вызывается файл graph.m, процедура линеаризации Тейлор записана в файл teylor.m, а процедура Дифф дифференцирования по функции (она потребуется один раз при определении обобщенной силы) — в файл diff.m. Все эти файлы находятся в корне диска C, но в общем они могут располагаться где угодно. Для того чтобы уравнения Лагранжа можно было сформировать в цикле, введен список обобщенных координат φ . Теперь $\varphi = \varphi_1$,

$\psi = q_2$, $y = q_3$. Также в цикле линеаризуем все пять уравнений — три уравнения Лагранжа и два уравнения связи. Угол α исключаем, решив уравнение Eq[5] (при этом Maple выражает и $\ddot{\alpha}$ через $\ddot{\varphi}$), а неопределенные множители λ_1 и λ_2 исключаем с помощью оператора `eliminate`.

Программа 29 (P29.mws)

```
> restart;# Задача 46. Паллограф. Множители Лагранжа
> read "C:\\diff.m"; # Программа 23, с. 378
> read "C:\\teylor.m";# Программа 24, с. 378
> read "C:\\graph.m"; # Программа 12, с. 361
> with(PDEtools):
> declare(phi(t),psi(t),alpha(t),y(t)):
> y1:=diff(y(t),t): # Скорость ползуна
> v[B,1]:=0:v[B,2]:=y1:
> G:=Граф([C,B],[b+c],[psi(t)],[w2]);
  G := [0 = vC,1 - (b + c) sin(ψ) w2, yt = vC,2 + (b + c) cos(ψ) w2]
> Решение:=solve({G[1],G[2]},{v[C,1],v[C,2]}):
> assign(Решение);
> Vc:=sqrt(v[C,1]^2+v[C,2]^2):# Скорость точки C
> T:=simplify(m*Vc^2/2); # Кинетическая энергия
      T :=  $\frac{1}{2} m (w_2^2 b^2 + 2 w_2^2 b c + w_2^2 c^2 + y_t^2 - 2 y_t \cos(\psi) w_2 b - 2 y_t \cos(\psi) w_2 c)$ 
> eq[4]:=a*cos(phi(t))+c*sin(psi(t))-y(t):# f1 связь
> eq[5]:=a*sin(phi(t))-c*cos(psi(t)): # f2 связь
> N:=-m*g*v[C,2]: # Мощность
> Q[1]:=diff(N,w1):# Обобщенные силы
> Q[2]:=diff(N,w2):
> Q[3]:=Дифф(N,y1):
> T:=subs(w2=diff(psi(t),t),T):
> q:=[phi(t),psi(t),y(t)]:# Обобщенные координаты
> for i to 3 do # Уравнения Лагранжа
>   eq[i]:=diff(Дифф(T,diff(q[i],t)),t)-
>     Дифф(T,q[i])-Q[i]-lambda[1]*Дифф(eq[4],q[i])-
>     lambda[2]*Дифф(eq[5],q[i]);
> od:
> psi(t):=Pi/2-alpha(t):
```

```

> Order:=2:
> z:=diff(alpha(t),t$2),diff(alpha(t),t),alpha(t),
>       diff(phi(t),t$2),diff(phi(t),t),phi(t),y1:
> for i to 5 do # Линеаризуем
>   Eq[i]:=Тейлор(eq[i],[z]);
> od;

```

$$Eq_1 := -\lambda_2 a + \lambda_1 a \phi$$

$$Eq_2 := -\lambda_2 c + \frac{1}{2} m (-2b^2 - 4cb - 2c^2) \alpha_{t,t} + (-\lambda_1 c + m g (-b - c)) \alpha$$

$$Eq_3 := m g + \lambda_1$$

$$Eq_4 := a + c - y$$

$$Eq_5 := a \phi - c \alpha$$

```

> alpha(t):=solve(Eq[5],alpha(t));
> Z:=eliminate({seq(Eq[i],i=1..3)},
>             {lambda[1],lambda[2]}):

```

$$\alpha := \frac{a \phi}{c}$$

Частота

```

> factor(sqrt(-solve(op(Z[2]),diff(phi(t),t$2))/
>                   phi(t)));

```

$$\sqrt{\frac{g(-c^2 + ab)}{a(b+c)^2}}$$

7.19. Моменты инерции поперечного сечения. Манлет

Существует несколько алгоритмов вычисления площади, координат центра тяжести и моментов инерции плоских фигур. Самый распространенный способ состоит в ручном разбиении фигуры на простейшие с последующим вычислением сумм. Для фигур, состоящих из небольшого числа простых составляющих, этот способ удобен, однако как способ надежной проверки ручного счета он не годится, так как ошибка в какой-либо характеристике отдельной составляющей не будет замечена. Чаще всего бывают ошибки со знаком центробежных моментов инерции и путаница в осях и в значениях b и h в формулах, взятых из таблиц 7, 8 стандартных фигур на с. 485–486.

В [36] приводится алгоритм вычисления интегрированием по площади. Обе функции, ограничивающие площадь сверху и снизу, задаются по точкам. У этого подхода есть ограничение — по оси y фигура должна быть правильной (т. е. ограничивающие функции однозначные).

Применим другой метод — метод интегрирования по контуру, основанный на замене двойного интеграла криволинейным¹.

Для площади фигуры $F = \int_D dF$ с помощью формулы Грина

$$\int_D \left(\frac{\partial P}{\partial y} - \frac{\partial Q}{\partial x} \right) dF = \oint_L P dx + Q dy$$

при $P = y$, $Q = -x$ получим²

$$F = \int_D dF = \frac{1}{2} \oint_L y dx - x dy.$$

Вычисляем криволинейный интеграл, используя на участке от точки i до точки $i + 1$ параметрическое представление прямой

$$x = (x_{i+1} - x_i)t + x_i, \quad y = (y_{i+1} - y_i)t + y_i, \quad 0 \leq t \leq 1. \quad (7.1)$$

Получаем формулу для площади фигуры, вычисленную по координатам точек ее контура:

$$F = \frac{1}{2} \sum_{i=1}^n (x_{i+1}y_i - x_iy_{i+1}). \quad (7.2)$$

Для контура, ограниченного ломаной L с угловыми точками (x_i, y_i) , $i = 1, \dots, m$, формула дает точный результат. Если контур сечения содержит криволинейные участки, то их можно приближенно заменить ломаной. Точность вычисления при этом зависит от числа введенных угловых точек и способа приближения кривой.

Для статического момента фигуры

$$S_y = \int_D x dF$$

¹Еще один метод предложили Кристалинский Р. Е. и Шапошников Н. Н. [34]. Метод реализован в системе Mathematica [19] и основан на интегрировании по площади некоторой логической функции, принимающей значение 1 внутри заданной фигуры и 0 — вне ее. Функция задается, например, в виде $F[x, y] = \text{Boole}[x+y>1, x>0, y>0]$. Статический момент вычисляется интегрированием $S_x = \text{Integrate}[y * F[x, y], \{x, -2, 2\}, \{y, -3, 3\}]$. Основное достоинство метода в том, что он легко переносится и на трехмерные тела. Метод не реализуется в Maple (здесь Maple проигрывает системе Mathematica!).

²Обход области принят по часовой стрелке. См. Фихтенгольц Г. М. Курс дифференциального и интегрального исчисления. Т. III, СПб.: Лань, 2009.

с помощью формулы Грина при $P = xy$, $Q = -x^2$ получим

$$S_y = \frac{1}{3} \oint_L xy dx - x^2 dy.$$

Вычисляем криволинейный интеграл, используя замену (7.1), и записываем формулу для статического момента

$$S_y = \frac{1}{6} \sum_{i=1}^n (x_{i+1}y_i - x_i y_{i+1})(x_i + x_{i+1}).$$

Таким же образом получим

$$S_x = \frac{1}{6} \sum_{i=1}^n (x_{i+1}y_i - x_i y_{i+1})(y_i + y_{i+1}).$$

Координаты центра тяжести имеют вид

$$x_c = \frac{S_y}{F}, \quad y_c = \frac{S_x}{F}.$$

Осевые моменты инерции рассчитываются по формулам

$$J_x = \frac{1}{12} \sum_{k=1}^n (y_{k+1}^2 + y_{k+1}y_k + y_k^2)(x_{k+1}y_k - x_k y_{k+1}),$$

$$J_y = \frac{1}{12} \sum_{k=1}^n (x_{k+1}^2 + x_{k+1}x_k + x_k^2)(x_{k+1}y_k - x_k y_{k+1}).$$

Центробежный момент инерции имеет вид

$$J_{xy} = \frac{1}{12} \sum_{k=1}^n \left(x_{k+1}y_{k+1} + x_k y_k + \frac{x_k y_{k+1} + y_k x_{k+1}}{2} \right) (x_{k+1}y_k - x_k y_{k+1}).$$

Центральные моменты инерции вычислим по формулам

$$J_{x_c} = J_x - y_c^2 F,$$

$$J_{y_c} = J_y - x_c^2 F,$$

$$J_{x_c y_c} = J_{xy} - x_c y_c F,$$

а главные моменты инерции $J_{\max, \min}$ — по формулам (4.4), с. 129.

Рассмотрим этот метод применительно к задаче 48 на с. 131. Принимаем обход контура по часовой стрелке. Нумеруем угловые точки



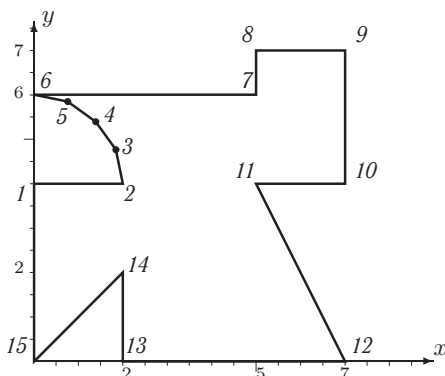


Рис. 418

контура (рис. 418). Дугу окружности заменяем ломаной, разбивая ее на четыре части. Координаты угловых точек на дуге вычисляем по формулам

$$x_{k+2} = \rho \cos(k\pi/8), \quad y_{k+2} = 2 + \rho \sin(k\pi/8), \quad k = 1, 2, 3.$$

Если взять $\rho = R$, то получим многоугольник, вписанный в окружность, и площадь получится в данном примере с завышением. Если $\rho = R/\cos(\pi/16)$, то многоугольник будет описывать окружность и площадь будет уменьшена¹.

Оператор `polygonplot` строит изображение контура, ограниченно-го точками, заданными в матрице `T`. Функция `add(c[i], i=1..N)` означает сумму $\sum_{i=1}^N c_i$. Точность вычислений зависит от числа точек на окружности и их расположения.

Программу оформим в виде маплета. Общая структура программы следующая:

```
MminModule:=module()
  SolvMMin:=proc()
    end proc;

  runMMin:=proc()
    end proc;
end module:
MminModule:-runMMin():
```

¹Для точной аппроксимации площади круга по этому методу необходимо $\rho = R\sqrt{\pi/(n \sin(\pi/n))}$, где n — число точек разбиения. При $n \rightarrow \infty$ согласно первому замечательному пределу $\rho \rightarrow R$.

В целом программа записана в модуль `MminModule` с двумя процедурами `SolvMmin` и `runMmin`. В первой производятся вычисления, во второй организован интерфейс программы (рис. 419). Программа запускается вызовом процедуры `runMmin` из модуля `MminModule`. Опишем структуру каждой процедуры. Схема процедуры вычислений `SolvMmin` имеет вид

```
SolvMmin:=proc()
use Maplets[Tools] in
  ввод данных оператором Get
  изображение контура Ris
  вычисление всех искомым величин (площадь, мом.ин.)
  занесение решения в таблицу оператором Set
end use;
Ris:
end proc;
```

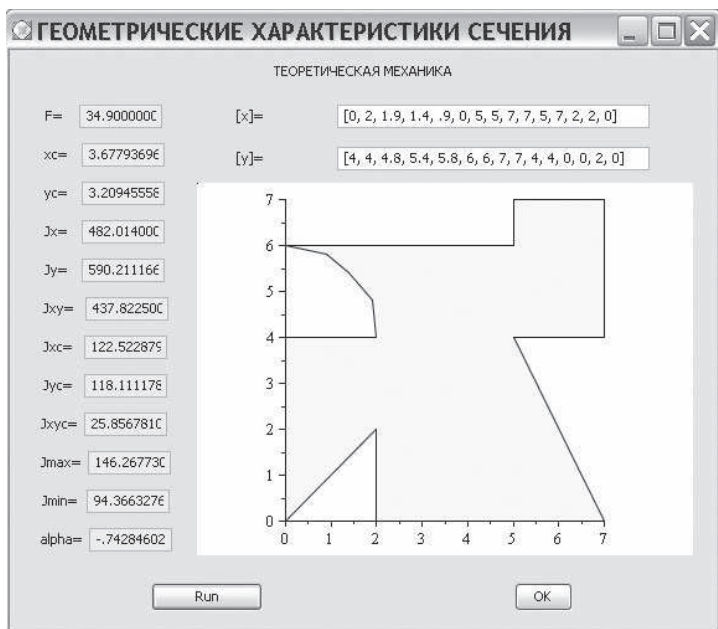


Рис. 419. P31.maplet

Последним в процедуре вызван рисунок `Ris`, поэтому при обращении `display(SolveMMin())` в процедуре интерфейса на дисплее появится изображение контура, заполненного желтым цветом (`Yellow`).

Опишем структуру процедуры `runMMin`, задающей интерфейс программы:

```
runMMin:=proc()
  переменные; константы;
  use Maplelets:-Elements in
  обозначения элементов интерфейса (a, c, T, p, mplt);
  mplt:= Maplet(
    Evaluate(),
    Window(tit, [txt, [BoxColumn(), [, ]], [, ]])
  ):
  Maplelets:-Display(mplt):
end use;
end proc;
```

Здесь обозначено:

```
tit - 'title'="ГЕОМЕТРИЧЕСКИЕ ХАРАКТЕРИСТИКИ СЕЧЕНИЯ",
txt - "ТЕОРЕТИЧЕСКАЯ МЕХАНИКА".
```

Предлагаемый метод позволяет рассчитывать и двусвязные фигуры. Пусть в трапеции $ABCD$ сделан треугольный вырез MNK (рис. 420). Сделаем разрез KC , соединяющий внутренний и внешний контур (рис. 421). При обходе фигуры внутренний контур будет пройден против часовой стрелки, и его площадь станет отрицательной, а разрез, пройденный дважды в противоположных направлениях, не внесет добавки в интеграл.

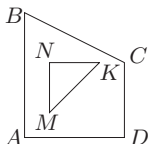


Рис. 420

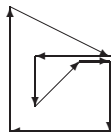


Рис. 421

Программа 30 (P30.mws)

```
> restart:with(plots):with(plottools):#Маплет
> MminModule:=module()
```

```

> export SolvMMin,runMMin:
> SolvMMin:=proc() local i, x, y, Ris, N, F, xc, yc, Jxy,
> Jx, Jy, Jxc, Jyc, Jxyc, Jmax, Jmin, alf, T, Z:
> use Maplets[Tools] in
> x:=Get('Tx'::list); # Координаты x
> y:=Get('Ty'::list); # Координаты y
> x:=[op(x),x[1]]; # Замыкание контура
> y:=[op(y),y[1]];
> N:=nops(x); # Число угловых точек
> T:=matrix(2,N,[x,y]); # Матрица координат
> Ris:=display(polygonplot(T,
> color="Yellow",scaling=constrained)):
> Z:=i->T[2,i]*T[1,i+1]-T[1,i]*T[2,i+1]:
> F:=add(Z(i),i=1..N-1)/2; #Площадь
  Координаты центра t.
> xc:=add(Z(i)*(T[1,i]+T[1,i+1]),i=1..N-1)/6/F;
> yc:=add(Z(i)*(T[2,i]+T[2,i+1]),i=1..N-1)/6/F;
  Осевые моменты инерции
> Jx:=add(Z(i)*(T[2,i+1]^2+
> T[2,i+1]*T[2,i]+T[2,i]^2),i=1..N-1)/12;
> Jy:=add(Z(i)*(T[1,i+1]^2+
> T[1,i+1]*T[1,i]+T[1,i]^2),i=1..N-1)/12;
> Jxy:=add(Z(i)*(T[1,i+1]*T[2,i+1]+(T[2,i+1]*T[1,i]+
> T[1,i+1]*T[2,i])/2+T[1,i]*T[2,i]),i=1..N-1)/12;
  Центральные моменты инерции
> Jxc:=Jx-yc^2*F; Jyc:=Jy-xc^2*F; Jxyc:=Jxy-yc*xc*F;
  Главные центральные моменты инерции
> Jmax:=( (Jxc+Jyc)/2+sqrt( (Jxc-Jyc)^2/4+Jxyc^2) );
> Jmin:=( (Jxc+Jyc)/2-sqrt( (Jxc-Jyc)^2/4+Jxyc^2) );
> alf:=arctan(Jxyc/(Jyc-Jmax) );
> Set(TF=F): Set(Txc=xc): Set(Tyc=yc):
> Set(TJx=Jx): Set(TJy=Jy): Set(TJxy=Jxy):
> Set(TJxc=Jxc): Set(TJyc=Jyc): Set(TJxyc=Jxyc):
> Set(TJmax=Jmax): Set(TJmin=Jmin): Set(Talf=alf):
> end use;
> Ris;
> end proc;#SolvMMin

```



```

> runMMin:=proc()
> local a1,a2,t,z,c1,c2,p1,mplt;
> z:='width'=7,editable = false;
> use Maplets:-Elements in
> t[1]:=["F=",TextField['TF'](z,'tooltip'="F")]:
> t[2]:=["xc=",TextField['Txc'](z)]:
> t[3]:=["yc=",TextField['Tyc'](z)]:
> t[4]:=["Jx=",TextField['TJx'](z)]:
> t[5]:=["Jy=",TextField['TJy'](z)]:
> t[6]:=["Jxy=",TextField['TJxy'](z)]:
> t[7]:=["Jxc=",TextField['TJxc'](z)]:
> t[8]:=["Jyc=",TextField['TJyc'](z)]:
> t[9]:=["Jxyc=",TextField['TJxyc'](z)]:
> t[10]:=["Jmax=",TextField['TJmax'](z)]:
> t[11]:=["Jmin=",TextField['TJmin'](z)]:
> t[12]:=["alpha=",TextField['Talf'](z)]:
> a1:["[x=",TextField['Tx'](30,
>   'value'=[0,2,1.9,1.4,0.9,0,5,5,7,7,5,7,2,2,0],
> 'tooltip'="Введите x, обход по часовой стрелке")]:
> a2:["[y=",TextField['Ty'](30,
>   'value'=[4,4,4.8,5.4,5.8,6,6,7,7,4,4,0,0,2,0])]:
> c1:=Button('width=90',"Run",
> Evaluate('PL1' = 'display(SolvMMin())')):
> c2:=Button("OK",Shutdown(['PL1'])):
> p1:=Plotter['PL1']('width=390'):
> mplt:=Maplet(Evaluate(function = 'SolvMMin()'),
> Window(
> 'title'="ГЕОМЕТРИЧЕСКИЕ ХАРАКТЕРИСТИКИ СЕЧЕНИЯ",
> ["ТЕОРЕТИЧЕСКАЯ МЕХАНИКА",
> [BoxColumn('spacing'=2,seq(t[i],i=1..12)),
>   [a1,a2,p1]], [c1,c2] ))
> ):#Maplet
> Maplets:-Display(mplt):
> end use;
> end proc;
> end module:
> MminModule:-runMMin():

```

Интересно отметить, что вычисления угла наклона главных осей по формуле (4.5): $\arctan(J_{xy}/(J_y - J_x))$, и по формуле (4.6): $\arctan(2*J_{xy}/(J_y - J_x))/2$ на с. 130, приводят к разным результатам. Конечно, теоретически результаты должны совпадать, однако в зависимости от точности вычислений, назначенной командой `Digits`, результаты отличаются в 4-м знаке после запятой при `Digits:=5` и в 9-м при значении `Digits:=10`, являющимся значением, принятым по умолчанию.

Для решения задачи о составном сечении из уголка и швеллера на рис. 174, с. 128, следует ввести координаты 16 угловых точек:

X:=0, 3, 6, 12, 168, 174, 177, 180, 180, 63, 63, 61, 6.1, 4, 2, 0

Y:=74, 74, 70, 5.1, 5.1, 70, 74, 74, 0, 0, -2, -4, -6.1, -61, -63, -63

Точность решения при этом получится невысокой: $F = 27,26 \text{ см}^2$ (против точного значения 27,16), $J_{\max} = 1417,9 \text{ см}^4$, $J_{\min} = 170,4 \text{ см}^4$, (точные значения 1437,84 и 170,76).

Еще одна программа 48.mws для вычисления моментов инерции сечения, состоящего из прямоугольников, треугольников и секторов круга, здесь не приведена, но содержится в архиве <http://vuz.exponenta.ru/progr.rar>. Для ввода данных в этой программе необходимо указать положение частей сечения, размеры и знак площади (вырезанная площадь отрицательная).

7.20. Напряженное состояние в точке

Здесь даны две программы для расчета компонент напряженного состояния в плоском и пространственном случае.

Основа программы 31 — решение системы уравнений равновесия, составленных вручную. Для другого варианта задачи, где иные неизвестные величины, уравнения остаются прежними, следует только подправить набор неизвестных оператора `solve`. В качестве исходных данных взяты данные задачи 55 на с. 158.

Величины `s0` и `s1` — вспомогательные, введенные для сокращения записи. Напряжения на наклонной площадке обозначены как `sigma[]` и `tau[]`, а не `sigma` и `tau`, как можно было бы предположить. Связано это с тем, что в одной программе нельзя присваивать значения одновременно переменной с некоторым именем, например `a:=1`, и переменной с тем же именем с индексом `a[2]:=3`. Результат получается всегда противоречивый и, кроме того, зависит от порядка присвоения. Можно также использовать обозначения `sigma_x`, `tau_xy`, но при этом вывод на экран не будет таким же наглядным, как при использовании греческих букв с индексом. Переменная `sigma[x]` на экране имеет естественный вид σ_x , а переменная `sigma_x` — `sigma_x`.

Умножением **A**.В матрицы поворота **A** на матрицу координат **B** получаем сразу все координаты вершин в повернутых осях (точка в Maple — знак умножения матриц).

Сортировка собственных значений (одно из которых для плоского напряженного состояния всегда нулевое) по убыванию выполняется оператором `s:=sort([sigma[max], sigma[min], 0], '>')`. Знак `>` означает направление сортировки. По умолчанию сортировка производится по возрастанию¹. Заметим, что в данной задаче не потребовалось загружать специальные пакеты ни для решения системы уравнений, ни для матричных преобразований.

Программа 31 (P31.mws)

```
> restart; # Задача 55, с. 158
> nu:=0.3:# Коэффициент Пуассона
> E:=2e3: # Модуль упругости
> AB:=10.: BC:=7.:
> x:=[2*AB/3, -AB/3, -AB/3]:#Координаты вершин
> y:=[-BC/3, -BC/3, 2*BC/3]:
> beta:=arctan(AB/BC):
> BC:=AC*cos(beta): AB:=AC*sin(beta):
> #Напряжения, МПа
> sigma[y]:=-6: tau[xy]:=-5: sigma[]:=-2:
> eqx:=-sigma[x]*BC-tau[xy]*AB+
> sigma[]*cos(beta)*AC-tau[]*sin(beta)*AC=0:
> eqy:=-sigma[y]*AB-tau[xy]*BC+
> sigma[]*sin(beta)*AC+tau[]*cos(beta)*AC=0:
> r:=solve({eqx,eqy},{tau[],sigma[x]}): assign(r):
> s0:=(sigma[x]+sigma[y])/2:
> s1:=sqrt(((sigma[x]-sigma[y])/2)^2+tau[xy]^2):
> sigma[max]:=s0+s1:
> sigma[min]:=s0-s1:
> tau[max]:=(sigma[max]-sigma[min])/2:
> tau[min]:=-tau[max]:
> alpha:=arctan((sigma[max]-sigma[x])/tau[xy]):
> alphaG:=evalf(alpha*180/Pi);# Угол в градусах
alphaG := -10.35546222
> s:=sort([sigma[max], sigma[min], 0], '>'):
> epsilon[x1]:=(sigma[max]-nu*sigma[min])/E:
> epsilon[y1]:=(sigma[min]-nu*sigma[max])/E:
```

¹Последовательности можно сортировать и по алфавиту (lexorder), по длине (length). Этот же оператор сортирует запись полиномов по степеням.

```

> epsilon[z]:=-nu*(sigma[min]+sigma[max])/E:
> УдИзмОбъема:=epsilon[x1]+epsilon[y1]+epsilon[z]:
> # Матрица поворота
> A:=Matrix([[cos(alpha), sin(alpha)],
>             [-sin(alpha), cos(alpha)]]):
> # Матрица координат вершин
> B:=Matrix([x, y]):
> Z:=A.B:# Координаты вершин в повернутых осях
> ДеформацииX=[seq(Z[1, k], k=1..3)]*epsilon[x1];
> ДеформацииY=[seq(Z[2, k], k=1..3)]*epsilon[y1];
ДеформацииX = [0.08176493987, -0.03350996750, -0.04825497235]
ДеформацииY = [0.007307116553, 0.01928096580, -0.02658808236]

```

Решим задачу 56 на с. 162 о трехмерном напряженном состоянии в точке. Для заполнения симметричной матрицы применяем режим записи в верхнюю треугольную часть симметричной матрицы — `shape=symmetric`. Это заметно ускоряет ввод данных.

Формулы (4.35), с. 164, для вычисления главных касательных напряжений τ_k , $k = 1, 2, 3$ внешне напоминают векторное произведение, поэтому оператор `CrossProduct` использован формально, только для сокращения записи. Операция `Tau[2]:=-Tau[2]` при этом подправляет знак у касательного напряжения τ_2 .

Длина вектора вычисляется с помощью оператора `Norm(Tau, 2)`. Опция 2 в операторе указывает на евклидову норму вектора. Оператор `sort` сортирует в порядке убывания (опция `>`) собственные значения. Вычисления деформаций по формулам (4.34), с. 164, построены с учетом очевидной круговой перестановки индексов в формулах. В пакете `ListTools` для этого есть специальный оператор `Rotate`.

Программа 32 (P32.mws)

```

> restart;# Задача 56, с. 162
> with(LinearAlgebra):
  Коэффициент Пуассона и модуль Юнга (МПа)
> nu:=0.4: E:=1e5:
  Заданные напряжения sigma11,tau12,tau13,sigma22,tau23,sigma33
> S:=Matrix([[11., 12, 14], [13, 1], [-33]]),
> shape=symmetric, scan=triangular[upper]);

```

$$S := \begin{bmatrix} 11. & 12 & 14 \\ 12 & 13 & 1 \\ 14 & 1 & -33 \end{bmatrix}$$



Главные напряжения

> S0:=Eigenvalues(S);

$$S0 := \begin{bmatrix} -37.1974763823047852 \\ 2.27082811323587076 \\ 25.9266482690689272 \end{bmatrix}$$

> E1:=Vector(3,[1,1,1]);# Вспомогательный вектор

$$E1 := \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

Главные касательные напряжения

> Tau:=CrossProduct(E1,S0)/2: Tau[2]:=-Tau[2]: Tau;

$$\begin{bmatrix} 11.8279100800000006 \\ 31.5620623249999994 \\ 19.7341522450000008 \end{bmatrix}$$

> 2/3*Norm(Tau,2);#Октаэдрическое напряжение

$$26.03843313$$

> s:=sort([S0[1],S0[2],S0[3]], '>');

> with(ListTools):

> for i to 3 do

> eps[i]:=(s[1]-(s[2]+s[3])*nu)/E:

> s:=Rotate(s,1);# Круговая перестановка

> od:

> evalf(seq(eps[i],i=1..3),4);

> ОтноситИзмОбъема=add(eps[i],i=1..3);

$$0.0003990, 0.00006779, -0.0004848$$

$$\text{ОтноситИзмОбъема} = -0.0000179984000$$

Собственные числа оператор `Eigenvalues` дает с излишней точностью: 16 знаков после запятой. При этом переменная `Digits` на это никак не влияет. Напомним, что переменная `Digits` управляет точностью вычислений. Задавая значение `Digits`, можно менять число значащих цифр в расчете. Ошибочное мнение, что эта переменная выполняет роль форматирования чисел при выводе на печать, приводит к неправильным результатам. В некоторых программах, в которых решаются системы линейных уравнений большого порядка, это весьма существенно. Например, если задать `Digits:=2` в программе 6 расчета фермы, с. 345, то вместо $S_6 = 8,485$ можно получить $S_6 = 8,6$. Очевидна высокая погрешность таких вычислений. По умолчанию `Digits:=10`. Оператор перевода в число с плавающей запятой `evalf(z,4)` с необязательной опцией, указывающей число значащих

цифр (в данном случае 4), легко справляется с проблемой разумной формы вывода результатов. Здесь это применено для вывода значений деформаций. Кроме того, можно обратиться к весьма богатому набору пользовательских переменных для вывода информации `interface`. Если записать `interface(displayprecision=3)`, то число знаков при выводе вещественных чисел будет равно 3.

7.21. Статически неопределимые системы

Решим задачу 59 на с. 170.

Вектора неизвестных усилий заданы направляющими единичными векторами, полученными по координатам начала (на балке) и конца (на шарнирной опоре) с помощью оператора `Normalize` с опцией `Euclidean` пакета линейной алгебры `LinearAlgebra`. Вектор нагрузки имеет номер 4. Параметр `n:= [1, 2, 4]` указывает номера векторов в уравнении равновесия. Для того чтобы задействовать операторы этого пакета, в том числе оператор векторного произведения `CrossProduct`, пришлось всем векторам данной плоской задачи дать фиктивную нулевую третью координату.

В переменной `EQ` записаны все три компонента результата векторного произведения. Уравнение моментов представляет собой проекцию этого произведения на ось z , т. е. третью компоненту `EQ[3]`, которая и используется для определения `s[3]` оператором `solve(EQ[3], s[3])`.

Для рисунка задачи потребовался оператор изображения опоры `Опора(i,R)` пакета `Podpr1.m`, с. 356, загружаемого в начале программы оператором `read`. В названиях переменных и операторов Maple допускаются русские буквы. Для удобства работы рекомендуем использовать абсолютный адрес файла.

Оператор `Опора(i,R)` помещает опору в точке номер i , координаты которой заданы списками `x[i]` и `y[i]`. Отметим, что переменные `x[i]` и `y[i]` являются глобальными, поэтому не следует менять имена списков с координатами опор. В операторе `Опора(i,R)` используются именно `x` и `y`. Параметр `R` задает масштаб изображения.

Программу можно улучшить. Так, выбор толщины стрелки `0.5` в опции `arw:=0.5,2,0.2,color=red` и масштаб `1` в операторе `Опора(i,1)` для упрощения программы здесь выполнен вручную. Читателю предоставляем возможность по собственному вкусу добавить операцию автоматического масштабирования, связанного, очевидно, с масштабом координат шарниров.

Программа 33 (P33.mws)

```
> restart;# Задача 59, с. 170. Статич.неопр.система
> read "C:\\Podpr1.m":# Программа 9, с. 356
```

```

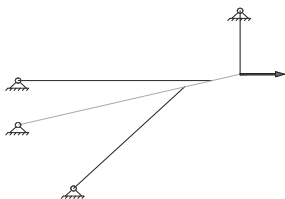
> n:=[1,2,4]:
> with(LinearAlgebra):
  Координаты шарнира балки
> x0:=0: y0:=0:
  Координаты шарниров на балке
> xb:=[80,70,60]: yb:=[20,17.5,15]:
  Координаты опорных шарниров
> x:=[80,0,20,x0]: y:=[45,17.5,-25,y0]:
  Длины стержней
> for i to 3 do
>   v[i]:=Vector([x[i]-xb[i],y[i]-yb[i],0]):
>   L[i]:=VectorNorm(v[i],2);
>   v1[i]:=s[i]*Normalize(v[i],Euclidean);
>   r[i]:=Vector([xb[i]-x0,yb[i]-y0,0]):
> od:
  Координаты точки приложения силы
> r[4]:=Vector([80,20,0]):
  Сила
> v1[4]:=Vector([16,0,0]):
  Метод сил
> N:=Matrix(3):
> for j to 3 do
>   i:=n[j]: s[i]:=1:
>   EQ:=CrossProduct(r[3],v1[3])+
>       CrossProduct(r[i],v1[i]):
>   N[j,j]:=1: N[j,3]:=evalf(solve(EQ[3],s[3]));
> od:
  Интеграл Максвелла–Мора
> for i to 2 do
>   for j to 3 do
>     d[i,j]:=evalf(add(L[k]*N[i,k]*N[j,k],k=1..3));
>   od;
> od;
  Каноническая система метода сил
> for i to 2 do
>   eq[i]:=d[i,1]*x1+d[i,2]*x2+d[i,3];
> od:
> s1:=solve({eq[1],eq[2]},{x1,x2}):assign(s1);
> s[1]:=x1; s[2]:=x2;

```

```

s1 := 3.679831111
s2 := 0.2874868064
> EQ:=add(CrossProduct(r[k],v1[k]),k=1..4):#Ур.мом.
> s[3]:=evalf(solve(EQ[3],s[3]));
s3 := -0.6468453185
> with(plots): with(plottools):# Рисунок
> for i to 3 do # Три стержня
> R[i]:=PLOT(CURVES([[xb[i],yb[i]],[x[i],y[i]]])):
> od:
Параметры стрелки
(толщина, толщина наконечника, отн. длина наконечника)
> arw:=0.5,2,0.2,color=red:
> Nagr:=arrow( [r[4][1],r[4][2]],v1[4],arw):#Нагрузка
Балка
> balka:=PLOT(CURVES([ [x0,y0],[r[4][1],r[4][2]]],
> COLOR(RGB,0,1,0)),THICKNESS(3)):
> display(seq(Опора(i,1),i=1..4),Nagr,balka,
> seq(R[i],i=1..3),scaling=constrained,axes=none);

```



Следующая программа решает задачу 60 на с. 177 о статически неопределимой системе кинематическим методом. В начале программы вводятся исходные данные: число стержней n , координаты характерных точек и нагрузки. Искомые усилия составляют список $s[i]$, $i = 1, \dots, n$. Орты усилий в стержнях $s1[i]$ вычисляются по координатам концов стержней. Составляется система трех уравнений равновесия пластины. С помощью оператора скалярного произведения `DotProduct` записываются уравнения проекций, а с помощью оператора векторного произведения `CrossProduct` — уравнения моментов. Моментная точка — начало координат. При необходимости (для проверки) эту точку можно перенести в другое место, добавив к радиус-векторам точек приложения сил $r[i]$ радиус-вектор новой точки.

Оператор `assume` определяет свойства переменных. В данном случае неизвестные компоненты скорости полюса v_x , v_y и угловая скорость w вещественные. Если не назначать свойства этим переменным,

то окончательный результат не изменится, однако уравнение совместности деформаций, связывающее Δl_i , $i = 1, 2, 3, 4$, будет иметь сложный вид.

Программа позволяет рассчитывать и более сложные системы. Если число деформируемых опорных стержней больше четырех, то надо увеличить число n и длины списков \mathbf{x}_b , \mathbf{y}_b , \mathbf{x} и \mathbf{y} . При этом после исключения кинематических параметров \mathbf{v}_x , \mathbf{v}_y и \mathbf{w} число уравнений совместности деформаций получится больше: $n - 3$. Число нагрузок можно увеличить, не меняя существенно программу. Для этого векторам нагрузок надо дать номера, большие $n + 1$.

Исключение кинематических параметров здесь производится с помощью оператора решения `solve`. На с. 178 для этих же целей используется оператор `eliminate`.

Так как из всех решений \mathbf{z} системы `Eq[1], Eq[2], Eq[3], Eq[4]`, связывающей удлинения стержней с введенными кинематическими параметрами \mathbf{v}_x , \mathbf{v}_y и \mathbf{w} , нас интересует только Δl_1 , то для передачи решений, полученных в результате работы `solve`, соответствующим переменным вместо оператора `assign`, который назначает решения всем переменным, здесь достаточно применить оператор `subs(z, DL[1])`.

Другой способ исключения переменных из системы уравнений описан на с. 178.

Программа 34 (P34.mws)

```
> restart; # Задача 60, с. 177. Кинематический метод
> read "C:\\Podpr1.m": # Программа 9, с. 356
> n:=4: # Число опорных стержней
> with(LinearAlgebra):
  Координаты шарниров на пластине
> xb:=<1, 3, 3, 1>: yb:=<3, 3, 1, 1>:
  Координаты опорных шарниров
> x:=<0, 4, 2, 1>: y:=<2, 2, 0, 0>:
  Длины и орты стержней
> for i to n do
>   S[i]:=Vector([x[i]-xb[i], y[i]-yb[i], 0]):
>   L[i]:=VectorNorm(S[i], 2);
>   sl[i]:=s[i]*Normalize(S[i], Euclidean);
>   r[i]:=Vector([xb[i], yb[i], 0]):
> od:
  Координаты точек приложения сил
> h:=1: # плечо пары (для рисунка)
> r[n+1]:=Vector([2, 2, 0]):
```

```

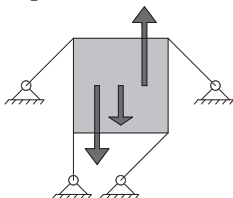
> r[n+2]:=Vector([2+h/2,2,0]):
> r[n+3]:=Vector([2-h/2,2,0]):
Нагрузки
> m0:=10:# Момент
> s1[n+1]:=Vector([0,-5,0]):
> s1[n+2]:=Vector([0,m0/h,0]):
> s1[n+3]:=Vector([0,-m0/h,0]):
> ###
> Моменты:=add(CrossProduct(r[i],s1[i]),i=1..n+1):
> ОптX:=Vector([1,0,0]): ОптY:=Vector([0,1,0]):
> eq[1]:=add(DotProduct(ОптX,s1[i]),i=1..n+1):
> eq[2]:=add(DotProduct(ОптY,s1[i]),i=1..n+1):
> eq[3]:=Моменты[3]+m0:
> assume(vx,real,vy,real,w,real);
> V[1]:=Vector([vx,vy,0]):
> omega:=Vector([0,0,w]): # Угловая скорость
> for i from 2 to n do
>   V[i]:=V[1]+CrossProduct(omega,r[i]-r[1]):
> od:
> for i to n do
>   Eq[i]:=DL[i]-DotProduct(V[i],s1[i])/s[i];
> od:
> z:=solve({seq(Eq[k],k=1..4)},{vx,vy,w,DL[1]}):
> eq[n]:=DL[1]-subs(z,DL[1]):
   Закон Гука (4.54), с. 179.
> for i to n do DL[i]:=s[i]*L[i]/(E*F); od:
> evalf(solve({seq(eq[i],i=1..n)},
>
>               {seq(s[i],i=1..n)}));
{s1 = -2.2147762, s2 = -0.2980062, s3 = 1.9167700, s4 = -4.5785556}
   Рисунок
> with(plots): with(plottools):
   Стержни
> for i to n do
>   R[i]:=PLOT(CURVES([[xb[i],yb[i]],[x[i],y[i]]]])):
> od:
   Параметры стрелки
   (толщина, толщина наконечника, относит. длина наконечника)
> Параметры:=0.1,0.5,0.2,color=red:

```

```

> Нагрузка:=PLOT(seq(arrow([r[n+i][1],r[n+i][2]],
>                               s1[n+i]/6,Параметры),i=1..3)):
> Пластина:=PLOT(POLYGONS([seq([xb[j],yb[j]],
>                               j=1..4]),COLOR(HUE,0.5))):
> display(seq(Опора(i,0.1),i=1..n),Нагрузка,Пластина,
>           seq(R[i],i=1..n),scaling=constrained,axes=none);

```



7.22. Кручение

7.22.1. Кручение неоднородного вала

Следующая простая программа имеет расчетный характер. Все вычисления производятся по готовым формулам. Число участков вала определяется параметром n . Числовые данные о вале и нагрузках вводятся в виде списков (в Maple — тип `exprseq`). Рассмотрена задача 49 на с. 135. При расчете конкретного вала в программе следует уделить внимание только вычислению моментов инерции. Так, моменты инерции кругового сечения вычислены в цикле, а момент инерции кольца — в виде отдельного оператора. Далее все расчеты производятся автоматически. При вычислении касательных напряжений моменты переводим в Нм (коэффициент 10^3), а результат в МПа (делим на 10^6). Программу можно дополнить несложной процедурой построения эпюр. Результаты счета для сокращения записи здесь не видны, хотя в действительности они выводятся на экран, так как после всех необходимых операторов стоит точка с запятой.

Программа 35 (P35.mws)

```

> restart;           # Кручение
> n:=3:             # Число участков
> d:=0.09,0.14,0.08,0.07: # Диаметры, м
> G:=8*10^10:      # Модуль сдвига, Па или Н/м2
> M0:=30,-10,20:  # Моменты, кНм
> l:=1.1,1.2,0.8: # Длины, м
> for i to 2 do    # Моменты инерции, м4

```

```

> J[i]:=evalf(Pi*d[i]^4/32);
> od;
> J[3]:=evalf(Pi*(d[3]^4-d[4]^4)/32);
> for i to n do # Моменты в сечениях
> M[i]:=add(M0[k],k=i..n);
> od;
> for i to n do # Углы относительные
> ft[i]:=M[i]*1000*l[i]/G/J[i];
> od;
> for i to n do # Углы абсолютные
> f[i]:=add(ft[k],k=1..i);
> od;
> for i to n do
> W[i]:=J[i]/(d[i]/2); #Моменты сопротивления
> tau[i]:=M[i]*1e3*d[i]/J[i]/2/1e6; #МПа
> od;

```

7.22.2. Статически неопределимая задача о кручении

Вал, защемленный по концам, представляет собой один раз статически неопределимую систему.

В программе использованы данные задачи 50 на с. 137.

Следует отметить конструкцию `seq(op([A1,A2]),i=0..n-1)`, где A1 и A2 есть пары чисел типа $[x[i+1], \tau[i+1]]$. Оператор `seq` не может использовать в качестве аргумента перечисление A1,A2, поэтому эту пару необходимо взять в скобки. Но эти скобки в операторе `plot` приводят к тому, что вместо одного графика строится n графиков. Для непрерывных функций это допустимо, а для разрывной, как в этом случае, результат получается неудовлетворительным. Поэтому, чтобы снять скобки, применен оператор `op`.

Программа 36 (P36.mws)

```

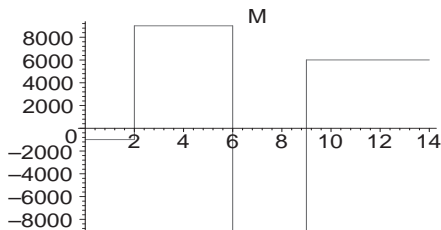
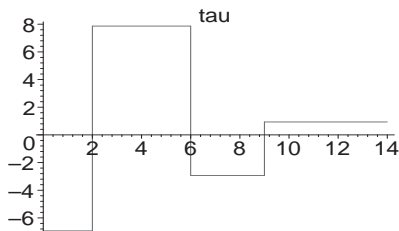
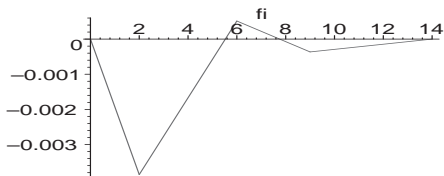
> restart; n:=4: # Число участков
> m:=0,10,-18,15:# Моменты (кН)
> L:=2,4,3,5: # Длины участков(м)
> dm:=0.09,0.18,0.25,0.32:# Диаметры (м)
> G:=80e9: # Модуль сдвига(Па)
> M[0]:=-Ma: f[0]:=0: x[0]:=0:
> for i to n do
> Ji:=Pi*dm[i]^4/32:

```

```

> Wi:=2*Ji/dm[i]:
> M[i]:=M[i-1]+m[i]*1e3:
> ffi:=M[i]*L[i]/G/Ji;
> f[i]:=f[i-1]+ffi:
> x[i]:=x[i-1]+L[i]:
> tau[i]:=M[i]/Wi/1e6;# МПа
> od:
> Ma:=solve(f[n],Ma);# Момент на опоре (Нм)
      Ma := 993.2303550
> plot([seq([x[i],f[i]],i=0..n)],title='fi');
> plot([seq(op([[x[i],tau[i+1]], [x[i+1],tau[i+1]]]),
      i=0..n-1)],title='tau');
> plot([seq(op([[x[i],M[i+1]], [x[i+1],M[i+1]]]),
      i=0..n-1)],title='M');

```



```

> evalf(seq(tau[i],i=1..n));# Касательные напряжения
    -6.938930946, 7.865415144, -2.931337349, 0.9336006649

> evalf(seq(f[i]*1000,i=1..n-1));# Углы*1000
    -3.854961636, 0.5147134425, -0.3646877606

> evalf(seq(M[i],i=1..n));# Моменты (Нм)
    -993.2303550, 9006.769645, -8993.230355, 6006.769645

```

7.22.3. Кручение вала прямоугольного сечения

В задаче кручения прямоугольного сечения максимальное значение касательного напряжения τ_{\max} на длинной стороне прямоугольника ($b \geq a$) вычисляется через крутящий момент M_k и момент сопротивления W_k по формуле

$$\tau_{\max} = M_k/W_k = \tau_B, \quad (7.3)$$

где $W_k = \alpha ba^2$. Коэффициент α зависит от соотношения сторон прямоугольника. Напряжение на середине короткой стороны (рис. 422) сечения вычисляется по формуле

$$\tau_A = \gamma \tau_{\max} = \gamma \tau_B. \quad (7.4)$$

Относительный угол закручивания θ можно вычислить, если известен крутящий момент M_k , модуль упругости при сдвиге G и геометрический фактор жесткости J_k

$$\theta = M_k/(GJ_k). \quad (7.5)$$

Для круглого сечения диаметром d момент инерции при кручении¹ J_k совпадает с полярным моментом инерции $\pi d^4/32$. Для эллипса (полуоси a, b) имеем $J_k = \pi(ab)^3/(a^2 + b^2)$, для прямоугольника вводится поправочный коэффициент

$$J_k = \beta ba^3. \quad (7.6)$$

В следующей программе можно вычислить значения касательных напряжений при кручении вала прямоугольного сечения и поправочные коэффициенты α, β, γ (коэффициенты Сен-Венана). Используется

¹Фактор жесткости.

известная функция напряжений Прандтля¹ из решения С. П. Тимошенко², удовлетворяющая уравнению Пуассона

$$\frac{\partial^2 \Phi}{\partial x^2} + \frac{\partial^2 \Phi}{\partial y^2} = -2. \quad (7.7)$$

Начало координат помещается в центре прямоугольника. Выражения для касательных напряжений в любой точке сечения вала имеют вид

$$\tau_{zy} = -G\theta \frac{\partial \Phi}{\partial x}, \quad \tau_{zx} = G\theta \frac{\partial \Phi}{\partial y}. \quad (7.8)$$

Крутящий момент пропорционален погонному углу закручивания θ :

$$M = 2G\theta \iint \Phi dx dy.$$

Известно³ решение уравнения (7.7):

$$\Phi = \frac{8a^2}{\pi^3} \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{k^3} \left(1 - \frac{\operatorname{ch}(ky\pi/a)}{\operatorname{ch}(kb\pi/(2a))} \cos(kx\pi/a) \right),$$

где $k = 2n - 1$. Время счета существенно зависит от числа удерживаемых членов ряда N . Оценивает время счета оператор `time`. В начале программы этот оператор заносит значение текущего времени в переменную `t1`, затем, в конце счета, возвращает разность текущего времени и `t1`, т. е. время счета в секундах. Для ускорения вычисления двойного интеграла на функцию F действуем оператором `evalf`, переводящим интегрирование в численное. Иначе по умолчанию **Maple** вычисляет интегралы аналитически, делая это для

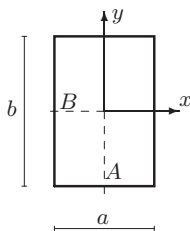


Рис. 422

каждого члена ряда, что приводит к катастрофическому росту времени счета⁴. Результат вычисления коэффициентов α , β , γ при $N = 8$ и соотношении сторон закручиваемого прямоугольника $b/a = 4$ показывает, что точнее всего получается значение β (три правильных значащих цифры). Наиболее чувствительным к числу членов ряда

¹Ludwig Prandtl (1875–1953) — немецкий физик.

²Тимошенко Степан Прокофьевич (1878–1972) — русский ученый, один из основоположников механики сплошных сред в США.

³Тимошенко С. П., Гудьер Дж. Теория упругости — М.: Наука, 1975.

⁴Время вычисления какого-либо одного оператора можно вычислить, заключив его в аргумент оператора `time`. Например, время счета определенного интеграла `time(int(x/sin(x), x=1..6/5))` составит около шести секунд.

является коэффициент γ . Для получения трех правильных цифр в γ требуется более 300 членов ряда.

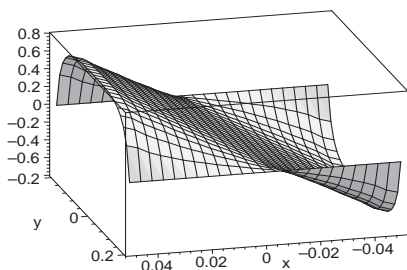
Заметим, что коэффициент γ обозначен как `gamma1`, так как `гамма` в **Maple** используется как константа Эйлера.

Программа 37 (P37.mws)

```
> restart; N:=8: k:=2*n-1:#Вычисл. поправочных коэфф.
> t1:=time():# Засекаем время счета
  Функция напряжений (решение Тимошенко С. П.)
> F:=evalf(8*a^2/Pi^3*sum(1/k^3*(-1)^(n-1)*(1-
> cosh(k*y*Pi/a)/cosh(k*b*Pi/2/a))*
>
>                                     cos(k*x*Pi/a),n=1..N)):
> b:=4*a:# Соотношение сторон
> tau_zy:= -G*teta*diff(F,x):#Касательные напряжения
> tau_zx:=G*teta*diff(F,y):
> tau_max:=subs(y=0,x=a/2,tau_zy): #tau B
> tau_min:=subs(y=-b/2,x=0,tau_zx): #tau A
> gamma1:=tau_min/tau_max; #tau A/tau B
>                                      $\gamma_1 := 0.7624800802$ 
  Крутящий момент
> M:=2*G*teta*int(int(F,x=-a/2..a/2),y=-b/2..b/2):
> beta:=M/(G*teta*a^3*b);
>                                      $\beta := 0.2807997925$ 
> alpha:=M/(tau_max*a^2*b);
>                                      $\alpha := 0.2889852618$ 
> time()-t1;# Время счета
>                                     6.891
```

Вычисления можно дополнить трехмерной иллюстрацией распределения касательных напряжений по сечению вала. В **Maple** для этого есть удобная процедура изображения трехмерных графиков `plot3d`. Например, распределение τ_{zy} (в ГПа) по прямоугольному сечению $a \times b$ получаем следующим образом:

```
> a:=0.1: teta:=0.1:
> G:=80: # Модуль сдвига(ГПа)=80*10^9 Па
> plot3d(tau_zy,x=-a/2..a/2,y=-b/2..b/2,
>
>                                     axes=BOX,lightmodel=light2,
>                                     orientation=[75,60],shading=NONE);
```

Опции цветового решения `lightmodel` (выбрана вторая модель из четырех), ориентации графика `orientation` (углы заданы в градусах) и освещения `shading` не являются обязательными. По умолчанию они, как правило, задаются приемлемыми.

Результаты вычислений по программе 37 для различных соотношений b/a сторон прямоугольника занесем в таблицу:

Коэффициенты Сен-Венана

Таблица 6

b/a	1	1,5	2	2,5	3	4	8	10	999
α	0,208	0,231	0,246	0,258	0,267	0,282	0,307	0,312	0,333
β	0,141	0,196	0,229	0,249	0,263	0,281	0,307	0,312	0,333
γ	1,000	0,859	0,795	0,766	0,753	0,745	0,742	0,742	0,742

Очевидно, для длинных узких прямоугольников с большой степенью точности можно считать $\alpha = \beta = 1/3$.

7.23. Кручение стержня открытого профиля

7.23.1. Секториальные характеристики сечения. *Маллет*

Программа вычисления секториального момента инерции сечения на примере решения задачи 52 оформлена в виде маплета. Модуль `JOmegaModule` состоит из двух процедур: `JOmг` и `runJOmг`. В первой производится вычисления, во второй организован интерфейс программы (рис. 423). Программа запускается вызовом процедуры `runJOmг`.

В процедуре `JOmг` оператором `Get` вводим данные о сечении: толщину h , координаты угловых точек (все в сантиметрах) и порядок обхода сечения. В программе предусмотрен выбор полюса. Этот ввод вполне можно опустить, так как от выбора полюса окончательный ответ не зависит. Однако в учебных целях полезно посмотреть, как зависят (или не зависят) расчетные значения от полюса. По умолчанию полюс

находится в начале координат. Это отражено в опции 'value'='0' в процедуре runJong при описании интерфейса программы.

Особое внимание следует уделить вводу порядка обхода точек, определяемого двойным списком n . Сразу отметим, что список обязательно должен быть двойным, даже в тех случаях, когда обход состоит из единственной ветви, как, например, в описанном примере. Двойной список открывается и закрывается двумя скобками $[[]]$. Обход начинается из точки отсчета секториальных координат. Номер этой точки заносится в переменную $n0$. Если сравнивать сечение

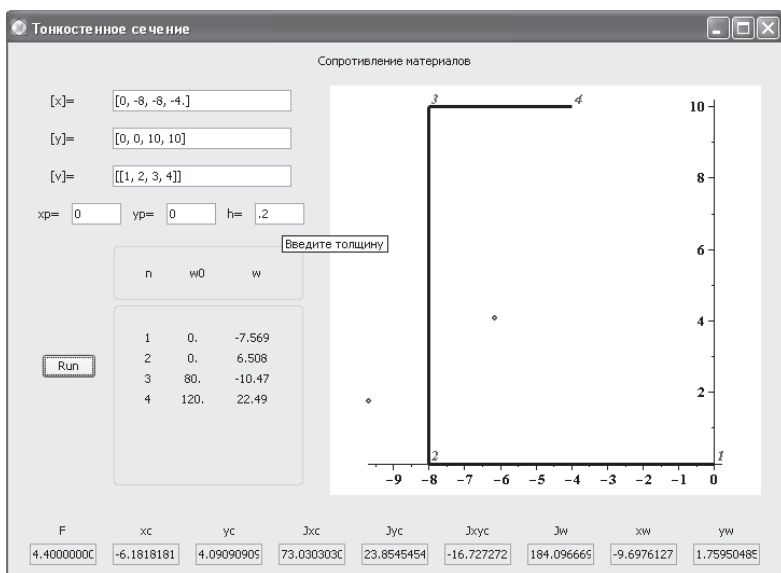


Рис. 423. P38.maplet. Решение задачи 52

открытого профиля с деревом теории графов [30], то $n0$ — корень дерева, а элементы списка n — ветви дерева. При вводе есть ограничения:

1. Каждый элемент списка n является списком, начинающимся с уже пройденной точки или с $n0$.
2. Каждый участок проходим по одному разу.
3. Проходим все участки.

Рассмотрим ввод данных, например, в задаче 51 на с. 143. Пронумеруем угловые точки (рис. 424):



Рис. 424

Выбирая точку 1 за начало отсчета, получим следующую последовательность: $[[1, 2, 3, 4], [2, 5, 6], [3, 7, 8]]$. Возможен также следующий вариант: $[[8, 7, 3, 2, 5, 6], [2, 1], [3, 4]]$. В первом случае началом отсчета будет точка 1, во втором — 8. Это обеспечивается присвоением значения переменной $n0:=n[1][1]$.

В программе можно легко предусмотреть расчет сечений из элементов разной толщины. Здесь толщина принята одинаковой и вводится в переменную h в поле **Th**.

Построения эпюры секториальных координат оформлено в процедуру **W**, аргументом которой являются координаты полюса. Процедура в программе вызывается дважды. Сначала строится некоторая эпюра секториальных координат при произвольном выборе полюса, затем, после нахождения центра кручения, полюс помещается в этот центр и строится эпюра главных секториальных координат. Интегрирование по Верещагину занесено в процедуру **Intgr**, аргументами которой являются списки значений ординат перемножаемых эпюр. Как оператор процедура **Intgr** коммутативна. Если требуется вычислить интеграл не от произведения, а от одной функции, как, например, при вычислении площади, то в качестве второго аргумента берем список из единиц $v1:=\text{Vector}(1..N, 1)$, т. е. умножаем функцию по правилу Верещагина на эпюру, состоящую из прямоугольников единичной высоты.

Для рисования профиля использован пользовательский оператор **Линия**. В результате счета на экран помимо самого сечения выводятся две точки: центр тяжести и центр кручения.

Заметим, что результаты расчетов существенно зависят от параметра точности счета **Digits**, который лучше не уменьшать от стандарта 10, заданного по умолчанию¹.

Для сокращения текста в программе нет процедуры построения эпюр, вместо этого в два столбца выводятся списки их ординат. Первый столбец, обозначенный $w0$, соответствует эпюре секториальных координат с полюсом в точке (xp, yp) , второй столбец w — эпюра главных секториальных координат (полюс в центре изгиба (xw, yw)). Все эти результаты оформлены в виде подписей **caption** списка меток **Label**.

¹Параметр точности можно задать так: $\text{Digits}:=12$, а для проверки его значения достаточно просто вызвать переменную **Digits**. Для того чтобы узнать предельно допустимую точность в используемой версии **Maple**, следует вызвать функцию $\text{kernelopts}(\text{maxdigits})$.

Кроме того, опущены почти все "всплывающие" подсказки tooltip. В архиве <http://vuz.exponenta.ru/progr.rar> дана полная, более подробная версия программа со всеми подсказками и программа 51.mws для решения задачи 51, с. 143, где участки имеют разные толщины.

Программа 38 (P38.mws)

```
> restart; with(plots): with(plottools):# Задача 52.
> JOmegaModule:=module()
> export JOmg,runJOmg:
> JOmg:=proc() local N,n,alpha_x,alpha_y,x,y,ЛинияЯ,
> xP,yP,xP0,yP0,Sx,Sy,Swx,Swy,F,Jx,Jy,Jxy,Jxc,Jyc,
> Jxyc,Jw,xc,yc,Intgr,W,v1,D1,Центр,n0,h,Шрифт,i,
> Сечение,Точка,Ris:
> global k,w:
> ЛинияЯ:=proc(i,j)
>     CURVES([[x[i],y[i]],[x[j],y[j]]],THICKNESS(3)):
>     end:
> use Maplets[Tools] in
> x:=Get('Tx'::list); y:=Get('Ty'::list);# Координаты
> h:=Get('Th::float');# Толщина
> N:=nops(x):      # Число точек
> n:=Get('Tn'::list); # Ветви
> xP0:=Get('Tpx'::integer); # x полюса
> yP0:=Get('Tpy'::integer); # y полюса
> k:=nops(n):      # Число ветвей
> n0:=n[1][1]:     # Начало отсчета
> W:=proc(x0,y0) local i,j,p,q:
> for j to k do w[n0]:=0:
>   for i to nops(n[j])-1 do
>     p:=n[j,i]:     q:=n[j,i+1]:
>     w[q]:=- (x[p]-x0)*(y[q]-y0)+
>                                     (x[q]-x0)*(y[p]-y0)+w[p];
>   od:
> od: end proc:
> Шрифт:=FONT(TIMES,BOLD,10):
> for i to N do
>   Точка[i]:=PLOT(ТЕХТ([x[i]+0.2,y[i]+0.2],
```

```

> convert(i, symbol)), Шрифт, COLOR(HUE, 0.7)):
> od:
> Сечение:=seq(seq(Линия(n[j], i), n[j, i+1]),
> i=1..nops(n[j])-1), j=1..k),
> seq(Точка[i], i=1..N):
> W(xP0, yP0): # Получаем эпюру w
> for i to N do
> Set(TLw0[i](caption)=
> convert(evalf(w[i], 4), string)):
> od:
> v1:=Vector(1..N, 1):
  Интегрируем по правилу Верещагина
> Intgr:=proc(A, B) local dst, s, j, i, p, q: global k:
> s:=0:
> for j to k do
> for i to nops(n[j])-1 do
> p:=n[j, i]: q:=n[j, i+1]:
> dst:=h*sqrt((x[p]-x[q])^2+(y[p]-y[q])^2):
> s:=s+dst/6*(A[q]*(2*B[q]+B[p])+A[p]*(B[q]+2*B[p])):
> od;
> od; return(s); end:
> F:=Intgr(v1, v1): # Площадь
> Jx:=Intgr(y, y): # Моменты инерции
> Jy:=Intgr(x, x): Jxy:=Intgr(x, y):
> Sx:=Intgr(v1, y): Sy:=Intgr(v1, x): # Статич. моменты
> xc:=Sy/F: yc:=Sx/F: # Центр тяжести
> Jxc:=Jx-F*yc^2: Jyc:=Jy-F*xc^2: Jxyc:=Jxy-F*xc*yc:
> x:=evalm(x-xc): y:=evalm(y-yc):
> Swy:=Intgr(w, x): Swx:=Intgr(w, y):
> alpha_x:=(-Swx*Jyc+Swy*Jxyc)/(Jxc*Jyc-Jxyc^2):
> alpha_y:=(Swy*Jxc-Swx*Jxyc)/(Jxc*Jyc-Jxyc^2):
> xP:=alpha_x+xP0-xc: yP:=alpha_y+yP0-yc:
> W(xP, yP): # Получаем эпюру w с полюсом P(xP, yP)
> D1:=Intgr(w, v1)/F:
> w:=map(x->x-D1, w): #Эпюра w главных сект. коорд.
> Jw:=Intgr(w, w):
> Центр:=PLOT(POINTS([xP+xc, yP+yc], [xc, yc])):
> Ris:=display(Центр, Сечение, scaling=CONSTRAINED):

```

```

> Set(TF=F):      Set(Txc=xc):      Set(Tyc=yc):
> Set(TJxc=Jxc): Set(TJyc=Jyc): Set(TJxyc=Jxyc):
> Set(TJw=Jw):   Set(Txw=xP+xc):Set(Tyw=yP+yc):
> for i to N do
>   Set(TL[i](caption)=convert(i, string)):
>   Set(TLw[i](caption)=convert(evalf(w[i], 4), string)):
> od:
> end use:
> Ris:
> end proc:

> runJOMg:=proc() local xp0, yp0, a, k, BRun, p1, mplt,
>                   z, z0, z1, z2, H0, Sn, Sw, Sw0, Rez;
> z0:=editable = false: z:='width'=7, z0:
> z1:='width'=1, 'height'=15', z0:
> z2:='width'=5, 'height'=15', z0:
> use Maplets:-Elements in
> a[1]:=["x=", TextField['Tx']('value'=[0, -8, -8, -4])]:
> a[2]:=["y=", TextField['Ty']('value'=[0, 0, 10, 10])]:
> a[3]:=["v=", TextField['Tn']('value'=[[1, 2, 3, 4]])]:
> xp0:="xp=", TextField['Tpx'](5, 'value'='0'):
> yp0:="yp=", TextField['Tpy'](5, 'value'='0'):
> H0:="h=", TextField['Th'](5, 'value'='0.2'):
> k[1]:=["F", TextField['TF'](z, 'tooltip'="Площадь")]:
> k[2]:=["xc", TextField['Txc'](z)]:
> k[3]:=["yc", TextField['Tyc'](z)]:
> k[4]:=["Jxc", TextField['TJxc'](z)]:
> k[5]:=["Jyc", TextField['TJyc'](z)]:
> k[6]:=["Jxyc", TextField['TJxyc'](z)]:
> k[7]:=["Jw", TextField['TJw'](z)]:
> k[8]:=["xw", TextField['Txw'](z, 'tooltip'="Ц.изг.")]:
> k[9]:=["yw", TextField['Tyw'](z)]:
> BRun:=Button("Run", Evaluate('P'='display(JOMg())')):
> p1:=Plotter['P']('height'=380):
> Sn:=seq(Label[TL[i]]( ), i=1..14):
> Sw0:=seq(Label[TLw0[i]]( ), i=1..14):
> Sw:=seq(Label[TLw[i]]( ), i=1..14):
> Rez:=seq(t[i], i=1..9):

```

```

> mplt:= Maplet(Evaluate(function='JOMg()'),
> Window('title'="Тонкостенное сечение",
>         ["Сопротивление материалов",
>         [BoxColumn(seq(a[i], i=1..3), [xp0, yp0, H0], [BRun,
>         [BoxColumn(border, ["n", " w0  ", " w  "]),
>         BoxColumn(border, [[Sn], [Sw0], [Sw]])]]), p1], [Rez]
> ]      )
> ):#Maplet
> Maplets:-Display(mplt):
> end use;
> end proc;
> end module: JOmegaModule:-runJOMg():

```

7.23.2. Интегрирование уравнения кручения

Рассмотрим решение задачи 54 на с. 155. Геометрические характеристики J_ω и J_k и секториальные площади в точках K1, K2 приложения продольных сил (рис. 221, с. 156) могут быть вычислены по программе 38, рис. 425.

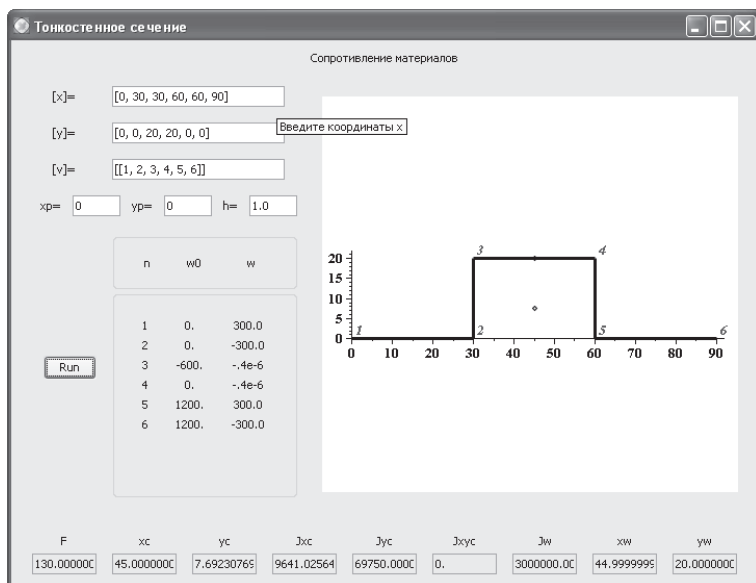


Рис. 425. P39.maplet

Все данные и промежуточные результаты приводятся в системе СИ. Дифференциальные уравнения решаются в системе оператором `dsolve`, где во временную переменную `bc` записаны краевые условия и условия сопряжения. Таким образом, нет необходимости отдельного вычисления констант интегрирования — все вычисления скрыты в операторе `dsolve`. Решение представлено в экспоненциальной форме. Для того чтобы найденные решения передать искомым функциям $\theta_1(z)$ и $\theta_2(z)$, используется оператор `assign(S)`. После интегрирования $\theta_1(z)$ и $\theta_2(z)$ по z можно построить график угла закручивания по всей длине стержня. Решения на участках склеиваем оператором для получения кусочно-заданных функций `piecewise`.

Программа 39 (P39.mws)

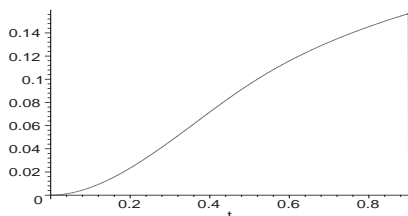
```
> restart;
> with(PDEtools,declare): declare(theta1(z)):
  Нагрузки
> M0:=3: P1:=100: P2:=-200: M[2]:=0: M[1]:=M0:
  Размеры
> l1:=0.5:l2:=0.4: l3:=0.4: l:=l1+l2:
  Моменты инерции
> Jw:=3e-12: Jd:=43.33e-12:
  Упругие свойства материала (Па). 1МПа=106Па
> E:=2e11: nu:=0.3: G:=E/2/(1+nu):
  Секториальные площади в точках K1, K2
> omegaK1:=300*1e-6: omegaK2:=-300*1e-6:
  Изгибно-крутильная характеристика
> k:=sqrt(G*Jd/(E*Jw));
  k := 2.356931947
  Дифференциальные уравнения
> k1:=k^2/G/Jd:
> eq1:=diff(theta1(z),z$2)-k^2*theta1(z)=-k1*M[1]:
> eq2:=diff(theta2(z),z$2)-k^2*theta2(z)=-k1*M[2]:
  Бимомент
> B:=omegaK1*P1+omegaK2*P2:
  Краевые условия и условия сопряжения
> bc:=theta1(0)=0,
>
  D(theta2)(1)=-B/E/Jw,
>
  D(theta1)(l1)=D(theta2)(l1),
>
  theta2(l1)=theta1(l1):
  Решение дифференциального уравнения
```



```

> S:=evalf(dsolve({eq1,eq2,bc},
>                                     {theta1(z),theta2(z)}));
> assign(S):
      S := {theta2(z) = 0.00240715 e^(2.35693z) + 0.698358 e^(-2.35693z),
      theta1 = -0.136091 e^(2.35693z) - 0.763978 e^(-2.35693z) + 0.900069}
      Угол на 1 участке (0<t<l1)
> f1:=evalf(int(theta1(z),z=0..t)):
> f11:=subs(t=l1,f1):
      Угол на 2 участке (l1<t<l1+l2)
> f2:=int(theta2(z),z=l1..t)+f11:
      Угол
> f0:=piecewise(t>0 and t<l1,f1,t>l1 and t<l,f2);
> plot(f0,t=0..l);
> Угол:=evalf(subs(t=l3,f0));

```



Угол := 0.0716696063

7.24. Балка. Маплет

Программа построения эпюр предназначена для балок с произвольным числом моментов, вертикальных сосредоточенных сил и равномерно распределенных нагрузок. Программа оформлена в виде модуля PR, содержащего две процедуры: VLK и runVLK. В первой программе производятся вычисления и построение эпюр, во второй содержится графический пользовательский интерфейс программы в виде Maplet-приложения (маплета). Структура программы следующая:

```

PR:=module()
  VLK:=proc()
    end proc;

```

```

runBLK:=proc()
    end proc;
end module:
PR:-runBLK():

```

Последний оператор программы вызывает процедуру `runBLK` созданного модуля и открывает приложение для счета. Разберем работу процедур.

Так как данные вводятся непосредственно пользователем из интерфейса программы (рис. 426) с использованием процедуры `Get` пакета `Maplets[Tools]`, а результаты потом заносятся в нужные окна программы оператором `Set`, то почти вся программа охвачена скобками `use Maplets[Tools] in — end use`.

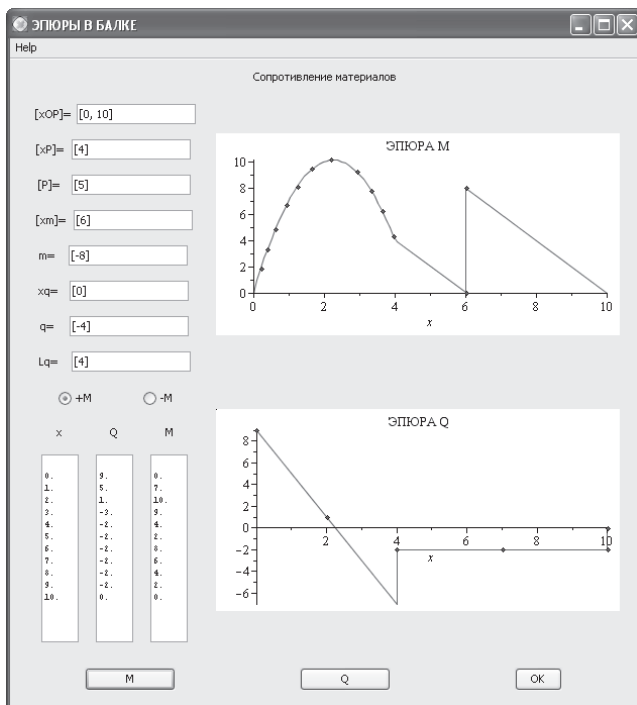


Рис. 426. P40.maplet

Координаты нагрузок и их величины записываются в списки (list). Длины списков `nm`, `nF` и `nq` определяются после ввода оператором

порс. В данном примере списки значений моментов m , распределенных нагрузок q и их координат xm и xq имеют по одной компоненте, а в число сосредоточенных сил включены и реакции опор, поэтому их три.

Функция `op` использована для преобразования списка в последовательность¹, с которой может работать оператор `max`.

Для определения реакций опор сумма моментов всех нагрузок относительно опоры A разбита на три отдельные суммы: сумму моментов сил mF , распределенных нагрузок mQ и пар mm . При построении эпюр использовалась удобная форма для кусочно-заданных функций *piecewise*, идеально подходящих для эпюр, имеющих разрывы и изломы. Эпюра моментов оформлена в виде функции $mB(x)$, эпюра поперечной силы — в виде $qB(x)$, где x — продольная координата точек балки. Начало координат принято на левом конце балки. Графики эпюр занесены оператором `plot` в переменные GM и GQ . Они же записаны последними в процедуре `BLK`, что позволяет вызвать их в окно интерфейса в виде `BLK()[1]` и `BLK()[2]`. Числовые характеристики эпюр в n равноотстоящих по оси x точках заносятся оператором `Set` в таблицы с последующим добавлением строк (`appendline`). При этом приходится преобразовывать вещественные значения в строки. Оператор `cst` выполняет это с точностью до четырех значащих цифр. Заголовок таблицы `Set` (`TB1="\n"`, `TB2="\n"`, `TB3="\n"`), помимо своего основного назначения, выполняет еще две функции — очищает таблицы от результатов предыдущего счета и с помощью команды форматирования `\n` (принятой в языках, берущих начало от Си) выводит первую строку результатов без отступа.

Опишем структуру процедуры `runBLK`, задающей интерфейс программы:

```
runBLK:=proc()
  Переменные; константы;
  use Maplets:-Elements in
  Обозначения элементов интерфейса (a,b,c,T,p);
  маплет:= Maplet(
    Evaluate(),
    ButtonGroup(),
    Window(),
    Window['Справка'](),
    MenuBar()
  ):
  Maplets:-Display(маплет):
```

¹Чтобы не запутаться в типах, близких по форме, но все-таки различных для `Maple`, рекомендуется пользоваться оператором определения типов `whattype`. Например, запись `whattype(op(xP))` дает результат: `exprseq`.

```
end use;
end proc;
```

Команда на выполнение процедуры имеет вид `Evaluate(function = 'BLK()')`. Следующая запись `ButtonGroup['Bt']()` регистрирует под именем `Bt` группу зависимых переключателей, управляющих видом эпюры моментов (на сжатом или на растянутом волокне, согласно принятому определению). В зависимости от выбранной кнопки результат записывается в логическую переменную `u` (см. процедуру `BLK`). Основное окно программы `Window` специального имени (первый параметр в квадратных скобках) не имеет, но имеет заголовок `title`, в нем зарегистрировано меню `'menubar'='MB'` с единственной кнопкой помощи. Далее записаны в столбцы и строки данные ввода, вывода и управляющие кнопки. Для наглядности записи все эти элементы определены отдельно в самостоятельные элементы `a[1]`, `a[2]` и т. д. Входные данные оформлены в отдельный столбец `BoxColumn` с перечислением списком `seq`. Далее в маплете описано окно помощи под именем `Справка`. Имя здесь необходимо, так как к нему потребуется обращаться для вызова и закрытия. В этом окне содержится один элемент `BoxColumn`, в котором, в свою очередь, записаны два элемента `TextBox` с текстом справки в виде строки (она содержится в константах программы) и кнопка закрытия `Button("Close", CloseWindow('Справка'))`. Эту кнопку для сокращения программы можно и не задавать, а закрывать окно обычным образом, нажав на крестик в правом верхнем углу. Последний элемент в маплете `MenuBar` содержит только один пункт списка `MenuItem`, в котором по щелчку мыши `onclick` открывается окно помощи `RunWindow('Справка')`.

В качестве примера используются данные задачи 61 на с. 182.

Программа 40 (P40.mws)

```
> restart; with(plots):# Маплет. Балка.
> PR:=module()
> export BLK,runBLK:
> BLK:=proc() local cst,GM,GQ,nP,nq,nm,xP,xOP,temp,
> m,xm,xq,q,l,L,mP,mq,mm,Rb,Ra,qB,mB,z,i,x,P,N,u:
> global MB,QB:
> use Maplets[Tools] in
> xOP:=Get('TxO'::list);# Координаты опор
> xP:=Get('TxP'::list); # Координаты сил
> P:=Get('TP'::list); # Силы
> m:=Get('Tm'::list); # Моменты
```

```

> xm:=Get('Txm'::list): # Координаты моментов
> xq:=Get('Txq'::list): # Координаты левых концов q
> q:=Get('Tq'::list); # Распр. нагрузки q
> l:=Get('Tl'::list); # Длины уч.распр.нагр. q
> u:=Get('RB1'::boolean);
> nP:=nops(P): # Число P
> nm:=nops(m): # Число m
> nq:=nops(q): # Число q
> P:=[op(P), 0, 0]:
> xP:=[op(xP), op(xOP)]:
> L:=max(op(xP), op(xm), op(xq+1)): #Вычисление длины
> mP:=add(P[i]*(xP[i]-xOP[1]), i=1..nP): #Моменты P
    Моменты q
> mQ:=add(q[i]*l[i]*(xq[i]+l[i]/2-xOP[1]), i=1..nq):
> mm:=add(m[i], i=1..nm): # Моменты m
    Правая реакция
> Rb:=- (mP+mQ+mm) / (xOP[2]-xOP[1]);
    Левая реакция
> Ra:=-Rb-add(P[i], i=1..nP)-add(q[i]*l[i], i=1..nq);
    Зачисление реакций опор в множество сил
> P[nP+1]:=Ra: P[nP+2]:=Rb:
    Эюра поперечной силы
> qB:=x->add(piecewise(
>             x>=xP[i], P[i], x<=xP[i], 0),
>             i=1..nP+2)+
> add(piecewise(
>     x>=xq[i] and x<=xq[i]+l[i], q[i]*(x-xq[i]),
>     x<=xq[i], 0, x>=xq[i]+l[i], q[i]*l[i]), i=1..nq):
    Эюра моментов
> z:=-1;
> if u then z:=1: end if; # Для строит.спец.z:=-1;
> cst:=x->convert(evalf(x,4), string);
    Процедура вычислений моментов
> mB:=x->add(piecewise(
>             x>=xm[i], -z*m[i], x<=xm[i], 0, 0),
>             i=1..nm)+
> add(piecewise(
>     x>=xP[i], z*(x-xP[i])*P[i], x<=xP[i], 0, 0), i=1..nP+2)+
> add(piecewise(

```

```

>      x>xq[i] and x<xq[i]+l[i], (x-xq[i])^2/2*q[i]*z,
>      x>=xq[i]+l[i], z*(x-xq[i]-l[i]/2)*l[i]*q[i], 0),
>
>      i=1..nq):
> GM:=plot(mB(x), x=0..L, title="ЭПЮРА M"):
> GQ:=plot(qB(x), x=0..L, title="ЭПЮРА Q"):
> N:=10:
> Set (TB1=" \n", TB2=" \n", TB3=" \n"):
> for i from 0 to N do
>   x:=L/N*i:
>   Set(TB1(appendline)=cst(x)):
>   Set(TB2(appendline)=cst(qB(x))):
>   Set(TB3(appendline)=cst(mB(x))):
> od:
> end use:
> GM, GQ:
> end proc:
> runBLK:=proc()
> local маплет, z, a, L, b1, b2, T1, T2, T3, c1, c2, c3,
>
>      p1, p2, helpStr;
> z:='width'=5, 'height'=15, 'editable'='false':
> helpStr:="Момент положительный, если он вращает
> против часовой стрелки. Силы и распределенные
> нагрузки, направленные вверх, положительные.
> Начало координат в левом конце балки.":
> use Maplets:-Elements in
> a[1]:=["[xo]=", TextField['TxO'](14, 'value'=[0, 10])]:
> a[2]:=["[xP]=", TextField['TxP'](14, 'value'=[4])]:
> a[3]:=["[P]=", TextField['TP'](14, 'value'=[5])]:
> a[4]:=["[xm]=", TextField['Txm'](14, 'value'=[6])]:
> a[5]:=["m=", TextField['Tm'](14, 'value'=[-8])]:
> a[6]:=["xq=", TextField['Txq'](14, 'value'=[0])]:
> a[7]:=["q=", TextField['Tq'](14, 'value'=[-4])]:
> a[8]:=["Lq=", TextField['Tl'](14, 'value'=[4])]:
> b1:=RadioButton['RB1']("+M", 'value'=true,
> 'group'='Bt', 'tooltip'="Эпюра на сжатом волокне"):
> b2:=RadioButton['RB2']("-M", 'value'=false,
>
>      'group'='Bt'):

```

```

> T1:=TextBox['TB1'](z):
> T2:=TextBox['TB2'](z):
> T3:=TextBox['TB3'](z):
> c1:=Button('width=90',"M",
>           Evaluate('PL1' = 'display(BLK()[1])')):
> c2:=Button('width=90',"Q",
>           Evaluate('PL2' = 'display(BLK()[2])')):
> c3:=Button("OK",Shutdown(['TxP','TP','Txm'])):
> p1:=Plotter['PL1']('height=200'):
> p2:=Plotter['PL2']('height=200'):
> маплет:= Maplet(
>   Evaluate(function = 'BLK()'),
>   ButtonGroup['Bt'](),
>   Window(
>     'title'="ЭПЮРЫ В БАЛКЕ",
>     'menubar'='MB',
>     ["Соппротивление материалов",
>     [BoxColumn(seq(a[i],i=1..8),
>     [b1,b2],["x","Q","M"],[T1,T2,T3]),
>     [p1,p2]], [c1,c2,c3]],
>   Window['Справка'](
>     BoxColumn(TextBox('height'=10,'value'=helpStr),
>     Button("Close",CloseWindow('Справка')))),
>   MenuBar['MB']( Menu(
>     "Help",
>     MenuItem(
>       "Правило знаков",
>       'onclick'=RunWindow('Справка'))))
> ):#Maplet
> Maplets:-Display(маплет):
> end use;
> end proc;
> end module:
> PR:-runBLK():

```

7.25. Статически определимая рама

7.25.1. Алгоритм

Рассмотрим работу программы построения эпюр моментов и перерезывающих сил в статически определимых рамах. Программа состоит из основной части (программа 41), где из специально подготовленного текстового файла вводятся исходные данные, и нескольких подпрограмм для расчета (EprigM, с. 440; RQ, EM, Lxy, с. 442) и отображения результатов (RisRama, PrintRez, с. 442; RisEprig, с. 444), размещенных в программах 42 и 43 и вызываемых из основной оператором `read`.

Алгоритм расчета следующий. Рама разбивается на участки — отдельные стержни или части стержней так, чтобы нагрузка (силы и моменты) была приложена по концам участка, а распределенная нагрузка действовала на всю длину либо отсутствовала. В отличие от задач определения реакций опор составной конструкции в статике, где для того чтобы не добавлять в число неизвестных реакций моменты, сечения производятся только по шарнирам, здесь сечения проводим по участкам нагрузок и сочленениям (например, шарнирам), поэтому число участков зависит и от нагрузки.

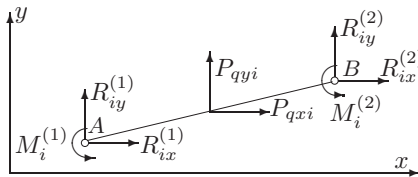


Рис. 427

Выделенный участок (стержень) i имеет шесть реакций. Стержень представляется в виде вектора, началу которого присписывается номер 1, концу — 2. Реакции в начале стержня $M_i^{(1)}$, $R_{ix}^{(1)}$, $R_{iy}^{(1)}$, в конце — $M_i^{(2)}$, $R_{ix}^{(2)}$, $R_{iy}^{(2)}$ (рис. 427).

Уравнения равновесия i -го участка ($i = 1, 2, \dots, n$) в программе EprigM (с. 440) имеют вид

$$\begin{aligned} \sum X_i &= R_{ix}^{(1)} + R_{ix}^{(2)} + P_{qxi} = 0, \\ \sum Y_i &= R_{iy}^{(1)} + R_{iy}^{(2)} + P_{qyi} = 0, \\ \sum M_A &= -R_{ix}^{(2)}(y_B - y_A) + R_{iy}^{(2)}(x_B - x_A) - P_{qxi}(y_B - y_A)/2 + \\ &\quad + P_{qyi}(x_B - x_A)/2 + M_i^{(1)} + M_i^{(2)} = 0. \end{aligned} \quad (7.9)$$

Для равновесия m узлов необходимы условия

$$\begin{aligned} -\sum R_{xi} + \sum P_{xi} + \sum R_{xi}^o &= 0, \\ -\sum R_{yi} + \sum P_{yi} + \sum R_{yi}^o &= 0, \\ -\sum M_i + \sum M_i^e + \sum M_i^o &= 0, \end{aligned} \quad (7.10)$$

где R_{xi}^o , R_{yi}^o , M_i^o — реакции связей, P_{xi} , P_{yi} , M_i^e — узловые нагрузки. Знак минус у первых слагаемых в системе уравнений равновесия узлов возник из-за того, что реакции стержней и узлов противоположны по направлениям.

В подпрограмме `ЕРАМА`, с. 440, формируется и решается система уравнений (7.9–7.10). Система уравнений состоит из трех списков. Первые $3m$ уравнений — это уравнения равновесия сил, действующих на стержни, где m уравнений — это уравнения проекций на ось x , следующие m — уравнения проекций на y , затем записаны m уравнений моментов относительно точки 1. Второй список состоит из $6m$ уравнений `Eq[k][i]` равновесия узлов. Индекс $k=1$ соответствует проекции на ось x , $k=2$ — на ось y , $k=3$ соответствует равенству нулю моментов.

Нагрузки и неизвестные реакции опор, приложенные к узлам, добавляются в уравнения уже после того, как эти уравнения сформировались в тройном цикле по стержням (1,...,m), номерам концов (1,2) и номерам уравнений (1,2,3).

Последний список из ks уравнений необходим только для составных конструкций. Уравнения эти наиболее просты. Они выражают равенство нулю определенных внутренних реакций, в результате чего образуются различного рода сочленения. Например, равенство нулю момента приводит к внутреннему шарниру. Вектор `Vns1` для статически определимых систем состоит из нулей. Он введен для того, чтобы можно было рассчитывать и статически неопределимые системы, где, согласно методу сил, иногда надо прикладывать единичные внутренние усилия.

С помощью оператора перечисления `seq` легко составляется вся система уравнений `sstm` и список неизвестных `unp`. Отметим, что искомые величины в процедуре должны быть локальными. Для использования их вне процедуры применяются глобальные переменные `Rx`, `Ry`, `Rm`, `Ro`.

Подпрограмма `RQ` (с. 442) вычисляет перерезывающие силы по реакциям концов стержня. Правило знаков: положительные перерезывающие силы вращают стержень по часовой стрелке. При расчете учтено, что $\text{sign}(a)\text{sign}(b) \neq \text{sign}(ab)$, а на стержень не действуют сразу две распределенные нагрузки (поэтому использована сумма $Q_x + Q_y$). При необходимости можно написать аналогичную программу для определения нормальных сил.

Подпрограмма `PrintRez` служит для отображения результатов в виде таблиц значений моментов и перерезывающих сил.

Подпрограмма `EM` возвращает зависимость момента от линейной координаты v (координаты вдоль стержня k) по данным на концах стержня и величине распределенной нагрузки. В условиях поставленной задачи с равномерно распределенной нагрузкой это может быть парабола или прямая. Найденная зависимость используется в формуле Максвелла–Мора.

Подпрограмма `RisRama` отображает на экране стержни рамы. Это изображение сначала служит для контроля ввода данных, потом для вывода эпюр моментов и перерезывающих сил. В подпрограмме используются процедура рисования линии `Линия(i, j)`, соединяющей точки с номерами i и j , и параметр `setN`, содержащий характеристики шрифта. Подпрограмма `Lxy` (с. 442) находит массивы длин участков и их проекции.

7.25.2. Файл данных о раме

Все данные о раме и нагрузках записываются в отдельный файл. Данные в файл записываются через запятую по строкам. Оператор `readline` по строчкам считывает из файла данные. Так как этот оператор дает результат в виде строки вида "1, 2, 3", то для снятия кавычек и превращения данных в список применяется оператор `parse`. Рассмотрим файл данных к задаче 64, с. 191. Слева в рамке записан файл `r64.txt`, справа — пояснения (они в файл не попадают).

1)	0, 8, 12, 16, 0, 4, 8, 0	Координаты x
2)	10, 10, 10, 10, 5, 5, 5, 0	Координаты y
3)	0, 0, 0, 0, 0, 0, 0, 0	Горизонтальная нагрузка P_1
4)	0, 0, 10, 0, 0, 10, 0, 0	Вертикальная нагрузка P_2
5)	0, 0, 0, 0, 0, 0, 0, 0	Моменты в узлах P_3
6)	[1,2],[2,3],[3,4],[5,1],[5,6],[6,7],[7,2],[8,5]	Стержни [начало-конец]
7)	[1,2,3],[4,2,3],[7,2,3],[5,2,3]	Внутренние соединения
8)	Опоры:[Узел, направление(1-Х,2-У,3-М)]	Комментарий
9)	[8,1],[8,2],[4,1],[4,2]	Реакции опор [узел, напр.]
10)	0, 0, 0, 0, 0, 0, -4, 0	Горизонт. распр. нагр. Q_x
11)	0, 0, 5, 0, 0, 0, 0, 0	Вертик. распр. нагр. Q_y
12)	1, 1, 1, 1, 1, 1, 1, 1	Жесткость стержней EJ
13)	2,6	Направл. и место смещения

Рис. 428. Файл данных задачи 64 для программы 41

Для подготовки данных о раме сначала надо пронумеровать узлы и стержни (рис. 429). Номера узлов на рисунке выделены курсивом, стержней — полужирным шрифтом. Вводим систему координат с началом в опоре A . Первые две строки файла — координаты узлов. Пробелы в записи игнорируются. Оператор `readline` заносит эти

данные в списки x и y , а оператор `pops` подсчитывает число узлов. Следующие три строки — узловые нагрузки. Это сосредоточенные горизонтальные нагрузки P_1 , вертикальные P_2 и пары P_3 . Положительные P_1 направлены направо, P_2 — вверх, пары P_3 — против часовой стрелки. Первые пять строк файла образуют матрицу $5 \times n$, где n — число узлов. Шестая строка содержит информацию о стержнях и

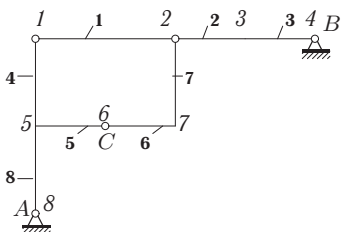







Рис. 429

записана в виде последовательности пар. Первое число пары — номер начальной точки стержня (участка), вторая — конечная точка. Начало и конец выбираются произвольно, но для ввода дальнейшей информации, в частности при описании внутренних соединений, выбранная ориентация стержня учитывается. Число квадратных скобок

равно числу стержней. Номера концов стержней заносятся в список `str`. Число стержней вычисляется оператором `m:=pops(str)`. В *седьмой* строке указывается тип внутренних соединений. Если рама не имеет таких соединений (задача 1), то эта строка пустая. Закрывающие и открывающие скобки тоже не пишутся. Если конструкция составная, то в зависимости от типа соединения¹ в точке сочленения необходимо положить нулю соответствующую реакцию. Это дает дополнительное уравнение к системе (7.9), (7.10). Число этих уравнений в программе 41 обозначается `ks`. Для цилиндрического шарнира  необходимо уравнение $M_i = 0$, для горизонтально скользящей заделки  — $R_{ix} = 0$, а для вертикально скользящей заделки  — $R_{iy} = 0$. Сочленения записываются в виде списка скобок по три числа в каждой. Первое число — номер стержня, которому принадлежит сочленение, второе (1 или 2) указывает на начало (1) или конец (2) стержня, третье — тип реакции, значение которой полагается равным нулю (1 — горизонтальная сила, 2 — вертикальная сила, 3 — пара). Например, запись [11, 2, 3] означает, что в стержень 11 в конце закреплен цилиндрическим шарниром, а запись [10, 1, 2] соответствует вертикальной скользящей заделке в начале (второе число 1) стержня номер 10. Кратные шарниры должны быть описаны для всех входящих в шарнир стержней, кроме одного. Этот стержень может быть произвольным, но если к такому шарниру приложить пару, то она будет действовать именно на него. В данном примере узел 2 представляет собой двукратный шарнир (соединены три стержня), поэтому в *седьмой*

¹В этой программе используются соединения, содержащие только две реакции. Это может быть либо шарнир, либо скользящая заделка.

строке записываем [1,2,3],[7,2,3]. Одну из этих скобок можно заменить информацией о стержне 3, который также подходит к узлу 2: [3,1,3]. *Восьмая* строка приведена как пример комментария. Такие строки можно было предусмотреть перед каждой строкой с данными. Она просто считывается оператором `readline` и никуда не заносится. В ней указано правило ввода данных об опорах. Соответствующая строка содержит набор пар целых чисел в квадратных скобках. Первый элемент пары — номер узла, второй — номер направления реакции. Направление силы вдоль оси x (горизонтально направо) имеет номер 1, вдоль y — 2, пара сил (против часовой стрелки) — номер 3. Например, запись [1, 1], [1, 2], [1, 3], [3, 2] означает, что конструкция имеет в точке 1 заделку  (с тремя реакциями опор), а в точке 3 вертикальный опорный стержень (или, что то же, шарнир  с горизонтальной подвижностью).

Следующие три строки образуют матрицу, в которой число столбцов равно числу участков. В двух строках указаны распределенные нагрузки: сначала — горизонтальные Qx (положительные направо), потом — вертикальные Qy (положительные вверх). Есть одно ограничение: на один стержень не может действовать и вертикальная, и горизонтальная нагрузка. Это ограничение введено для упрощения программы и при желании его легко снять. Предпоследняя строка — относительные жесткости стержней на изгиб¹. Обычно все жесткости одинаковые и строчка состоит из единиц по числу участков. Если жесткость какого-либо стержня, например третьего, вдвое больше остальных, то на третьем месте в строке ставится 3, а остальные элементы равны 1.

В последней строке указываются два числа — направление единичной силы (1, 2 или 3) и номер узла, к которому она прикладывается. Эта сила указывает на искомое перемещение рамы. Если требуется только построить эпюру или найти реакции опор рамы, то эта информация лишняя, и можно просто написать 1,1.

Общее число уравнений в системе равно $3n + 3m + ks$, число неизвестных $6m + ko$. Эти числа выводятся на экран, и, если они не сходятся, то следует искать ошибку в моделировании рамы. Обычно бывают ошибки с неправильным указанием промежуточных шарниров или опор.

Замечание. Считать данные из файла можно также с помощью оператора `readdata` (с. 241, 349) или оператором `fscanf`. В последнем случае это выглядит, например, так:

```
> Файл := fopen("C:/Maple/P64.txt", READ);
> Данные := op(fscanf(Файл, "%hg"));
> x := convert(Данные[1], list);
> y := convert(Данные[2], list);
```

¹Произведение момента инерции I на модуль упругости (кНм²).

Числа в файл при этом записываются через пробел в виде таблицы. Именно поэтому такой способ не очень удобен в рассматриваемой задаче, где данные представлены списками разной длины.

7.25.3. Основная программа. Примеры

Пример 1

Приведем решение задачи 64, с. 191, для которой файл дан на с. 430.

Программа 41 (P41.mws)

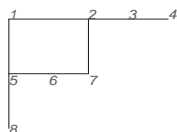
```
> restart:# Основная программа. Ввод данных, вычисления
> read "C:\\\epura.m"; # Программа 42 с. 440
> read "C:\\\Riseपुरa.m";# Программа 43 с. 444
> read "C:\\\Podpr1.m"; # Программа 9 с. 356
```

Ввод данных

```
> fd := fopen("C:/M13/p64.txt",READ):#
> x:=[parse(readline(fd))];# Координаты узлов
> y:=[parse(readline(fd))];
> for i to 3 do# Нагрузки в узлах
> P||i:=[parse(readline(fd))];
> od;
> str:=[parse(readline(fd))]; #Стержни. Номера узлов
> Vns:=[parse(readline(fd))]; #Внутренние шарниры
> readline(fd); #Строка-комментарий
> OPOR:=[parse(readline(fd))];#Опоры
> Qx:=[parse(readline(fd))];#Qx Стержни. Распр. нагр.
> Qy:=[parse(readline(fd))];#Qy
> EJ:=[parse(readline(fd))];#Относительная жесткость
> Napravl,Nuzel:=parse(readline(fd));#Место прогиба
> fclose(fd);
> n:=nops(x): m:=nops(str): # Число узлов и стержней
> ks:=nops(Vns): #Число внутренних шарниров
> Vns1:=Vector(n): #Нулевой вект. внутр. мом. в шарн.
> ko:=nops(OPOR): #Количество опор
> printf("Число неизвестных %d + %d, уравнений
> %d + %d + %d",ko,6*m,ks,3*m,3*n);
Число неизвестных 4 + 48, уравнений 4 + 24 + 24
```

Решение

```
> Lxy():RisRama(); # Вычисл. длин участков, рисунок
```



```
> ЕpurаM(P1, P2, P3):#Моменты
```

```
> Реакции=seq(Ro[i], i=1..ko);
```

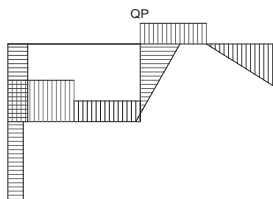
Реакции = (-7, -20, 27, -20)

```
> RQ(Rx, Ry): #Поперечная (перерезывающая) сила
```

```
> PrintRez(Q1, "Qp");
```

N	Qp(1)	Qp(2)
1	0.00	0.00
2	-10.00	-10.00
3	0.00	20.00
4	9.00	9.00
5	-20.00	-20.00
6	-10.00	-10.00
7	-2.00	18.00
8	7.00	7.00

```
> QP:=Matrix(m, 2, Q1): RisEpur(QP, 'QP', 0);
```

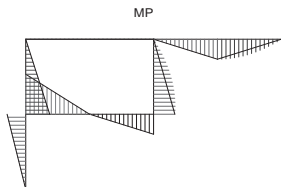


```
> MP:=Matrix(m, 2, Rm):
```

```
> RisEpur(MP, 'MP', 1);
```

```
> PrintRez(Rm, "Mp");
```

```
> Qx0:=Qx: Qy0:=Qy: #Сохраняем данные (распр.нагр)
```



N	Mr(1)	Mr(2)
1	0.00	0.00
2	0.00	-40.00
3	40.00	0.00
4	45.00	0.00
5	-80.00	0.00
6	0.00	-40.00
7	40.00	0.00
8	0.00	35.00

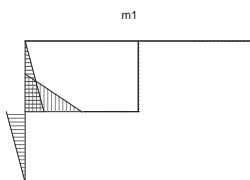
Обнуляем нагрузки

```

> P1:=Vector(n):#Горизонтальная сила
> P2:=Vector(n):#Вертикальная сила
> P3:=Vector(n):#Моменты
> Qx:=Vector(m): Qy:=Vector(m):#Распределенные нагр.
> P||Napravl[Nuzel]:=1; #Прикладываем ед.нагр.
                                P26 := 1

> EпuraM(P1,P2,P3):
> m1:=Matrix(m,2,Rm):
> RisEпur(m1,"m1",1);

```



```

> PrintRez(m1,"m1");

```

N	m1(1)	m1(2)
1	0.00	0.00
2	0.00	0.00
3	0.00	0.00
4	2.00	0.00
5	-4.00	0.00
6	0.00	0.00
7	0.00	0.00
8	0.00	2.00

*Перемножаем эпюры Mr, m1 (процедура EM из файла eпura.m)
Получаем прогиб:*

```

> Delta1:=evalf(add(int(EM(MP,1,u,k)*EM(m1,0,u,k)/
>                                EJ[k],u=0..L[k]),k=1..m));
                                Δ1 := 693.3333333.

```

Пример 2

Рассмотрим ввод данных к задаче из сборника задач [46], с. 72. На рисунке 430 даны размеры рамы (в метрах), равномерно распределенная вертикальная нагрузка $q = 4$ кН/м. Жесткость всех участков одинаковая. Данные для программы 41 занесены в файл MMik.txt. Начало координат находится в опоре 1. Первые две строки файла — координаты узлов, три следующие строки содержат только нули, т. к. сосредоточенные силы и моменты к раме не приложены. Далее идет строка с номерами концов стержней, потом указан внутренний шарнир. Запись [3,2,3] означает, что стержень 3 (он идет от точки 3 к точке 4) в конце, на который указывает число 2, имеет шарнир, т. е. реакция № 3 (а это момент) равна нулю. Следующая строка — опоры. Их две, но на них приходится четыре реакции, описанные в этой строке скобками. В каждой скобке первое число — номер узла, второе — направление реакции. В следующих трех строках даны распределенные нагрузки, приложенные к стержням и относительные жесткости стержней. Последняя строка нужна для определения перемещения. В данном случае определяется угол поворота (направление 3) в узле 2.

1)	1, 3, 3, 5, 6, 0
2)	0, 0, 3, 3, 3, 3
3)	0, 0, 0, 0, 0, 0
4)	0, 0, 0, 0, 0, 0
5)	0, 0, 0, 0, 0, 0
6)	[1,2],[2,3],[3,4],[4,5],[6,3]
7)	[3,2,3]
8)	Опоры: [Узел,направл.(1-X,2-Y,3-M)]
9)	[1,2],[5,1],[5,2],[5,3]
10)	0, 0, 0, 0, 0
11)	0, 0, -4, -4, -4.
12)	1, 1, 1, 1, 1
13)	3,2

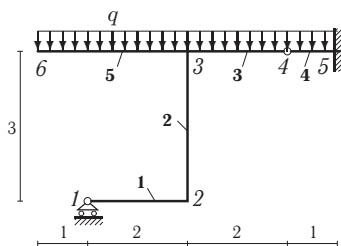


Рис. 430. Файл данных MMik.txt для программы 41

Пример 3

Дана рама [45], нагруженная двумя силами $P = 15$ кН, рис. 431. Жесткость всех участков одинаковая: $EJ = 1,144 \cdot 10^6$ Нм², $a = 1,1$ м, $b = 0,55$ м. Необходимо найти вертикальное перемещение точки 5¹. Данные для программы 41 занесены в файл SoplM-M.txt. Внутренних

¹В сборнике [45] эта задача (модель опоры турбогенератора) решается средствами Mathcad. Жесткость EJ по условию задачи подбирается из условия прочности и определяется перемещение точки 5 за счет температурного воздействия.

шарниров в раме нет, поэтому строка 7 пустая. Строки 4 и 5 состоят из нулей, так как к раме не приложены внешние вертикальные силы и моменты в узлах.

1)	0,	1,1,	1,1,	1,1,	0
2)	0.825,	0.825,	0.275,	0,	0.275
3)	0,	-15,	-15,	0,	0
4)	0,	0,	0,	0,	0
5)	0,	0,	0,	0,	0
6)	[1,2],[2,3],[3,4],[3,5]				
7)					
8)	Опоры:[Узел,направл.(1-X,2-Y,3-M)]				
9)	[1,2],[4,1],[4,2]				
10)	0, 0, 0, 0				
11)	0, 0, 0, 0				
12)	1, 1, 1, 1				
13)	2,5				

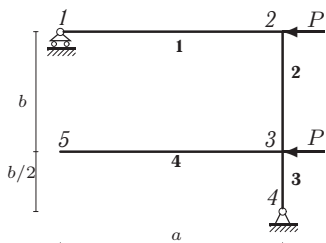


Рис. 431. Файл данных SorpM-M.txt для программы 41

Пример 4

Дана рама, состоящая из двух частей, соединенных скользящей заделкой (рис. 432). Рама закреплена на одном неподвижном шарнире и двух подвижных. К верхней части рамы приложена пара $M=11$ кНм, к нижней — две наклонные силы $P=4$ кН, $Q=2$ кН. Заданы углы: $\alpha=30^\circ$, $\beta=45^\circ$. Построить эпюру моментов и перерезывающих сил в раме.

Файл данных r41a.txt для программы 41 имеет вид

1)	14,	14,	7,	0,	0,	21,	21
2)	0,	4,	4,	4,	10,	10,	14
3)	0,	-1.41,	0,	3.464,	0,	0,	0
4)	0,	-1.41,	0,	2,	0,	0,	0
5)	0,	0,	0,	0,	0,	-11,	0
6)	[1,2],[2,3],[3,4],[4,5],[5,6],[6,7]						
7)	[4,2,2]						
8)	Опоры:[Узел,направл.(1-X,2-Y,3-M)]						
9)	[1,2],[3,2],[7,1],[7,2]						
10)	0, 0, 0, 0, 0, 0, 0						
11)	0, 0, 0, 0, 0, 0, 0						
12)	1, 1, 1, 1, 1, 1, 1						
13)	3,5						

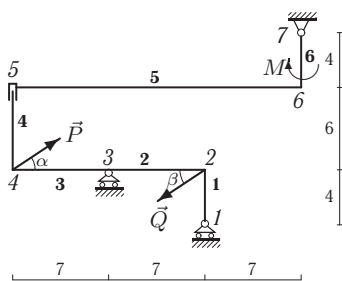


Рис. 432

Приведем результаты счета по программе 41:

Эюры моментов и перерезывающих сил

N	Qp(1)	Qp(2)	N	Mp(1)	Mp(2)
1	0.00	0.00	1	0.00	0.00
2	-0.64	-0.64	2	0.00	-4.50
3	2.00	2.00	3	4.50	9.50
4	-2.05	-2.05	4	-9.50	-2.80
5	0.00	0.00	5	2.80	-2.80
6	-2.05	-2.05	6	-8.20	0.00

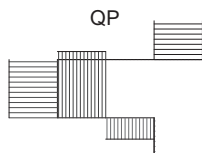


Рис. 433

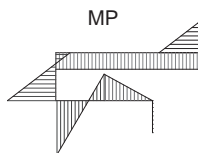


Рис. 434

Пример 5

Рассмотрим задачу 5, с. 19. Найдем реакции опор и построим эюры в раме. Пронумеруем узлы и стержни, рис. 435. Начало координат поместим в узел 2. Момент, приложенный к стержню BD , для определенности сосредоточим в специально введенном узле 10. Заметим, что этот узел можно и не вводить, и приложить момент к узлу 5, но тогда программа отнесет его к тому стержню, конец которого не описан во внутренних связях. За этим можно не уследить и допустить ошибку. В данном случае момент будет автоматически приложен к части DC . Поэтому лучше ввести дополнительный узел, тем более при этом эюра моментов примет более реальный вид.

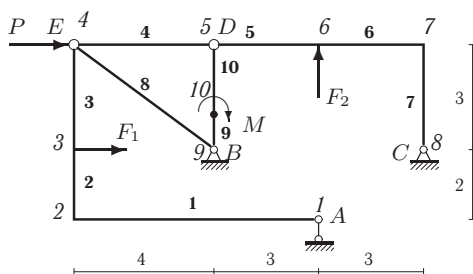


Рис. 435

1)	7, 0, 0, 0, 4, 7, 10, 10, 4, 4
2)	0, 0, 2, 5, 5, 5, 5, 2, 2, 3
3)	0, 0, 7, 4, 0, 0, 0, 0, 0, 0
4)	0, 0, 0, 0, 0, 2, 0, 0, 0, 0
5)	0, 0, 0, 0, 0, 0, 0, 0, 0, -3
6)	[1,2],[2,3],[3,4],[4,5],[5,6],[6,7],[7,8],[9,4],[9,10],[10,5]
7)	[3,2,3],[8,2,3],[10,2,3],[9,1,3],[4,2,3]
8)	Опоры: [Узел, направление (1-X, 2-Y, 3-M)]
9)	[1,2],[8,1],[8,2],[9,1],[9,2]
10)	0, 0, 0, 0, 0, 0, 0, 0, 0, 0
11)	0, 0, 0, 0, 0, 0, 0, 0, 0, 0
12)	1, 1, 1, 1, 1, 1, 1, 1, 1, 1
13)	1,2

Рис. 436. Файл данных r5.txt для программы 41

В узле 4 надо учесть его кратность. Этот шарнир двукратный, поэтому в списке внутренних соединений (седьмая строка файла) стоят две скобки [3,2,3] и [8,2,3], т. е. стержни 3 и 8 в конце имеют шарниры. Узел 9 однократный шарнир. Скобка [9,1,3] означает, что стержень 9 в начале имеет шарнир. Шарнир 5 двукратный, к нему подходит три стержня. Таким образом в этой строке столько скобок, сколько шарниров в конструкции с учетом их кратности¹.

Цифры 1,2 в строке 13 означают, что определяется горизонтальное перемещение в узле 2. Точнее, определяется числитель в формуле $\delta = \int m_1 M_p ds / EJ$. Для определения перемещения (в метрах) результат надо поделить на жесткость EJ . Модуль упругости берется в Па, момент инерции в м⁴.

Пример 6. Балка

Построим эпюры моментов и перерезывающих сил в балке 61 на с. 182. В следующем файле данных (рис. 437) указаны координаты узлов (начало координат на левом конце), нагрузки, стержни и опоры. Строка 7 пустая, так как внутренних шарниров нет. Согласно строке 13 определяется вертикальное смещение левой консольной части балки. Расчет показывает, что смещение равно $92/EJ$. В эту формулу остается подставить жесткость в кНм² и получить смещение в метрах. Жесткость берем в кНм², так как и нагрузки даны в кНм, кН/м, и кН. Смещение получилось положительным, т. е. вверх. Заметим, что этот вариант расчета выгодно отличается от других тем, что здесь можно рассчитывать балку переменной жесткости

¹Кратность шарнира — число стержней на шарнире без единицы.

- | | |
|-----|---------------------------------------|
| 1) | 0, 4, 6, 10 |
| 2) | 0, 0, 0, 0 |
| 3) | 0, 0, 0, 0 |
| 4) | 0, 5, 0, 0 |
| 5) | 0, 0, -8, 0 |
| 6) | [1,2],[2,3],[3,4] |
| 7) | |
| 8) | Опоры:[Узел,направление(1-X,2-Y,3-M)] |
| 9) | [1,1],[1,2],[4,2] |
| 10) | 0, 0, 0 |
| 11) | -4, 0, 0 |
| 12) | 1, 1, 1 |
| 13) | 2,3 |

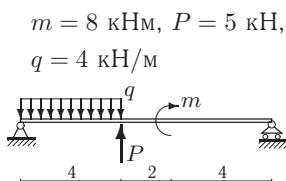


Рис. 437. Файл данных balka.txt для программы 41

Файлы данных для решения задач 1, 2, 3, 4, 6, 7 (определение реакций рам, глава 1) по программе 41 находятся в архиве <http://vuz.exponenta.ru/progr.rar> под именами p1.txt, p2.txt и т. д.

7.25.4. Подпрограммы

Все расчетные подпрограммы занесены в отдельный файл `epura.m`. Этот файл является результатом выполнения программы 42. Основная программа 41 считывает подпрограммы из `epura.m` оператором `read` так, как если бы они были записаны в этом месте. Для обмена информацией между файлами лучше использовать абсолютные адреса. В данном случае файлы считываются из корневой директории диска C.

Программа 42 (P42.mws)

К программе 41, с. 433

Подпрограммы `EpuraM`, `RQ`, `PrintRez`, `EM`, `RisRama`, `Lxy`

```

> restart;
> with(plots): with(plottools):
> EpuraM:=proc(P1,P2,P3)
> local R,i,j,k,Lx,Rr,Ly,eq,eqs,S,Eq,sstm,unn:
> global Rm,Rx,Ry,Ro:
> for k to 3 do
>   for i to 3*n do
>     Eq[k][i]:=0;
>   od:
> od:
> for i to m do # Уравн. равновесия стержней (7.9)
>   Lx:=L1[i]:   Ly:=L2[i]:# См. Lxy, с. 443

```

```

> eq[i]:=R[1][i,1]+R[1][i,2]+Qx[i]*abs(Ly);# (7.9)
> eq[i+m]:=R[2][i,1]+R[2][i,2]+Qy[i]*abs(Lx);
> eq[i+2*m]:=R[2][i,2]*Lx-R[1][i,2]*Ly+
>           Qy[i]*abs(Lx)*Lx/2-Qx[i]*abs(Ly)*Ly/2
>           +R[3][i,1]+R[3][i,2];
> for k to 3 do #Равновесие узлов, (7.10)
>   for j to 2 do
>     Eq[k][str[i,j]]:=Eq[k][str[i,j]]-R[k][i,j];
>   od;
> od;
> od:
> for i to n do #Нагрузки на узлы
>   for k to 3 do
>     Eq[k][i]:=Eq[k][i]+P||k[i]:
>   od:
> od:

Vns[i,1] — номер стержня (1...m), Vns[i,2] — номер конца (1,2),
Vns[i,3] — направление реакции (1-х,2-у,3-т)
> for i to ks do #Внутренний шарнир на стержне
>   eqs[i]:=R[Vns[i,3]][Vns[i,1],Vns[i,2]]+Vns1[i]:
> od;
> for i to ko do #Опора. Rr - реакции опор
>   Eq[OPOR[i,2]][OPOR[i,1]]:=
>           Eq[OPOR[i,2]][OPOR[i,1]]+Rr[i]:
> od;
  Все уравнения системы
> sstm:=seq(eq[i],i=1..3*m),
>           seq(seq(Eq[k][i],k=1..3),i=1..n),
>           seq(eqs[i],i=1..ks);
  Все неизвестные системы
> unn:=seq(Rr[i],i=1..ko),
>           seq(seq(seq(R[k][i,j],k=1..3),i=1..m),j=1..2);
  Решение
> S:=solve({sstm},{unn}):assign(S):
> Rx:=R[1]; Ry:=R[2]; Rm:=R[3]; Ro:=Rr;
> end proc:#EपुरaM
> #####

```

Определение поперечных сил

```

> RQ:=proc(Rx,Ry) local QQ,k,sn,cs,Qs:
> global Q1:
> QQ:=x->Ry[k,1]*cs-Rx[k,1]*sn+Qs*x:
> for k to m do
>   sn:=L2[k]/L[k]:   cs:=L1[k]/L[k]:
>   Qs:=(Qx[k]+Qy[k])*sign(-L2[k])*sign(L1[k]):
>   Q1[k,1]:=QQ(0):   Q1[k,2]:=QQ(L[k]):
> od;
> end proc:#RQ
> #####
Печать таблиц результатов (эпюры моментов и сил)
> PrintRez:=proc(M,Txt) local i,M1:
> printf("\n N      %s(1)      %s(2) \n",Txt,Txt);
> for i to m do
>   printf("%d %7.2f %7.2f \n",i,M[i,1],M[i,2]);
> od;
> end proc:#PrintRez
> #####
Получение функции M(v) по эпюре.
z=0, если нет распределенных нагрузок (для единичн. нагрузок)
> EM:=proc(M,z,v,k) M[k,1]-(M[k,2]+M[k,1])/L[k]*v+
>                               z*(Qx0[k]+Qy0[k])*v*(L[k]-v)/2*
>                               sign(-L2[k])*sign(L1[k]):
> end proc:#EM
> #####
Рисунок каркаса рамы
> RisRama:=proc()
> local Ns;
> global rama;
> Ns:=PLOT(seq(TEXT([x[i],y[i]],convert(i,symbol),
>                               ALIGNRIGHT),i=1..n),setN):
> rama:=seq(Линия(str[i,1],str[i,2]),i=1..m):
> display(rama,Ns,axes=NONE);
> end proc:#RisRama
> #####
Вычисление длин (L) и проекций (L1, L2) участков
> Lxy:=proc()

```

```

> local k;
> global L1,L2,L:
> for k to m do      # Длины участков
>   L1[k]:=x[str[k,2]]-x[str[k,1]]:
>   L2[k]:=y[str[k,2]]-y[str[k,1]]:
>   L[k]:=sqrt(L1[k]^2+L2[k]^2)
> od:
> end proc:#Lxy
> #####
   Запись процедур на диск
> save Lxy,RisRama,EM,PrintRez,RQ,EपुरaM,"C:\\epura.m";

```

Подпрограмма `RisEपुर` может строить эпюры моментов и перерезывающих сил. При построении эпюры моментов аргументом `Rm` является матрица моментов, а параметр учета кривизны эпюры `z=1`. Для эпюры перерезывающих сил `z=0`.

Цвета эпюр соседних участков берутся разными. В формате RGB первое и последнее число получается как элемент векторов с чередующимися числами. Применен необычный способ заполнения вектора с помощью тут же созданной функции деления по модулю 2 или 3. При $m = 5$ имеем `clb:=Vector([1,0,1,0,1])` и `clr:=Vector([0.5,1,0,0.5,1])`.

Для масштабирования эпюр сначала определяем максимальную длину участка `maxL`, затем плотность штриховки `s0:=maxL/40`, где коэффициент 40 подбирается вручную (для большинства задач это число подходит).

Максимальное значение абсолютного значения элемента матрицы определяется в несколько этапов. Оператором `convert` матрица преобразуется в вектор `r1`, затем оператор `abs` с помощью `map` применяется к каждому элементу вектора `r1`, и результат, опять с помощью `convert`, превращается в список, максимальное значение которого можно найти, предварительно сняв скобки оператором `op`. Полученное значение делится на `maxL/3` (число 3 взято достаточно произвольно, для того чтобы ординаты эпюры составляли не более 1/3 максимальной длины участка).

Эпюры моментов строятся так, как принято в машиностроительных вузах — на сжатом волокне. Для архитектурно-строительных специальностей, где эпюра моментов строится на растянутом волокне, `Масштаб` должен быть положительным.

Для подбора масштаба можно также использовать норму матрицы `MatrixNorm(Rm,1)` из пакета `LinearAlgebra`.

В некоторых задачах эюры могут быть вырожденными — все значения матрицы равны нулю. Условный оператор `if Масштаб<>0 then ... fi` предусмотрен как раз на этот случай.

Программа 43 (P43.mws)

К программе 41, с. 433

Подпрограмма RisEpur

```
> restart;
> with(plots): with(plottools):
> RisEpur:=proc(Rm:Matrix,Txt,z)
> local f,clb,clr,k,x1,x2,y1,y2,sn,maxL,cs,d,mm1,r1,
> r2,zn,opc,strih,stm,n0,Qs,Q1,EpM,mm,op1,s0,Масштаб;
> global L:
> zn:=2*z-1::
> clb:=Vector(m,(j) -> j mod 2):#Цвета эюры
> clr:=Vector(m,(j) -> (j mod 3)/2.):
> maxL:=max(op(convert(L,list))):
> s0:=maxL/40.:#Плотность штрихов на эюре
> f:=seq(abs(Qx[k]+Qy[k]))*L[k]^2/8,k=1..m):
> r1:=convert(Rm,vector):
> r2:=convert(map(abs,r1),list):
  Масштаб:= (+) — архитектура и строят., (-) — машиностр.
> Масштаб:=-3*max(op(r2),f)/maxL:
> if Масштаб<>0 then
> for k to m do
>   x1:=x[str[k,1]]: x2:=x[str[k,2]]:
>   y1:=y[str[k,1]]: y2:=y[str[k,2]]:
>   sn:=L2[k]/L[k]:cs:=L1[k]/L[k]:
>   n0:=L[k]/s0:
>   opc:=THICKNESS(1),COLOR(RGB,clr[k],0,clb[k]):
>   strih:=proc(t)
>     d:=s0*t;
>     mm1:=(-(zn*Rm[k,2]+Rm[k,1])/L[k]*d+Rm[k,1])
>
>
>     Qs:=(Qx[k]+Qy[k])*sign(-L2[k])*sign(L1[k]):
>     mm:=mm1+z*Qs/2*d*(L[k]-d)/Масштаб:
>     PLOT(CURVES([[x1+d*cs,y1+d*sn,
>
>
>     [x1+d*cs-mm*sn,y1+d*sn+mm*cs]]),opc):
```



```

> end proc:
> stm:=seq(strih(t),t=0..n0):
> EpM[k]:=PLOT(CURVES([[x1-Rm[k,1]/Масштаб*sn,
>   y1+Rm[k,1]/Масштаб*cs],[x2+zn*Rm[k,2]/Масштаб*sn,
>   y2-zn*Rm[k,2]/Масштаб*cs]])),stm:
> od:#k
> op1:=axes=NONE,scaling=UNCONSTRAINED:
> Эпюра||Txt:=display(seq(EpM[i],i=1..m),
>   рама,op1,title=Txt):
> fi;#Масштаб<>0
> end proc:
  Запись процедур на диск
> save RisEpur, "C:\\Riseपुरa.m";

```

7.25.5. Определение реакций стандартным методом

Рассмотрим еще раз задачу 64 на с. 191. Если не пользоваться программой 41, то систему уравнений (4.60)–(4.63), с. 192, составленную вручную, лучше решать в системе **Maple**.

Воспользуемся простой программой 44, использующей стандартный оператор **solve**. Если значения, полученные в результате решения, требуется где-либо использовать, например, для проверки равновесия рамы в целом, то следует использовать такую конструкцию **S:=solve(...): assign(S)**, иначе оператор **solve**, решив систему, не присвоит полученные значения искомым неизвестным.

Уравнения системы можно обозначать иначе, без использования индекса, например **eq1, eq2,...,eq12**, тогда оператор записи последовательности примет вид **seq(eq||i,i=1..12)**. Здесь использована удобная операция "приклеивания" номера **||**. Возможно использование и русских слов, например **Уравнение1**.

Программа 44 (P44.mws)

```

> restart;# Задача 64, с. 191. Уравнения равновесия
> q1:=5: q2:=4: P:=10:
> Pq1:=q1*4: Pq2:=q2*5:
> eq[1]:= Xa+Xc+Xe:
> eq[2]:= Ya+Yc+Ye:
> eq[3]:= -10*Xe-5*Xc+4*Yc:
> eq[4]:= -Xe+Xd1:
> eq[5]:= -Ye+Yd1:

```

```

> eq[6] := 8*Yd1:
> eq[7] := -Xc+Xd2-Pq2:
> eq[8] := -Yc+Yd2+P:
> eq[9] := -5*Xd2+4*Yd2+2.5*Pq2:
> eq[10] := -Xd1-Xd2+Xb:
> eq[11] := -Yd1-Yd2+Yb+P+Pq1:
> eq[12] := 8*Yb+6*Pq1+4*P:
> solve({seq(eq[i], i=1..12)}),
>       {Xa, Xb, Xc, Xe, Xd1, Xd2, Ya, Yb, Yc, Ye, Yd1, Yd2});

```

7.26. Разрезная балка

7.26.1. Ввод данных

Балку можно условно рассматривать как частный случай рамы, в которой все стержни расположены на одной прямой, а нагрузки вертикальные. Программа состоит из основной части (программа 45), где исходные данные считываются из файла с учетом специфики задачи, и нескольких подпрограмм для расчета (Epuram, с. 440; RQ, с. 442; Lxu, с. 442) и отображения результатов (PrintRez, с. 442, RisEpur, с. 444), размещенных в программах 42 и 43 и вызываемых из основной оператором `read`.

Рассмотрим решение задачи 62 на с. 186. Нумерация узлов и участков приведена на рис. 438. Номера участков выделены полужирным шрифтом¹.

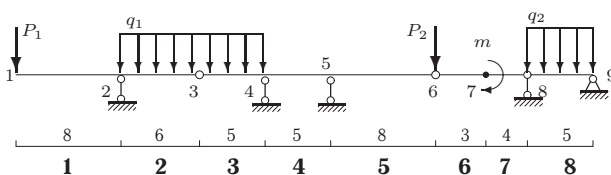


Рис. 438

Опишем файл данных. В общем он аналогичен файлу для рамы, но, конечно, проще, как проще и сама схема многопролетной балки. Файл P62.txt (слева в рамке) имеет вид

¹Участок 7 поделен точкой приложения момента не посередине, как сказано в условии. Это сделано, чтобы продемонстрировать, что реакции опор не зависят от точки приложения момента в пределах одного участка, меняется только вид эпюры моментов.

8, 6, 5, 5, 8, 3, 4, 5	Длины участков a
6, 0, 0, 0, 0, 5, 0, 0, 0	Силы
0, 0, 0, 0, 0, 0, -35, 0, 0	Моменты
2, 5, 7	Номера участков с шарниром в конце
2, 4, 5, 8, 9	Опоры
0, 2, 2, 0, 0, 0, 0, 4	Распределенная нагрузка

Рис. 439. Файл данных для программы 45

Первая строка — длины участков. Так как в этом случае нумерация узлов простая — слева направо, координаты легко вычисляются в цикле через длины $a[i]$, $i = 1, \dots, n$. Координаты y вводить не требуется, но оператор $y := \text{Vector}(n)$, обнуляющий координаты y , в программе есть. Вторая строка — силы, приложенные к узлам. В балках эти силы вертикальные и, как правило, направлены вниз. Поэтому правило знаков не как в раме — положительные нагрузки направлены вниз. Третья строка — моменты. Положительные моменты вращают против часовой стрелки. В первой строке на одно значение меньше, чем во второй и третьей. При вводе надо тщательно следить за числом вводимых значений.

Четвертая строка в данном примере состоит всего из трех чисел — это номера участков, в конце которых есть промежуточный шарнир. Всего в балке три таких шарнира: 3, 6, 8. Шарнир 8 одновременно и опорный, и промежуточный. Узлы 2, 4, 5, и тем более, 9 являются опорами.

Отдельные части многопролетной балки могут соединяться только шарнирами¹, поэтому в списке соединений Vns все элементы имеют вид $[N, 2, 3]$, где N — номер участка, в конце которого расположен шарнир. Пятая строка — номера опор. В задаче нет горизонтальных нагрузок и горизонтальных реакций опор, однако одну опору (все равно какую) с неподвижным шарниром указать надо. При вводе это не требуется, а в программе в списке $OPOR$ первая скобка имеет вид $[Nopor[-1], 1]$. В **Maple** запись $Nopor[-1]$ означает обращение к последнему элементу списка, в данном примере это опора 9. Согласно правилу, принятому в программе 41, горизонтальная опора кодируется единицей, поэтому второй элемент в этой скобке равен 1.

Шестая строка — распределенные нагрузки по участкам. Если нет нагрузки — ставим 0, нагрузки направлены вниз, при вводе положительные. Знак минус приписывается уже в программе: $Qy := [-\text{parse}(\text{readline}(\text{fd}))]$.

¹Теоретически возможна скользящая заделка, соединяющая части, но практически такие сочленения не используются.

7.26.2. Программа

Как и при расчете рамы, подпрограмма $E_puraM(P1, P2, P3)$ определяет моменты в балке, а подпрограмма $RQ(Rx, Ry)$ — перерезывающие силы.

Программа 45 (P45.mws)

```
> restart:# Разрезная балка.
> read "C:\\Podpr1.m":#Линия, программа 9, с. 356
> read "C:\\epura.m"; #PrintRez, RQ, EpuraM, с. 440
> read "C:\\Riseपुरa.m"; #Riseपुर, с. 444
> fd :=fopen("C:/M13/p62.txt",READ):#Открываем файл
    Ввод данных
> a:=[parse(readline(fd))]:# Длины участков
> n:=nops(a)+1:          # Число узлов
> P1:=Vector(n):
> P2:=[-parse(readline(fd))]:# Нагрузки в узлах
> P3:=[parse(readline(fd))]: # Моменты в узлах
> m:=n-1:              # Число стержней
> s1:=parse(readline(fd));# Стержни с шарн. в конце
> Nopor:=parse(readline(fd));# Опоры
> nops([Nopor]);      # Число опор
> Qx:=Vector(m):
> Qy:=[-parse(readline(fd))];# Нагрузки распредел.
> fclose(fd):        # Закрываем файл
    Координаты узлов
> x[1]:=0:for i to n-1 do x[i+1]:=x[i]+a[i];od:
> y:=Vector(n):
    Все внутренние соединения — шарниры [* , 2, 3]
> Vns:=[seq([s1[i],2,3],i=1..nops([s1]))]:
> ks:=nops(Vns):      #Число внутренних шарниров
> Vns1:=Vector(n):#Нулевой вект. внутр. мом. в шарн.
    Опоры: [Узел, направление (1-X, 2-Y, 3-M)]
> OPOR:=[[Nopor[1],1],
>         seq([Nopor[i],2],i=1..nops([Nopor]))];
> ko:=nops(OPOR):    # Число опор
> printf("Число неизвестных %d + %d, уравнений
>         %d + %d +%d", ko, 6*m, ks, 3*m, 3*n);
    Число неизвестных 6 + 48, уравнений 3 + 24 + 27
```

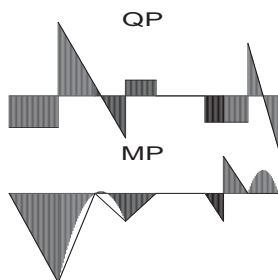


Решение

```
> for i to n do str[i,1]:=i;str[i,2]:=i+1;od:
> Lxy(): RisRama():
> ЕpurаM(P1,P2,P3):# Моменты
> Реакции=seq(Ro[i],i=1..ko);
      Реакции = (0, 20, 11, -3, 15, 10)
> RQ(Rx,Ry):      # Поперечная сила
> PrintRez(Q1,"Qp");
```

N	Qp(1)	Qp(2)
1	-6.00	-6.00
2	14.00	2.00
3	2.00	-8.00
4	3.00	3.00
5	0.00	0.00
6	-5.00	-5.00
7	-5.00	-5.00
8	10.00	-10.00

```
> QP:=-Matrix(m,2,Q1):
> RisEpur(QP,'QP',0);
> MP:=Matrix(m,2,Rm):
> RisEpur(MP,'MP',1);
```



```
> PrintRez(Rm,"Mp");
```

N	Mp(1)	Mp(2)
1	0.00	-48.00
2	48.00	0.00
3	0.00	-15.00
4	15.00	0.00
5	0.00	0.00
6	0.00	-15.00
7	-20.00	0.00
8	0.00	0.00

7.26.3. Балка на двух опорах

Обычная балка на двух опорах также может быть рассчитана по этой программе. Рассмотрим задачу 61 на с. 182. Ввод, конечно, в этом случае короче, номера стержней с шарнирами вводить не требуется, поэтому четвертая строка пустая (файл r61.txt):

4,2,4	Длины участков
0, -5, 0, 0	Сосредоточенные нагрузки
0, 0, -8, 0	Моменты
1,4	Опоры
4, 0, 0	Распределенная нагрузка

Рис. 440. Файл данных для программы 45

Приведем результаты счета. Значения на эпюрах выводятся по строкам. Номер строки соответствует номеру участка, начало участка $Qp(1)$, конец $Qp(2)$. Аналогично записаны и моменты.

N	Qp(1)	Qp(2)	N	Mp(1)	Mp(2)
1	9.00	-7.00	1	0.00	4.00
2	-2.00	-2.00	2	-4.00	0.00
3	-2.00	-2.00	3	-8.00	0.00

Моменты и перерезывающие силы не отличаются от рис. 256, с. 185. Эпюры моментов и перерезывающих сил имеют вид

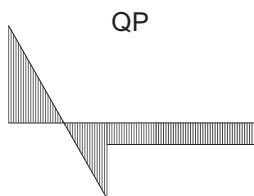


Рис. 441

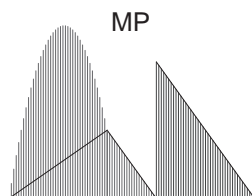


Рис. 442

7.27. Линии влияния в разрезной балке. *Маллет*

Линии влияния реакций опор строим в программе кинематическим способом (задача 63, с. 188). Пусть заданная балка длиной L состоит из n отдельных балок, закрепленных на $n+1$ вертикальной опоре. Одна из опор неподвижная. Так как горизонтальных нагрузок по условию нет, то и горизонтальных реакций не возникает. Поэтому не существенно, какая из опор является неподвижной.

Линия влияния совпадает с изогнутой формой балки, если освободить ее от опоры, линия влияния которой строится. Линия получается ломаной с изломами в промежуточных шарнирах.

Уравнение оси каждой из n частей балок в отклоненном от горизонтали положении представим линейной функцией $f_k(x) = a_k x + b_k$, $k = 1, \dots, n$. Для определения $2n$ неизвестных коэффициентов составляем систему уравнений. Из условия неразрывности на шарнирах имеем уравнения $f_k(x_{k+1}) = f_{k+1}(x_{k+1})$, где $k = 1, \dots, n - 1$. Начало координат примем на левом конце балки, $x_1 = 0$. Координата правого конца $x_{n+1} = L$. Остальные уравнения системы $f_i(x) = d_i$ задают значения функций на опорах $i = 1, \dots, n + 1$. Для построения линии влияния реакции опоры j полагаем $d_j = 1$, на остальных $d_i = 0$, $i \neq j$. Решаем систему уравнений (оператор `solve`), составляем кусочно-линейную функцию $L_V = LV(x)$, представляющую собой искомую линию влияния. Площадь ω , ограниченную линией влияния, вычисляем интегрированием. Для того чтобы вычислить положительную часть площади, используем тождество $\omega^+ = \omega - \omega^-$, где

$$\omega^- = 0.5 \int_0^L LV(x) - |LV(x)| dx.$$

Для вычислений по программе требуется ввести координаты опор `хо`, координаты сочленяющих шарниров `sr`, распределение опор по балкам `Н`. Последний список состоит из 0, 1 или 2. Причем, если в списке есть два значения 2, то между ними обязательно должен быть нуль. Не может быть два нуля подряд, а также нуль в начале или на конце списка.

Программу оформим в виде маплета. Его структура отличается от аналогичных программ (программы 6, 30, 40) и имеет вид

```

InfLin:=proc()
  use Maplets[Tools] in
    Ввод данных оператором Get
  end use:
  Вычисление Vmax, Vmin
  Построение графика GR

  Vmax, Vmin, GR;
end proc;
with(Maplets[Elements]):
  Определение элементов интерфейса (с, mx, mn и др)
mplt:= Maplet(Window(, [[, [[, , , , , ], ], [, ], [, , , , , ]]])):
  Maplets[Display](mplt):

```

В структуре не отображены элементы, а указаны только запятые, их разделяющие. Интерфейс приведен на рисунке 443. Последними в процедуре `InfLin` приведены `Vmax`, `Vmin`, `GR`, поэтому для вызова из процедуры соответствующих результатов надо указать порядковый номер: `InfLin()[1]`, `InfLin()[2]`, `InfLin()[3]`. Кнопка "max" служит для определения максимального значения реакции опоры от совместного действия временной и постоянной нагрузки, кнопка "min" — минимального значения. Обе эти кнопки можно объединить в одну. Нажатие на нее даст сразу оба значения, каждое значение — в своем окне. Для этого можно обозначить элементы `Evaluate` как `E1` и `E2`, а потом объединить их функционально скобкой `Action`:

```
> E1:=Evaluate('Tmx' = "InfLin()[1]");
> E2:=Evaluate('Tmn' = "InfLin()[2]");
> bmx:=Button("Run",Action(E1,E2));
```

Можно, конечно, не вводить дополнительные обозначения `E1` и `E2`, а писать эти элементы непосредственно в `Action`, но длинные строки с множеством открывающих и закрывающих скобок плохо читаются.

Элемент `Evaluate('PL1' = 'display(InfLin()[3])'` обозначить целиком за новую переменную, например `E3`, а потом присоединить к `E1` и `E2` не удастся. В лучшем случае можно обозначить `E3:=display(InfLin()[3])'`.

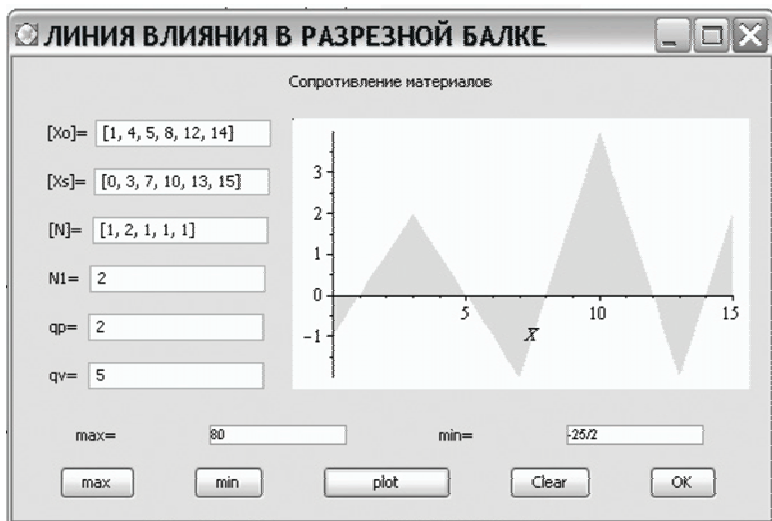


Рис. 443. P46.maplet

В программу введены данные задачи 63 на с. 188.

Для обозначения отрицательной площади использовано слово `mins`, так как слово `minus` означает бинарный инфиксный оператор, используемый для вычитания множеств. Суммарную площадь, ограниченную линией влияния, обозначаем `omega[]`, а не просто `omega`, для того чтобы переменная с индексом и одноименная переменная без индекса не вступали в противоречие после операции присвоения.

Программа 46 (P46.mws)

```
> restart; # Маплет. Линии влияния в разрезной балке
> with(plots):
> InfLin:=proc() local j, omega, k, n, n1, qf, d, f, eq,
> q, r, qv, qp, xo, N, N1, sr, i, LV2, LV, GR, Vp, Vmax,
> Vmin, VvrMax, VvrMin, a, b, c, plt, mx, mn, so1, so2, Ev:
> use Maplelets[Tools] in
> N1:=Get('TN1'::posint);# Исследуемая опора
> xo:=Get('Txo'::list); # Координаты опор
> sr:=Get('Tsr'::list); # Координаты шарниров
> qp:=Get('Tqp'::algebraic): # Значение распр. нагр.
> qv:=Get('Tqv'::algebraic):
> N:=Get('TN'::list);# Число опор на балку
> end use:
> n:=nops(N): # Число балок
> n1:=n+1: # Число опор
> d:=[0$n1]: # Все смещения = 0, кроме № N1
> d[N1]:=1: # Единичное смещение на опоре N1
> f:=proc(x,i) a[i]*x+b[i] end:
> for i to n-1 do # Усл.непрерывн. на шарнирах
> eq[i]:=f(sr[i+1],i)-f(sr[i+1],i+1):
> od:
> k:=0:
> for i to n do
> for j to N[i] do
> k:=k+1:
> eq[k+n-1]:=f(xo[k],i)=d[k];
> od;
> od;
> r:=solve({seq(eq[i],i=1..2*n)},
```

```

>      {seq(op({a[i],b[i]}),i=1..n)}): assign(r);
> LV:=piecewise(seq(op([sr[i]<X and
>      X<sr[i+1],f(X,i)]),i=1..n)):
> GR:=plot(LV,X=0..sr[n1],filled=true,color=green):
> omega[]:=int(LV,X=0..sr[n1]);
> LV2:=piecewise(seq(op([sr[i]<X and
>      X<sr[i+1],abs(f(X,i))]),i=1..n)):
> omega[mins]:=int((LV-LV2)/2,X=0..sr[n1]);
> omega[plus]:=omega[]-omega[mins];
> Vp:=qp*omega[]:      # Усилие от пост. нагр.
> VvrMin:=omega[mins]*qv:# Усилие от врем. нагр.
> VvrMax:=omega[plus]*qv:
> Vmax:=Vp+VvrMax;      # Максим. нагрузка
> Vmin:=Vp+VvrMin;      # Миним. нагрузка
> Vmax,Vmin,GR;
> end proc:#InfLin

> Xo:=[1,4,5,8,12,14]:
> Xs:=[0,3,7,10,13,15]:
> Tn:=[1,2,1,1,1]:
> a1:='editable' = 'false',1..8:
> with(Maplets[Elements]):
> c[1]:=["[Xo]=",TextField['Txo'](14,'value'=Xo)]:
> c[2]:=["[Xs]=",TextField['Tsr'](14,'value'=Xs)]:
> c[3]:=["[N]=", TextField['TN'](14,'value'=Tn)]:
> c[4]:=["[N1]=", TextField['TN1'](14,'value'=2)]:
> c[5]:=["qp=", TextField['Tqp'](14,'value'=2)]:
> c[6]:=["qv=", TextField['Tqv'](14,'value'=5)]:
> plt:=Plotter['PL1']('height=190'):
> mx:="max=",TextBox['Tmx'](a1):
> mn:="min=",TextBox['Tmn'](a1):
> bmx:=Button("max",Evaluate('Tmx'="InfLin()[1]")):
> bmn:=Button("min",Evaluate('Tmn'="InfLin()[2]")):
> sol:=SetOption('Tqv'=" "):
> so2:=SetOption('Tqp'=" "):
> bcl:=Button("Clear",Action(sol,so2)):
> bsd:=Button("OK",

```

```

> Shutdown(['Tsr', 'Tmx', 'Tmn', 'PL1']):
> mplt:= Maplet(Window(
> 'title'="ЛИНИЯ ВЛИЯНИЯ В РАЗРЕЗНОЙ БАЛКЕ", [
> "Сопротивление материалов",
> [[seq(c[i], i=1..6)], plt], [mx, mn],
> [bmx, bmn, Button('width=90',
> "plot", Evaluate('PL1' = 'display(InfLin())[3]')]),
> bcl, bsd] ])):
> Maplets[Display](mplt):

```

7.28. Статически неопределимая рама

7.28.1. Ввод данных. Примеры

Программа 47 расчета статически неопределимой рамы основана на программе 41, с. 433. В начале программы считываются файлы `epura.m`, `Risepura.m` и `Podpr1.m`, содержащие необходимые подпрограммы. Затем вводятся данные об основной системе рамы, производится счет и вывод результатов — эпюр и таблиц к эпюрам.

Пример 1

Рассмотрим задачу 65, с. 196. Опишем файл данных для расчета рамы (файл `r65.txt`). Пронумеруем узлы: A — № 1, C — № 2, D — № 3, B — № 4. Начало координат поместим в заделку A .

1)	0, 0, 4, 6	(Координаты x)
2)	0, 3, 3, 3	(Координаты y)
3)	0, 0, 0, 0	(Горизонтальная нагрузка $P_x=P1$)
4)	0, 0, -6, 0	(Вертикальная нагрузка $P_y=P2$)
5)	0, 0, 0, 0	(Моменты в узлах $M=P3$)
6)	[1,2], [2,3], [3,4]	(Стержни [начало, конец])
7)		(Внутренние соединения)
8)		(Внутренние неизвестные)
9)	[1,1], [1,2], [1,3]	(Реакции опор [узел, напр.])
10)	8, 0, 0	(Горизонтальная распр. нагр. $q_x=Qx$)
11)	0, 0, 0	(Вертикальная распр. нагр. $q_y=Qy$)
12)	1, 1, 1	(Жесткость стержней EJ)
13)	[4,1], [4,2]	(Внешние неизвестные)

Рис. 444. Файл данных для программы 47

Ввод данных почти не отличается от программы 41. Строки 1, 2 – ввод координат узлов. Следующие три строки содержат информацию об узловых нагрузках. В шестой строке указаны три стержня (или участка) рамы, в квадратных скобках записаны номера узлов в начале и конце стержня. Ориентация стержня произвольная и на ответы не влияет. Затем идет строка, в которой указываются внутренние соединения в раме, если они есть. Здесь таковых нет, и строка оставляется пустой. Восьмая строка описывает неизвестные реакции метода сил, введенные в местах внутренних соединений. В этой задаче все неизвестные являются внешними силами, поэтому строка остается пустой. Опорные узлы описаны в девятой строке. Первое число в скобке – это номер узла, второе – направление реакции. Как и в статически определимой раме горизонтальное направление кодируется как 1, вертикальное – 2, момент – 3. В этой задаче в узле 1 заделка, которая соответствует трем скобкам. Для неподвижного шарнира надо писать две скобки, для подвижного – одну. Последняя строка содержит информацию о внешних неизвестных метода сил. Принцип тот же: сначала в скобке записан номер узла, потом направление силы (горизонтальной – 1, вертикальной – 2) или момента (3). Эта строка будет пустой (надо поставить в ней хотя бы один пробел), если основная система получается врезанием шарнира или разрезом рамы.

Пример 2

Рассмотрим задачу 10.5 из книги [14], с. 313. К раме, имеющей заделки на опорах, приложена распределенная нагрузка $q = 1$ кН/м и сосредоточенная $P = 2$ кН, размеры даны в метрах (рис. 445, *a*). Очевидно, система трижды статически неопределимая. В такой задаче можно выбрать различные основные системы. Например, в качестве основной системы можно взять трехшарнирную раму (два опорных шарнира и один в середине), которая статически определима (см. аналогичную задачу 4, с. 16). Неизвестными метода сил будут в этом случае три момента.

Выбираем другую основную систему, выполняя разрез в центре ригеля¹. Неизвестными метода сил здесь являются возникающие при этом реакции – горизонтальная сила X_1 , вертикальная X_2 и момент X_3 (рис. 445, *b*). Эти реакции приложены как к левой части рамы, так и к правой, в противоположные стороны. Рассмотрим файл данных `SorgM.txt` (рис. 445). Начало координат находится в левой заделке. В первых двух строках – координаты концов участков рамы. Три следующие строки – сосредоточенные нагрузки (силы и моменты).

¹Горизонтальный несущий стержень. Его жесткость EJ обычно больше жесткости вертикальных стоек, но в этом примере все жесткости одинаковые.

В строке 8 цифры 1, 2, 3 означают, что в соединениях 1, 2 и 3, указанных в строке 7, прикладываются единичные силы по направлениям, заданным третьей цифрой в скобке. В данном случае это две силы и момент.

1)	0,	0,	1,	2,	2	(x)
2)	0,	1,	1,	1,	0	(y)
3)	0,	0,	0,	0,	0	(P_x)
4)	0,	0,	2,	0,	0	(P_y)
5)	0,	0,	0,	0,	0	(M)
6)	[1,2],	[2,3],	[3,4],	[4,5]	(стержни)	
7)	[2,2,1],	[2,2,2],	[2,2,3]	(шарнир в 3)		
8)	1,	2,	3	(X ₁ , X ₂ , X ₃)		
9)	[1,1],	[1,2],	[1,3],	[5,1],	[5,2],	[5,3]
10)	0,	0,	0,	0	(q _x)	
11)	0,	-1,	-1,	0	(q _y)	
12)	1,	1,	1,	1	(EJ)	
13)						

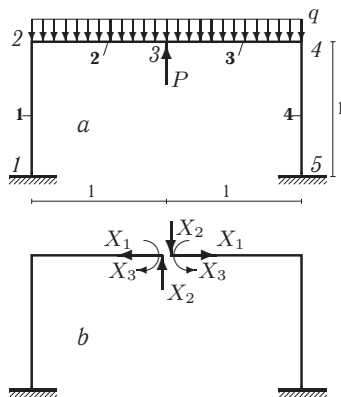


Рис. 445. Файл данных для программы 47

Строка 13 здесь пустая — внешних неизвестных в задаче нет.

В такой задаче есть еще один вариант основной системы — консольно закрепленная на одной опоре (безразлично какой) с тремя неизвестными на другой опоре. Если заделка остается на левой опоре, то в этом случае строка 13 имеет вид [5, 1], [5, 2], [5, 3], а строки 7 и 8 будут пустыми.

Для расчета рамы запускается программа 47. Вместо файла r65.txt в операторе `forin` необходимо вызвать файл `SortM.txt`.

Пример 3

Рассмотрим задачу из книги [49], с. 234. К раме, имеющей шарниры на опорах, приложена распределенная нагрузка $q = 10$ кН/м, размеры даны в метрах (рис. 446, а). Жесткость ригеля в два раза больше жесткости стоек. Горизонтальная стяжка закреплена на стойках шарнирно.

Система дважды статически неопределимая. Выбираем основную систему (рис. 446, б), выполняя разрез стяжки (внутренняя неизвестная X_1) и заменяя правую опору горизонтально подвижным шарниром (внешняя неизвестная X_2). Файл данных для программы 47 имеет вид (файл `StrMech.txt`):

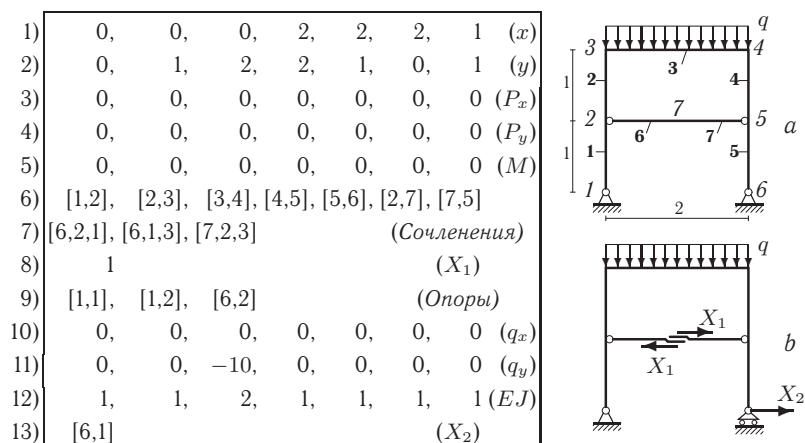


Рис. 446. Файл данных для программы 47

Комментарии в конце строк в скобках в файл не включаются. На рисунке силы X_1 смещены для наглядности по вертикали, в действительности же они действуют по одной прямой вдоль стяжки. В третьей позиции строки 12 стоит 2. Это отражает тот факт, что ригель (третий стержень) вдвое жестче стоек.

Пример 4

Рассмотрим пример 9.3 из книги [55], с. 231. Рама имеет заделку в одной опоре и шарнир — в другой. Приложена распределенная нагрузка $q = 22$ кН/м и вертикальная сила $F = 16$ кН, размеры даны в метрах (рис. 447, а). Жесткость ригеля равна EJ , жесткость стоек — kEJ , $k = 0,7$.

Система дважды статически неопределимая. Выбираем основную систему (рис. 447, б), отбрасывая шарнирную опору. Файл данных (SoprM-S.txt) для программы 47 приведен на рис. 447.

7.28.2. Работа программы

Построение эпюр в основной системе от внешних нагрузок производится с помощью подпрограммы `EpurM(P1, P2, P3)`. Здесь программа ничем не отличается от программы 41. В цикле `for j to n do` поочередно прикладываются единичные усилия на опорах (если таковые есть, т. е. $po \neq 0$). Внешние нагрузки полагаются равными нулю (значения распределенных нагрузок сохранены заранее в переменных $Qx0$ и $Qy0$ — они потребуются для построения эпюр). Вычисление

эпюр моментов от единичных сил производится также в подпрограмме

1)	0,	0,	5,	5,	6.25 (x)
2)	0,	4,	4,	0,	4 (y)
3)	0,	0,	0,	0,	0 (P_x)
4)	0,	0,	0,	0,	-16 (P_y)
5)	0,	0,	0,	0,	0 (M)
6)	[1,2],	[2,3],	[3,4],	[3,5]	(Стержни)
7)					
8)					
9)	[1,1],	[1,2],	[1,3]		(Опоры)
10)	22,	0,	0,	0	(q_x)
11)	0,	0,	0,	0	(q_y)
12)	0.7,	1,	0.7,	1	(EJ)
13)	[4,1],	[4,2]			(X_1, X_2)

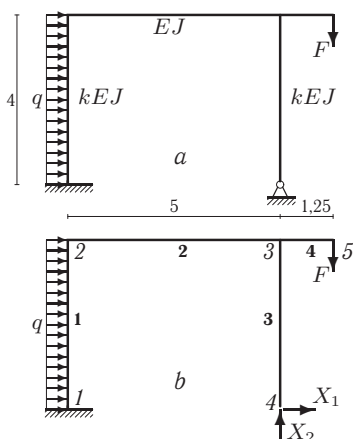


Рис. 447. Файл данных для программы 47

`Ерама(P1,P2,P3)`. В другом цикле `for j to nv do`, похожем на предыдущий, единичные усилия или моменты прикладываются к внутренним соединениям (если $nv \neq 0$). Результаты, полученные в этих циклах, выводятся все вместе в цикле `for j to nn do` по всем неизвестным метода сил. Построенные эпюры можно было бы вывести внутри циклов сразу после их получения, для чего надо только после окончания цикла `od` поставить не двоеточие, а точку с запятой, однако при этом на экран будет выведено много ненужной информации, поэтому эпюры были сохранены в переменных `Эпюраm1`, `Эпюраm2` и `Эпюраq1`, `Эпюраq2`, а затем уже выведены вместе с таблицами результатов в одном цикле.

В двойном цикле по i и j получаем коэффициенты Δ_{iP} , δ_{ij} системы канонических уравнений. Использована процедура `EM` из файла `ерама.m` (с. 442) преобразования матрицы значений моментов по концам участка в функцию.

Решение получаем с помощью обратной матрицы δ^{-1} и заносим в вектор X . Допустима и другая форма решения $X:=1/\delta.\Delta$. Заметим, что обе эти формы нахождения обратной матрицы не требуют подгружения пакета линейной алгебры `with(LinearAlgebra)`, в котором есть другой оператор определения обратной матрицы `MatrixInverse`. В старом пакете, применявшемся в ранних версиях `Maple`, и сохраненном в новых версиях только для совместимости программ `linalg`, этот оператор называется `inverse`.

Решение системы содержит избыточное число знаков после запятой. Изменить форму вывода результатов можно оператором `interface(displayprecision=3)`. Обращение к нему уменьшит число знаков после запятой до трех.

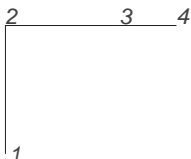
Программа 47 (P47.mws)

```
> restart: #Расчет статически определимой рамы
> read "C:\\epura.m"; # Программа 42 с. 440
> read "C:\\Riseपुरa.m"; # 43 с. 444
> read "C:\\Podpr1.m"; # Программа 9 с. 356
  Ввод данных
> fd := fopen("C:/M13/p65.txt",READ):# Открываем файл
> x:=[parse(readline(fd))]:# x
> y:=[parse(readline(fd))]:# y
> n:=nops(x): # Число узлов
> P1:=[parse(readline(fd))]:# Нагрузки в узлах Px
> P2:=[parse(readline(fd))]:# Нагрузки в узлах Py
> P3:=[parse(readline(fd))]:# Нагрузки в узлах M
> str:=[parse(readline(fd))]:# Стержни. Номера узлов
> m:=nops(str): # Число стержней
> Vns:=[parse(readline(fd))]:
> U:=[parse(readline(fd))]: # Из множества{1,...,ks}
> OPOR:=[parse(readline(fd))]:# Опоры
> Qx:=[parse(readline(fd))]:# Распределенная нагрузка
> Qy:=[parse(readline(fd))]:
> EJ:=[parse(readline(fd))]:# Относительная жесткость
> Nms:=[parse(readline(fd))]:# Неизвестные метода сил
> fclose(fd): # Закрываем файл
> ks:=nops(Vns): # Число внутренних шарниров
> Vns1:=Vector(n):
> nv:=nops(U): # Число внутренних неизвестных
> ko:=nops(OPOR): # Число опор
> printf("Число неизв. % d +%d, уравн. %d + %d + %d",
> # ко, 6*m, ks, 3*m, 3*n);
> no:=nops(Nms): # Число внешних неизвестных
> nn:=no+nv: # Общее число неизвестных
Число неизв. 3 + 18, уравн. 0 + 9 + 12
```


Решение

```
> Lxy(): # Определяем длины стержней
```

```
> RisRama();
```



```
> EпuraM(P1,P2,P3):#Моменты в основной системе
```

```
> Реакции=seq(Ro[i],i=1..ko);
    Реакции = (- 24, 6, 60)
```

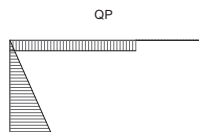
```
> RQ(Rx,Ry): #Поперечная сила в осн. системе
```

```
> PrintRez(Q1,"Qp");
```

N	Qp(1)	Qp(2)
1	24.00	0.00
2	6.00	6.00
3	0.00	0.00

```
> QP:=Matrix(m,2,Q1):
```

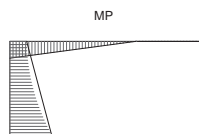
```
> RisEpur(QP,'QP',0);
```



```
> MP:=Matrix(m,2,Rm):
```

```
> RisEpur(MP,'MP',1); PrintRez(Rm,"Mp");
```

```
> Qx0:=Qx: Qy0:=Qy:#Сохраняем данные (распр. нагр)
```



N	Mp(1)	Mp(2)
1	60.00	-24.00
2	24.00	0.00
3	0.00	0.00

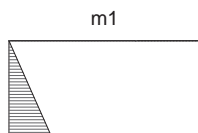
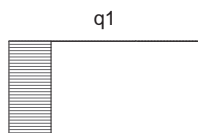
```

> for j to no do # Цикл по внешним неизвестным
    Обнуляем нагрузки
> P1:=Vector(n): P2:=Vector(n): P3:=Vector(n):
> Qx:=Vector(m): Qy:=Vector(m):
> z:=Nms[j, 2]:
    Прикладываем единичную нагрузку
> P||z[Nms[j, 1]]:=1:
> EpuraM(P1, P2, P3):
> m||j:=Matrix(m, 2, Rm):
> RisEpur(m||j, cat("m", j), 1);
> RQ(Rx, Ry):
> q||j:=Matrix(m, 2, Q1):
> RisEpur(q||j, cat("q", j), 0);
> od:
> for j to nv do # Цикл по внутренним неизвестным
    Обнуляем нагрузки
> P1:=Vector(n): P2:=Vector(n): P3:=Vector(n):
> Qx:=Vector(m): Qy:=Vector(m):
> Vns1:=Vector(n):
    Прикладываем единичную нагрузку
> Vns1[U[j]]:=1;
> EpuraM(P1, P2, P3): j1:=j+no:
> m||j1:=Matrix(m, 2, Rm):
> RisEpur(m||j1, cat("m", j1), 1); # Создаем Эпюрам
> RQ(Rx, Ry):
> q||j1:=Matrix(m, 2, Q1):
> RisEpur(q||j1, cat("q", j1), 0); # Создаем Эпюраq
> od:
> for j to nn do
> PrintRez(m||j, cat("m", j));
> PrintRez(q||j, cat("q", j));
> Эпюраq||j;
> Эпюрам||j;
> od;

```

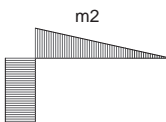
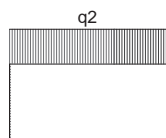
N	m1(1)	m1(2)
1	3.00	0.00
2	0.00	0.00
3	0.00	0.00

N	q1(1)	q1(2)
1	1.00	1.00
2	0.00	0.00
3	0.00	0.00



N	m2(1)	m2(2)
1	-6.00	6.00
2	-6.00	2.00
3	-2.00	0.00

N	q2(1)	q2(2)
1	0.00	0.00
2	-1.00	-1.00
3	-1.00	-1.00



```

> Deltap:=Vector(nn): delta:=Matrix(nn):
> for i to nn do# Коэффициенты канонических уравнений
>   Deltap[i]:=evalf(add(int(EM(MP,1,u,k)*
>     EM(m||i,0,u,k)/EJ[k],u=0..L[k]),k=1..m));
>   for j to nn do
>     delta[i,j]:=evalf(add(int(EM(m||i,0,u,k)*
>       EM(m||j,0,u,k)/EJ[k],u=0..L[k]),k=1..m));

```

```

> od:#j
> od:#i
  Решаем систему канонических уравнений.
  Умножение обратной матрицы на вектор
> X:=-delta^(-1).Deltap:
> delta,X,Deltap:
  
$$\begin{bmatrix} 9. & -27. \\ -27. & 180. \end{bmatrix}, \begin{bmatrix} -11.7575757575757544 \\ 3.08080808080808044 \end{bmatrix}, \begin{bmatrix} 189. \\ -872. \end{bmatrix}$$

  Моменты в статически неопределимой раме
> MR:=add(m||i*X[i],i=1..nn)+MP:
  Поперечные силы в статически неопределимой раме
> QR:=add(q||i*X[i],i=1..nn)+QP:
> PrintRez(QR,'Q');
> Qx:=Qx0: Qy:=Qy0: #Восстанавливаем нагрузки

```

```

N      Q(1)      Q(2)
1      12.24     -11.76
2       2.92       2.92
3      -3.08     -3.08

```

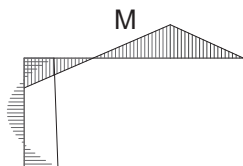
```
> PrintRez(MR,'M');
```

```

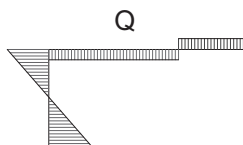
N      M(1)      M(2)
1       6.24     -5.52
2       5.52       6.16
3      -6.16       0.00

```

```
> RisEpur(MR,'M',1);
```



```
> RisEpur(QR,'Q',0);
```

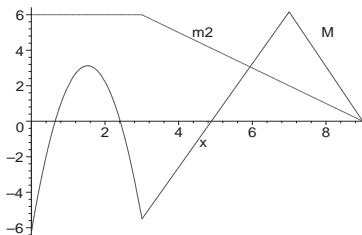


7.28.3. Кинематическая проверка

Проверку эпюры моментов M , построенной в задаче 65, удобно выполнить непосредственным интегрированием произведения эпюр M и m_2 . Произведение должно быть равно нулю. По значениям эпюр моментов на концах строим функциональные зависимости либо непосредственно задавая параболический вид кривой, либо получая кусочно-линейную зависимость с помощью оператора `spline`. Чтобы выделить из кривой нужный участок, используем оператор `piecewise`. Для наглядности раму "распрямляем" в точке C , располагая эпюру вдоль прямой: $0 \leq x \leq 3$ (участок AC), $3 \leq x \leq 9$ (участок CB).

Программа 48 (P48.mws)

```
> restart; #Кинематическая проверка
> q:=8: # Распределенная нагрузка
> a:=3: # Длина участка
    Криволинейная часть эпюры M на участке AC (рис. 299, с. 200)
> f11:=piecewise(x>0 and x<a, -x*(x-a)*q/2):
    Прямолинейная часть на участке AC
> f12:=spline([0,3],[-6.24,-5.52],x,linear):
    Эпюра на участке AC
> f1:=piecewise(x>0 and x<3,f12+f11):
    Участок CB
> f21:=spline([3,7,9],[-5.52,6.16,0],x,linear):
> f2:=piecewise(x>3 and x<9, f21):
    Эпюра m2 от единичной силы (рис. 296, с. 199)
> g:=spline([0,3,9],[6,6,0],x,linear):
    Эпюры m2 и M в одних осях
> plot({f1+f2,g},x=0..9);
```



```
> int(g*(f1+f2),x=0..9);# Интеграл
0.3666366667 10-5
```

7.28.4. Расчет пять раз статически неопределимой рамы

Программа позволяет рассчитывать рамы с произвольным числом статической неопределимости. Приведем ввод данных и результаты счета в пять раз статически неопределимой раме (рис. 448, а), используя метод сил. В двух опорах рама имеет жесткую заделку, в третьей — скользящую заделку. Даны нагрузки $P = 1$ кН, $q = 8$ кН/м. Жесткость всех стержней одинакова, размеры даны в метрах.

Выберем основную систему, отбросив все опоры, кроме левой заделки (рис. 448, б). Файл данных (p65a.txt) имеет вид

1)	0,	1,	2,	3,	2	(x)
2)	1,	1,	1,	1,	0	(y)
3)	0,	0,	0,	0,	0	(P_x)
4)	0,	-1,	0,	0,	0	(P_y)
5)	0,	0,	0,	0,	0	(M)
6)	[1,2],	[2,3],	[3,4],	[3,5]		(Стержни)
7)						
8)						
9)	[1,1],	[1,2],	[1,3]			(Опоры)
10)	0,	0,	0,	-8		(q_x)
11)	0,	0,	0,	0		(q_y)
12)	1,	1,	1,	1		(EJ)
13)	[5,1],	[5,2],	[5,3],	[4,2],	[4,3]	(X_i)

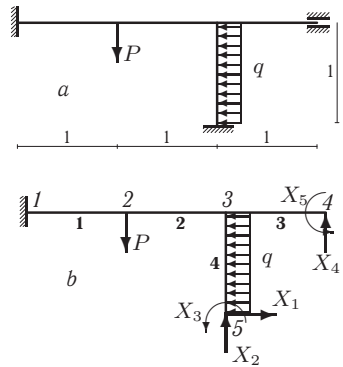


Рис. 448. Файл данных для программы 47

Так как внутренних сочленений в основной системе нет, то строки 7 и 8 пустые.

Из результатов счета по программе 47 приведем только эпюру моментов:

N	M(1)	M(2)
1	.21	.23
2	-.23	-.33
3	-.17	-.08
4	.50	-.75

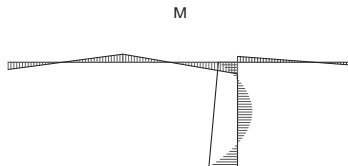


Рис. 449

Заметим, что данная рама значительно легче решается методом перемещений [49]. Потребуется найти только одну неизвестную — угол поворота узла 3.

7.29. Арка. Маплет

Маплет построен по той же схеме, что и программа 40 для расчета балки на с. 424. Организован модуль ArchМодуль из двух программ Решение и ПускArch. В первой программе вычисляются ординаты эпюр моментов и перерезывающих и нормальных сил в трехшарнирной и двухшарнирной (один раз статически неопределимой) арках и строятся соответствующие эпюры. Изображение арки с нагрузками и эпюры перечисляются через запятую в конце программы Решение для того, чтобы потом можно было обратиться к ним по номеру, например Решение () [1].

Вторая программа — организация интерфейса (рис. 450). Для каждой эпюры предназначена своя кнопка, а результат выводится в одно окно PL1, замещая предыдущее изображение.

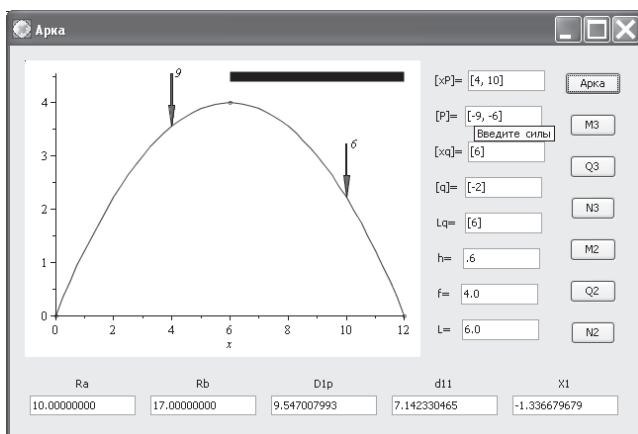


Рис. 450. P49.maplet

Использованы данные задачи 305, с. 203. Входными данными являются размеры арки и нагрузки. Программа рассчитана на произвольное число нагрузок, поэтому, даже если нагрузка одна, как q в рассматриваемом примере, эту нагрузку и ее характеристики (координата, длина) надо представлять не скаляром, а списком в квадратных скобках. Значения L , f , h предполагают вещественную форму ввода.

При вводе целого числа система выдает сообщение (Dispatch), в котором указывается о недопустимой форме ввода: *the entry 6 is not of type 'float'*.

Форма арки $y(x)$ оформлена в отдельную процедуру. Для расчета арки другой формы достаточно эту процедуру изменить. В случае круговой арки имеем

```
> y:=proc(x) local R: R:=(L^2+f^2)/(2*f):
> sqrt(R^2-x^2)+f-R:end:
```

Для цепной линии

```
> y:=proc(x) local a: a:=arccosh(2):
> (2-cosh(-a*x/L))*f: end:
```

Расчет производим по формулам плана на с. 202. Вычисление тригонометрических функций и построение эпюр записываем в процедуры. Основой расчета являются две функции для балочного момента и перерезывающей силы. Здесь существенно использована аналитическая мощь системы **Maple**. Эпюры представлены кусочно-заданными функциями, интегрирование которых для **Maple** выполняется просто. Так же просто строятся и изображения самих эпюр. Названия кнопок ("M3", "Q3" и др.) помещены в список Кнопка для того, чтобы вывести их в цикле. Введен вспомогательный оператор Ev.

Программа 49 (49.mws)

```
> restart:with(plottools):with(plots):# Маплет. Арка
> ArchМодуль:=module()
> export Решение,ПускArch:
> Решение:=proc() local величина, шарниры, m0, qA, NA,
> нагр, вид, распрнагр, L, f, y, a, F, b, q, l, n1, n2, n_1, m_1,
> q_1, c1, c2, mA2, qA2, NA2, X1, d11, D1p, X, h, Стрелка, Ra,
> Rb, cosf, sinf, qB, mB, mA, H, i, j,
> Арка, ЭпюрQ, ЭпюрM, ЭпюрN, ЭпюрQ2, ЭпюрM2, ЭпюрN2, Рис:
> use Maplets[Tools] in
> a:=Get('Ta'::list); # Коорд. точек прилож.сил, м
> F:=Get('TF'::list); # Силы, кН
> b:=Get('Tb'::list); # Координата начала q, м
> q:=Get('Tq'::list); # Распр. нагрузка q, кН/м
> l:=Get('Tl'::list): # Длина распр. нагрузки q, м
> h:=Get('Th'::float);# Высота сеч., м
> f:=Get('Tf'::float);# Стрела подъема, м
> L:=Get('TL'::float);# 0.5 пролета, м
```



```

> n1:=nops(a): n2:=nops(q): # Количество F и q
> Rb:=- (add(F[i]*a[i],i=1..n1)+
> add(q[i]*l[i]*(b[i]+l[i]/2),i=1..n2))/2/L:
> Ra:=-add(F[i],i=1..n1)-add(q[i]*l[i],i=1..n2)-Rb:
> cosf:=x->subs(t=x,1/sqrt(1+diff(y(t),t)^2)):
> sinf:=x->subs(t=x,diff(y(t),t))*cosf(x):
> c1:=(h/f)^2/12; c2:=h^2/f/12;
> m_1:=x-> y(x)/f: #Единиичн. эпюра m
> q_1:=x-> sinf(x)/f:#Единиичн. эпюра q
> n_1:=x-> cosf(x)/f:#Единиичн. эпюра n
> y:=proc(x) f/L^2*(2*L-x)*x:end: # Форма оси арки
> n1:=n1+1:
> F:=[Ra,op(F)]:#Добавл. реакц. опоры в число сил
> a:=[0,op(a)]: #Координата левой опоры
  Балочн. поперечн. сила
> qBF:=x->add(piecewise(x>=a[i],F[i],x<=a[i],0),
> i=1..n1):
> qB:=qBF+add(piecewise(x>=b[i] and
> x<=b[i]+l[i],q[i]*(x-b[i]),
> x<=b[i],0,x>=b[i]+l[i],q[i]*l[i]),i=1..n2):
  Балочный момент
> mBF:=x->add(piecewise(x>=a[i],(x-a[i])*F[i],
> x<=a[i],0,0), i=1..n1):
> mB:=mBF+add(piecewise(x>b[i] and x<b[i]+l[i],
> (x-b[i])^2/2*q[i],x>=b[i]+l[i],
> (x-b[i]-l[i]/2)*l[i]*q[i],0),i=1..n2):
> Стрелка:=width=0.05,head_length=0.4,
> head_width=0.2,color=red:# Параметры стрелки
> распрнагр:=CURVES(seq([b[i],f+0.5],
> [b[i]+l[i],f+0.5]),i=1..n2),THICKNESS(10)):
> нагр:=arrow({seq([a[i],y(a[i])+1],i=2..n1)},
> {seq([0,-1],i=2..n1)},Стрелка):
> величина:=PLOT(seq(TEXT([a[i],y(a[i])+1],
> convert(abs(F[i]),symbol),ALIGNRIGHT),i=2..n1)):
> шарниры:=PLOT(POINTS([L,f],[0,0],[2*L,0],
> SYMBOL(CIRCLE)),COLOR(RGB,1,0,0)):
> m0:=0: for j to n1 do
> if a[j]<L then m0:=m0+(L-a[j])*F[j]; fi;

```

```

> od:
> for i to n2 do if b[i]<L then
> if b[i]+l[i]<L
> then m0:=m0+(L-b[i]-l[i]/2)*q[i]*l[i];
> else m0:=m0+q[i]*(L-b[i])^2/2;
> fi;
> fi:
> od:
> H:=m0/f; # Распор
> qA:=x->qB(x)*cosf(x)-H*sinf(x): #Поперечн. сила
> NA:=x->-qB(x)*sinf(x)-H*cosf(x):#Продольн. сила
> mA:=x->mB(x)-H*y(x): # Момент в трехшарнирной арке
> вид:=color=[red,blue],thickness=[1,2],
> legend=["Arch", "Beam"]:
> D1p:=int(m_1(x)*mA(x)/cosf(x)+c2*NA(x),x=0..2.*L);
> d11:=int(m_1(x)^2/cosf(x)+c1*cosf(x),x=0..2.*L);
> X1:=-D1p/d11;
> mA2:=x->m_1(x)*X1+mA(x):
> qA2:=x->q_1(x)*X1+qA(x):
> NA2:=x->n_1(x)*X1+NA(x):
> Set(TRa=Ra): Set(TRb=Rb): Set(TX1=X1):
> Set(TD1p=D1p): Set(Td11=d11):
> X:=x=0..2.*L:
> Арка:=display(plot(y(x),X,color=blue)):
> ЭпюрM:=display(plot([mA(x),mB(x)],
> X,вид,title="Еpure M3"));
> ЭпюрQ:=display(plot([qA(x),qB(x)],
> X,вид,title="Еpure Q3"));
> ЭпюрN:=display(plot(NA(x),X,title="Еpure N3"));
> ЭпюрM2:=display(plot([mA2(x),mB(x)],
> X,вид,title="Еpure M2"));
> ЭпюрQ2:=display(plot([qA2(x),qB(x)],
> X,вид,title="Еpure Q2"));
> ЭпюрN2:=display(plot(NA2(x),X,title="Еpure N2"));
> Рис:=display(распрнагр, величина, шарниры, Арка, нагр);
> end use:
> Рис, ЭпюрM, ЭпюрQ, ЭпюрN, ЭпюрM2, ЭпюрQ2, ЭпюрN2
> end proc:

```

```
> ПускArch:=proc()
> local Кнопка,Ev,i,Txt,a,c,mplt,Экран;
> use Maplets:-Elements in
> Экран:=Plotter['PL1']('height=300'):
> a[1]:=["[xP]=",TextField['Ta'](9,'value'=[4,10])]:
> a[2]:=["[P]=", TextField['TF'](9,'value'=[-9,-6])]:
> a[3]:=["[xq]=",TextField['Tb'](9,'value'=[6])]:
> a[4]:=["[q]=", TextField['Tq'](9,'value'=[-2])]:
> a[5]:=["Lq=",TextField['Tl'](9,'value'=[6]
> a[6]:=["h=", TextField['Th'](9,'value'=0.6)]:
> a[7]:=["f=", TextField['Tf'](9,'value'=4.0)]:
> a[8]:=["L=", TextField['TL'](9,'value'=6.0)]:
> a[9]:=["Ra",TextField['TRa'](9)]:
> a[10]:=["Rb",TextField['TRb'](9)]:
> a[11]:=["D1p",TextField['TD1p'](9)]:
> a[12]:=["d11",TextField['Td11'](9)]:
> a[13]:=["X1",TextField['TX1'](9)]:
> Ev:=proc(n)
> Evaluate(PL1 = 'display(Решение()[n])'):
> end;
> Кнопка:="Арка","M3","Q3","N3","M2","Q2","N2":
> for i to 7 do
> c[i]:=Button(Кнопка[i],Ev(i)):
> od:
> mplt:= Maplet(Window('title'="Арка",
> [ BoxRow(Экран,BoxColumn(seq(a[i],i=1..8)),
> [seq(c[i],i=1..7)]),[seq(a[i],i=9..13)]]
> ) #Window
> ):#Maplet
> Maplets:-Display(mplt):
> end use;
> end proc;
> end module:
> ArchМодуль:-ПускArch():
```

7.30. Устойчивость

7.30.1. Критическая нагрузка. Маплет

Программа для расчета критической нагрузки продольно сжатого стержня оформлена в виде маплета. Интерфейс программы (рис. 451) состоит из пяти полей ввода данных (константы материала, радиус инерции, длина и площадь сечения), четырех полей вывода. Внизу окна приведены рисунки четырех видов закреплений, соответствующие коэффициенты приведения и кнопки выбора. В качестве

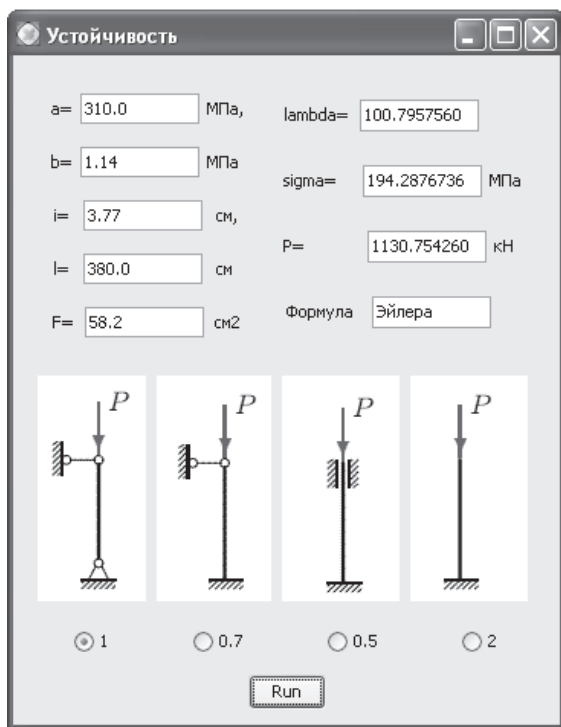


Рис. 451. P50.maplet

тестовой задачи использована задача 68 на с. 210. Для расчета достаточно указать способ закрепления и нажать кнопку Run. Программа произведет все расчеты и выдаст значение гибкости, критическое напряжение и критическую нагрузку, название примененной расчетной

формулы. В данной задаче для $\mu = 1$ и $\mu = 2$ расчет производится по формуле Эйлера, для $\mu = 0,5$ и $\mu = 0,7$ — по формуле Ясинского.

В целом программа записана в модуль `BucklModule` с двумя процедурами `SolvP` и `runB1`. В первой производятся вычисления, во второй организован интерфейс. Критическую гибкость, разделяющую потерю устойчивости в упругой стадии и потерю устойчивости в состоянии пластичности, берем равной $\lambda = 100$. Коэффициент $1/10$ в формуле $P_{kr} = \sigma * F / 10$ введен для того, чтобы согласовать силу P_{kr} в кН, напряжение σ в МПа, площадь F в см². Выбор способа закрепления (коэффициента μ) производит оператор, построенный на основе условного оператора. Работа этого оператора описана на с. 231.

Программа запускается вызовом процедуры `runB1` из модуля `BucklModule`.

Последними в процедуре `InfLin` приведены `Vmax`, `Vmin`, `GR`, поэтому для вызова из процедуры соответствующих результатов надо указать порядковый номер: `InfLin()[1]`, `InfLin()[2]`, `InfLin()[3]`.

Рисунки способов закрепления не интегрируются в программу, а хранятся в виде файлов `1.GIF`, `2.GIF`, `3.GIF`, `4.GIF` в том же каталоге, где и маплет, причем, если по какой-либо причине их там не оказалось, программа все равно будет работать, но без рисунков, а только с кнопками и соответствующими подписями. Более того, здесь несущественен регистр букв в имени файла: можно писать и `1.gif` вместо `1.GIF`.

Программа 50 (P50.mws)

```
> restart: # Маплет. Устойчивость стержня
> BucklModule:=module()
> export SolvP,runB1:
> SolvP:=proc() local sigma,a,
>         b,E,F,rho,l,mu,lambda:
> E:=2e5:
> use Maplelets[Tools] in
> a:=Get('Ta'::float); #Константы пластичности
> b:=Get('Tb'::float); #Константы пластичности
> rho:=Get('Tr'::float); #Радиус инерции
> l:=Get('Tl'::float); #Длина стержня
> F:=Get('TF'::float); #Площадь сечения
> if Get('C1') then mu:=1;
>     elif Get('C2') then mu:=0.7;
>     elif Get('C3') then mu:=0.5;
>     else mu:=2;
```

```

> fi;
> lambda:=mu*l/rho:          #Гибкость
> Set(Tlambda=lambda):
> if lambda>100 then        #Критическое напряжение
>   sigma:=evalf((Pi/lambda)^2*E); Set(Tl0="Эйлера"):
>   else sigma:=a*b*lambda: Set(Tl0="Ясинского"):
> fi;
> Set(Pkr=sigma*F/10):
> end use;
> sigma;
> end proc;

> runB1:=proc()
> local mplt,R,GB,W,i,l,I;
> use Maplets:-Elements in
> GB:=ButtonGroup[GG]();
> I[1]:=Image[im1]("1.GIF"):
> I[2]:=Image[im2]("2.GIF"):
> I[3]:=Image[im3]("3.GIF"):
> I[4]:=Image[im4]("4.GIF"):
> R[1]:=RadioButton['C1']("1",group='GG'):
> R[2]:=RadioButton['C2']("0.7",group='GG'):
> R[3]:=RadioButton['C3']("0.5",group='GG'):
> R[4]:=RadioButton['C4']("2",group='GG'):
> for i to 4 do l[i]:=Label(image=im||i): od:
> W:=width=9:
> mplt:=Maplet(seq(I[i],i=1..4),
>               Window("Устойчивость",
> [[BoxColumn(["a=",TextField['Ta']
>               (W,'value'=310.0),"МПа,"],
> ["b=",TextField['Tb'](W,'value'=1.14),"МПа"],
> ["i=",TextField['Tr'](W,'value'=3.77),"см"],
> ["l=",TextField['Tl'](W,'value'=380.0),"см"],
> ["F=",TextField['TF'](W,'value'=58.2),"см2"]],
> BoxColumn(BoxRow("lambda=",TextField['Tlambda'](W)),
> BoxRow("sigma="      ,TextField['sigm'](W),"МПа"),
> BoxRow("P="          ,TextField['Pkr'](W),"кН"),

```

```

> BoxRow("Формула ", TextField['T10'](W), " ")),
> [seq(l[i], i=1..4)],
> [seq(R[i], i=1..4)],
> Button("Run", Evaluate(sigm=SolvP))]
>
>                               ), :#Window
>                               GB):#Maplet
> Maplets:-Display(mplt):
> end use;
> end proc;
> end module:
> BucklModule:-runB1():

```

7.30.2. Подбор сечения

Подбор сечения сжатого стержня по заданной критической нагрузке традиционно производится методом итераций, как это было продемонстрировано в решении задачи 69, с. 213. Система **Maple** позволяет решить эту задачу иначе — разрешив уравнение

$$P_{кр} = [\sigma]F\varphi \quad (7.11)$$

относительно гибкости λ . Здесь $P_{кр}$ и $[\sigma]$ — заданные величины (нагрузка и допустимое напряжение). Пусть размер сечения характеризуется параметром b , имеющим размерность длины. Для квадратного сечения это длина стороны квадрата, для круглого — радиус. Покажем, что правая часть (7.11) зависит от λ . Зависимость $\varphi = \varphi(\lambda)$ известна и задана таблицей 13, с. 490. Имеем зависимости для площади и радиуса инерции: $F = F(b)$, $i = i(b)$. Кроме того, $\lambda = \mu l / i$. Исключая из последних зависимостей b , находим функцию $F = F(\lambda)$. Для квадрата¹ $F = b^2$, $J = b^4/12$ и $F = 12(\mu l / \lambda)^2$. Аналогично для круга $F = \pi b^2$, $J = \pi b^4/4$ и $F = 3\pi(\mu l / \lambda)^2$.

Таким образом, действительно получаем уравнение (7.11) для λ .

В решении этой задачи существенно используется возможность **Maple** численно решать нелинейные уравнения с кусочно-заданными функциями. Сначала функцию $\varphi(\lambda)$ гибкости аппроксимируем линейным сплайном с помощью оператора **spline** и заносим ее в процедуру **phi**. Длина списка аргументов λ , начинающегося от 0, определяется длиной списка значений φ , которая для стали и дерева равна 21, а для чугуна — 11. Для вычисления длины используем оператор **nops**, для создания последовательности значений λ с шагом 10 — оператор $[i*10\$i=0..nops(koef[k])-1]$, где $\$$ — оператор повтора. В программу введены списки φ для стали (**koef[1]**), чугуна (**koef[2]**) и

¹Площади и моменты инерции различных сечений даны в табл. 7, с. 485.

дерева (`koef[3]`). Поэтому процедура `phi` имеет два аргумента: x — это гибкость λ , k — номер материала (1, 2, 3).

Если использовать кубические сплайны (`cubic`), то результат мало изменится: $b = 0,5080969123$. Заметим также, что в **Maple** есть специальный пакет `CurveFitting` для интерполяции данных и в нем оператор `PolynomialInterpolation`. В примере взяты данные задачи 69, с. 213. Для решения другой задачи надо сменить значения P , σ , μ , n , коэффициент $K = b/i$, зависящий от формы сечения. Для круга $K = b/\sqrt{J/F} = b/\sqrt{\pi b^4/(4\pi b^2)} = 2$.

Программа 51 (P51.mws)

```

> restart;#Подбор сечения сжатого стержня
    сталь
> koef[1]:=[1,0.99,0.96,0.94,0.92,0.89,0.86,0.81,0.75,0.69,
> 0.60,0.52,0.45,0.40,0.36,0.32,0.29,0.26,0.23,0.21,0.19]:
    чугун
> koef[2]:=[1,0.97,0.91,0.81,0.69,0.57,
>
>                                0.44,0.34,0.26,0.20,0.16]:
    дерево
> koef[3]:=[1,0.99,0.97,0.93,0.87,0.8,0.71,0.61,0.48,0.38,
> 0.31,0.26,0.22,0.18,0.16,0.14,0.12,0.11,0.1,0.09,0.08]:
> phi:=proc(x,k)
> spline([i*10$ i=0..nops(koef[k])-1],
>
>                                koef[k],x,linear);end:
> P:=100:    # Нагрузка кН
> sigma:=130:# Допускаемое напряжение, МПа
> mu:=1:    # Коэффициент приведения (1,0.5,0.7,2)
> l:=1.1:   # Длина (м)
> n:=2:    # Чугун
> K:=2*sqrt(3.):# Квадрат
> b:=l*mu/lambda*K:
> eq:=P-sigma*1E3*b^2*phi(lambda,n):
> lambda:=fsolve(eq,lambda=0..200);
>
>                                lambda := 75.12552571
> b;
>
>                                0.05072193160

```

7.31. Стабильность

7.31.1. Обыкновенное дифференциальное уравнение

Следующая программа позволяет рассчитывать точки неустойчивости процесса, описываемого нелинейным обыкновенным дифференциальным уравнением произвольного порядка. Параметр `Nmax` определяет максимальный порядок варьируемой производной. Разность между `Nmax` и порядком дифференциального уравнения определяет число дополнительных линеаризованных уравнений, полученных из исходного линеаризованного уравнения дифференцированием.

Оператор `declare(x(t))` пакета `PDEtools` использован только для компактной записи производной функции $x(t)$ по времени в виде x_t . По умолчанию `Maple` выводит производные в виде $\frac{d}{dt}x(t)$. Пакет линейной алгебры `LinearAlgebra` требуется для вычисления определителя (`Determinant`).

Линеаризация в программе сводится к вычислению коэффициентов при соответствующих частных производных. Это выполнено в двойном цикле с вычислением элементов `A[i,j]` матрицы неустойчивости. При вычислении коэффициентов при Δx надо находить производные нулевого порядка, что оператор `diff` не позволяет делать. Поэтому в начале программы вводится пользовательский оператор `Dif`, в котором предусмотрена и такая производная. Для того чтобы не выводить на экран комплексные решения, необходимо задать глобальную переменную `_EnvExplicit:=false`. Другая глобальная переменная `_MaxSols`, использованная в программе, ограничивает число выводимых решений. Здесь мы воспользовались тем, что сначала `Maple` выдает вещественные решения, и назначили `_MaxSols:=4`, чтобы не выводить комплексные числа.

В программе использован оператор вычитания множеств `minus`.

Программа 52 (P52.mws)

```
> restart; # Точки неустойчивости. ОДУ
  Процедура дифференцирования по t любого порядка
> Dif:=proc(x,i) if i=0 then x
>                                     else diff(x,t$i) fi end:
> with(LinearAlgebra):
> with(PDEtools,declare): declare(x(t)):
  Максимальный порядок
> Nmax:=3: N:=Nmax-2: A:=Matrix(N+1):
  Исследуемое уравнение (задача 70, с. 217)
> eq:=16*diff(x(t),t$2)*sqrt(diff(x(t),t))+
```

> $b \cdot \text{diff}(x(t), t) + x(t)^4;$

$$eq := 16 x_{t,t} \sqrt{x_t} + b x_t + x^4$$

Производные уравнения по времени порядка от 0 до N

> for k from 0 to N do z[k+1]:=Dif(eq, k); od:

Задаем порядок особой точки и скорость

> Порядок:={0, 3}; v0:=1:

Порядок искомым производных (для уравнения 2-го порядка)

> SET:={seq(j, j=0..Nmax)} minus Порядок;

$$SET := \{1, 2\}$$

Заполнение матрицы

> for i to N+1 do

> for j to N+1 do

> A[i, j]:=subs(xx=Dif(x(t), SET[j]),

> diff(subs(Dif(x(t), SET[j])=xx, z[i]), xx));

> od:

> od:

> A;

Условие неустойчивости

> Dtr:=Determinant(A);

$$\begin{bmatrix} \frac{8 x_{t,t}}{\sqrt{x_t}} + b & 16 \sqrt{x_t} \\ \frac{8 x_{t,t,t}}{\sqrt{x_t}} - \frac{4 x_{t,t}^2}{x_t^{(3/2)}} + 4 x^3 & \frac{16 x_{t,t}}{\sqrt{x_t}} + b \end{bmatrix}$$

$$Dtr := -\frac{-192 x_{t,t}^2 - 24 x_{t,t} b \sqrt{x_t} - b^2 x_t + 128 x_{t,t,t} x_t + 64 x^3 x_t^{(3/2)}}{x_t}$$

Выражение 2-й производной функции из уравнения (необходимо, если в определитель Dtr входит 2-я производная)

> xtt:=solve(eq, diff(x(t), t\$2));

> xttt:=diff(xtt, t);

$$xtt := -\frac{1}{16} \frac{b x_t + x^4}{\sqrt{x_t}}$$

$$xttt := -\frac{1}{16} \frac{b x_{t,t} + 4 x^3 x_t}{\sqrt{x_t}} + \frac{1}{32} \frac{(b x_t + x^4) x_{t,t}}{x_t^{(3/2)}}$$

Подстановка 2-й и 3-й производной в условие неустойчивости

> Dtr1:=simplify(subs(diff(x(t), t\$3)=xttt, Dtr));

> Dtr2:=simplify(subs(diff(x(t), t\$2)=xtt, Dtr1));

$$Dtr1 := -\frac{-192 x_{t,t}^2 \sqrt{x_t} - 28 x_{t,t} b x_t - b^2 x_t^{(3/2)} + 32 x_t^2 x^3 + 4 x_{t,t} x^4}{x_t^{(3/2)}}$$

$$Dtr2 := -\frac{x^3(-x^5 + 32xt^{(5/2)})}{xt^2}$$

```
> _MaxSols:=4:
> solve(subs(diff(x(t),t)=v0,Dtr2),x(t));
0, 0, 0, 2
```

7.31.2. Условие стабильности процесса на плоскости

В программе исследована система уравнений (4.88–4.89) задачи 71, с. 218. В специальной процедуре `NC` в двойном цикле по i и j формируется матрица неустойчивости. Здесь же вычисляется определитель матрицы и подставляются значения вторых производных, найденных ранее из исследуемой системы уравнений. Числитель определителя выделяется оператором `numer`¹.

Кривую неустойчивости, полученную в неявном виде, удобнее всего строить с помощью специального оператора `implicitplot`. Для улучшения качества рисунка (сглаживания погрешности численных расчетов) можно увеличить число используемых для кривой точек. Для этого параметр `numpoints` следует увеличить. По умолчанию `numpoints=50`. Время счета при этом увеличивается.

Программа 53 (P53.mws)

```
> restart; # Точки неустойчивости. Система ОДУ
  Процедура дифференцирования по t любого порядка
> Dif:=proc(x,i) if i=0 then x
>                                     else diff(x,t$ i) fi end:
> with(LinearAlgebra): x1:=X(t): x2:=Y(t):
> with(PDEtools,declare): declare(x1,x2):
  Исследуемая система уравнения
> eq[1]:=diff(x1,t$2)*(x1^2+3*x2)+2*x1*x2
>                                     +diff(x1,t)*(2*x1+3*x2);
> eq[2]:=diff(x1,t$2)*x1+7*diff(x2,t$2)*x1+4*x1+3*x2;
  eq1 := X_{t,t}(X^2 + 3Y) + 2XY + X_t(2X + 3Y)
  eq2 := X_{t,t}X + 7Y_{t,t}X + 4X + 3Y
  Выражение ускорений из системы уравнений
> EQ:=subs(diff(x1,t$2)=xtt,diff(x2,t$2)=ytt,
>                                     {eq[1],eq[2]}):
> assign(solve(EQ,{xtt,ytt}));
```

¹Знаменатель в `Maple` выделяется оператором `denom`.

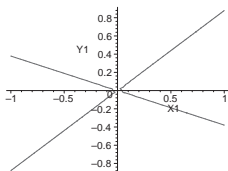


Процедура нахождения условия неустойчивости с подстановкой ускорений(xtt, ytt) из уравнений движения

```
> NC:=proc(M) local i,j,Dtr;global A;
> A:=Matrix(2):
> for i to 2 do
>   for j to 2 do
>     A[i,j]:=subs(xx=Dif(x||j,M),
>       diff(subs(Dif(x||j,M)=xx,eq[i]),xx));
>   od:
> od:
> Dtr:=simplify(Determinant(A));
> collect(numer(simplify(subs({diff(x1,t$2)=xtt,
>   diff(x2,t$2)=ytt},Dtr))),X(t));
> end proc:
  Условие неустойчивости с учетом значений скоростей
> U0:=subs(diff(x1,t)=1,diff(x2,t)=0,NC(0));
      
$$U0 := -6X^2 - 9XY + 18Y^2$$

> A; # Матрица исследуемой системы
      
$$\begin{bmatrix} 2X_{t,t}X + 2Y + 2X_t & 3X_{t,t} + 2X + 3X_t \\ X_{t,t} + 7Y_{t,t} + 4 & 3 \end{bmatrix}$$

> with(plots):
> W0:=subs(x1=X1,x2=Y1,U0):
> implicitplot(W0,X1=-1..1,Y1=-1..1,numpoints=2000);
```



```
> solve(W0,Y1);
```

$$\left(\frac{1}{4} + \frac{\sqrt{57}}{12}\right)X1, \left(\frac{1}{4} - \frac{\sqrt{57}}{12}\right)X1$$


```

> subs(xx=Dif(u,n,z),
>
> diff(subs(Dif(u,n,z)=xx,eqv[i]),xx))
> end proc:
> for i to 3 do
> A[i,1]:=sbs(i,0,x);#=sbs(i,0,y). Коэфф. при U
> A[i,2]:=sbs(i,1,x);# Коэфф. при Ux в yp-e № i
> A[i,3]:=sbs(i,1,y);# Коэфф. при Uy в yp-e № i
> od:

```

> A;# Матрица неустойчивости

$$\begin{bmatrix} xy & y(xy-1) & x(xy+1) \\ y & y^2+xy & 2xy+1 \\ x & 2xy-1 & xy+x^2 \end{bmatrix}$$

Условие неустойчивости

```
> Dtr:=simplify(Determinant(A));
```

$$Dtr := -xy + y^3 x - x^3 y$$

Выделяем множители

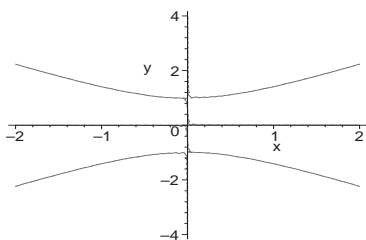
```
> factor(Dtr);
```

$$-xy(1 - y^2 + x^2)$$

```
> with(plots):
```

```
> implicitplot(Dtr,x=-2..2,y=-4..4,
```

```
> numpoints=2000,thickness=2);
```



7.31.4. Неустойчивость сжатого стержня в условии ползучести

В задаче 73, с. 221 опущены выкладки при выводе второго уравнения в системе для производных прогиба, из которой определяется условие неустойчивости. Методами Maple все выкладки существенно

упрощаются. Более того, есть возможность вывести условия неустойчивости произвольного порядка N . В приведенной программе принимаем $N = 2$.

В систему продифференцированных уравнений входят производные деформации ползучести \dot{p} , \ddot{p} и производные высших порядков. В силу определяющего соотношения $\dot{p}p^\alpha = f(\sigma) = \text{const}$, высшие производные выражаются через p и \dot{p} . Поэтому их надо исключить из системы. Для этого используется оператор `eliminate`. Исключаемые производные обозначаются как $D1$, $D2$, и т. д. и формируются в список `z2`. Аналогично формируется список `z4` производных в левой части системы (вектор \bar{U} , (4.106), с. 224). Заметим, что порядок ввода замен производных существенен. Сначала надо обозначать высшие производные, а потом низшие. В противном случае, как только будет как-то обозначена, например, первая производная, **Maple** сразу воспримет ее как константу, продифференцирует в ноль, и высшие производные уже не придется обозначать — их просто не будет. Условие неустойчивости можно получить двумя способами. Первый способ — решить систему (4.106) и приравнять знаменатель решения нулю. Знаменатель в **Maple** выделяется оператором `denom`. Другой способ — получить по системе уравнений ее матрицу и приравнять определитель матрицы нулю. В пакете `LinearAlgebra` имеется такой оператор: `GenerateMatrix`. В полученной матрице все элементы раскладываются на множители с помощью оператора `map`. Оба способа дают один и тот же результат. Чтобы не вводить временные переменные, в некоторых случаях применена дитто операция `%` (знак результата предыдущей операции). В таких программах запускать счет надо не с произвольной точки командой `ENTER`, а только с начала или, еще проще и надежнее, из меню, командой из трех восклицательных знаков «!!!».

Для $N > 2$ программа получает только условие неустойчивости. Решение же алгебраических уравнений оператором `solve` высоких порядков либо слишком громоздкое, либо решения нет.

Программа 55 (P55.mws)

```
> restart;# Точки нестаб. при ползучести (стержень)
> N:=2: # Порядок особой точки
> U:=u(t):# Прогиб
> P:=p(t):# Деформация ползучести
> with(PDEtools):
> declare(prime=t,u(t),p(t)):#Штрих-произв. по t
> eq1:=diff(U,t)+U*diff(P,t)*(alpha/P-lambda):
> ur1:=diff(P,t)*P^alpha=c: #Соотнош. ползучести
> s1:=seq(diff(eq1,t$1),i=1..N-1):#Продифф. уравн.
```

```

> s2:=seq(diff(ur1,t$xi),i=1..N-1):
> z1:=seq(diff(p(t),t$N+1-i)=D||i,i=1..N-1):
> z2:=seq(D||i,i=1..N-1): # Список произв. деф. полз.
> z3:=seq(diff(u(t),t$N+1-i)=w||i,(N+1-i),i=1..N):
> z4:=seq(w||i,i=0..N-1): # Список произв. прогиба
> syst:=subs(z1,{eq1,s1,s2}):# Полная система уравн.
> eq5:=eliminate(%,{z2}): # Исключаем производные
> syst:=op(eq5)[2]:
> S1:=subs(z3,u(t)=w0,syst):
> S2:=subs(lambda=xi/P,S1):
> solve(S2,{z4}):

```

$$\left\{ w_0 = \frac{w_2 p^2}{p'^2 (2\alpha^2 - 3\alpha\xi + \alpha + \xi^2)}, w_1 = \frac{p(-\alpha + \xi) w_2}{p' (2\alpha^2 - 3\alpha\xi + \alpha + \xi^2)} \right\}$$

```

> eq:=collect(denom(rhs(%[1])),xi,factor);

```

$$eq := p'^2 \xi^2 - 3 p'^2 \alpha \xi + p'^2 \alpha (2\alpha + 1)$$

```

> with(LinearAlgebra):
> A:=GenerateMatrix(S2,[z4]):

```

$$A := \begin{bmatrix} p' \alpha - p' \xi & p \\ p'^2 \alpha \xi - p'^2 \alpha - p'^2 \alpha^2 & p' p \alpha - p' p \xi \end{bmatrix}, \begin{bmatrix} 0 \\ -w_2 p^2 \end{bmatrix}$$

```

> map(factor,A[1]): # Разложение на множ. всех элем.

```

$$\begin{bmatrix} -p'(-\alpha + \xi) & p \\ p'^2 \alpha(-\alpha + \xi - 1) & -p' p(-\alpha + \xi) \end{bmatrix}$$

```

> Determinant(A[1]):
> solve(eq,xi): # Особая точка

```

$$\frac{3\alpha}{2} + \frac{\sqrt{\alpha^2 - 4\alpha}}{2}, \frac{3\alpha}{2} - \frac{\sqrt{\alpha^2 - 4\alpha}}{2}$$

Приложение 1. Геометрические характеристики сечений

Таблица 7

Прямоугольник, треугольник, круг и половина круга

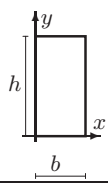
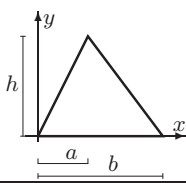
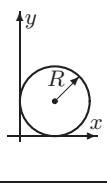
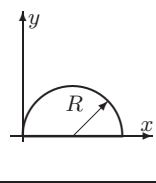
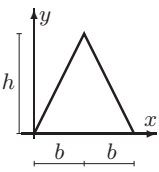
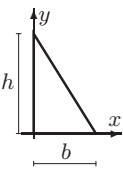
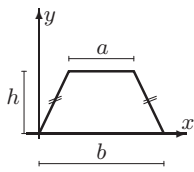
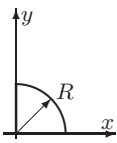
				
F	bh	$\frac{bh}{2}$	πR^2	$\frac{\pi R^2}{2}$
x_c	$\frac{b}{2}$	$\frac{a+b}{3}$	R	R
y_c	$\frac{h}{2}$	$\frac{h}{3}$	R	$\frac{4R}{3\pi}$
J_x	$\frac{bh^3}{3}$	$\frac{bh^3}{12}$	$\frac{5\pi R^4}{4}$	$\frac{\pi R^4}{8}$
J_y	$\frac{hb^3}{3}$	$\frac{hb(a^2 + ab + b^2)}{12}$	$\frac{5\pi R^4}{4}$	$\frac{5\pi R^4}{8}$
J_{xy}	$\frac{b^2 h^2}{4}$	$\frac{bh^2(b+2a)}{24}$	πR^4	$\frac{2R^4}{3}$
J_{xc}	$\frac{bh^3}{12}$	$\frac{bh^3}{36}$	$\frac{\pi R^4}{4}$	$R^4 \left(\frac{\pi}{8} - \frac{8}{9\pi} \right)$
J_{yc}	$\frac{hb^3}{12}$	$\frac{hb(a^2 - ab + b^2)}{36}$	$\frac{\pi R^4}{4}$	$\frac{\pi R^4}{8}$
J_{xyc}	0	$\frac{h^2 b(2a-b)}{72}$	0	0

Таблица 8

**Равнобедренный треугольник, прямоугольный треугольник,
равнобедренная трапеция и четверть круга**

				
F	bh	$\frac{bh}{2}$	$\frac{h(a+b)}{2}$	$\frac{\pi R^2}{4}$
x_c	b	$\frac{b}{3}$	$b/2$	$\frac{4R}{3\pi}$
y_c	$\frac{h}{3}$	$\frac{h}{3}$	$\frac{h(b+2a)}{3(a+b)}$	$\frac{4R}{3\pi}$
J_x	$\frac{bh^3}{6}$	$\frac{bh^3}{12}$	$\frac{h^3(3a+b)}{12}$	$\frac{\pi R^4}{16}$
J_y	$\frac{7hb^3}{6}$	$\frac{hb^3}{12}$	$\frac{h(a+b)(a^2+7b^2)}{48}$	$\frac{\pi R^4}{16}$
J_{xy}	$\frac{b^2h^2}{3}$	$\frac{b^2h^2}{24}$	$\frac{bh^2(2a+b)}{12}$	$\frac{R^4}{8}$
J_{xc}	$\frac{bh^3}{18}$	$\frac{bh^3}{36}$	$\frac{h^3((a+b)^2+2ab)}{36(a+b)}$	$R^4 \left(\frac{\pi}{16} - \frac{4}{9\pi} \right)$
J_{yc}	$\frac{hb^3}{6}$	$\frac{hb^3}{36}$	$\frac{h(a+b)(a^2+b^2)}{48}$	$R^4 \left(\frac{\pi}{16} - \frac{4}{9\pi} \right)$
J_{xyc}	0	$-\frac{h^2b^2}{72}$	0	$R^4 \left(\frac{1}{8} - \frac{4}{9\pi} \right)$

В таблицах 7 и 8 даны площадь фигуры F , координаты центра тяжести x_c , y_c , осевые (J_x , J_y) и центробежный J_{xy} моменты инерции относительно заданных осей и центральные моменты инерции J_{xc} , J_{yc} , J_{xyc} (относительно осей, проходящих через центр тяжести фигуры параллельно заданным).

Двутавры стальные горячекатаные, ГОСТ 8239-89

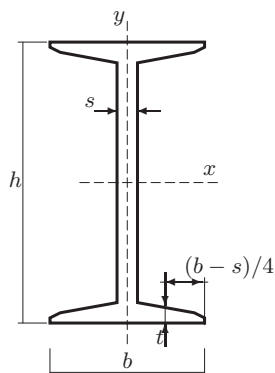


Таблица 9

№	ρ	Размеры, мм				F	J_x	W_x	S_x	J_y	W_y
	кг/м	h	b	s	t	см ²	см ⁴	см ³	см ³	см ⁴	см ³
10	9,46	100	55	4,5	7,2	12,0	198	39,7	23	17,9	6,49
12	11,5	120	64	4,8	7,3	14,7	350	58,4	33,7	27,9	8,72
14	13,7	140	73	4,9	7,5	17,4	572	81,7	46,8	41,9	11,5
16	15,9	160	81	5,0	7,8	20,2	873	109	62,3	58,6	14,5
18	18,4	180	90	5,1	8,1	23,4	1290	143	81,4	82,6	18,4
20	21,0	200	100	5,2	8,4	26,8	1840	184	104	115	23,1
22	24,0	220	110	5,4	8,7	30,6	2550	232	131	157	28,6
24	27,3	240	115	5,6	9,5	34,8	3460	289	163	198	34,5
27	31,5	270	125	6,0	9,8	40,2	5010	371	210	260	41,5
30	36,5	300	135	6,5	10,2	46,5	7080	472	268	337	49,9
33	42,2	330	140	7,0	11,2	53,8	9840	597	339	419	59,9
36	48,6	360	145	7,5	12,3	61,9	13380	743	423	516	71,1
40	57,0	400	155	8,3	13,0	72,6	19062	953	545	667	86,1
45	66,5	450	160	9,0	14,2	84,7	27696	1231	708	808	101
50	78,5	500	170	10,0	15,2	100	39727	1589	919	1043	123
55	92,6	550	180	11,0	16,5	118	55962	2035	1181	1356	151
60	108,0	600	190	12,0	17,8	138	76806	2560	1491	1725	182

Швеллеры стальные горячекатаные, ГОСТ 8240-89

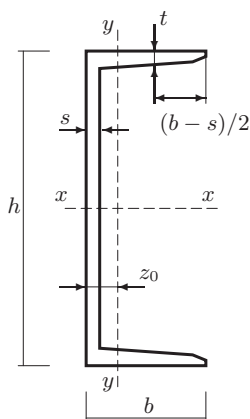


Таблица 10

№	ρ	Размеры, мм				F	J_x	W_x	S_x	J_y	W_y	z_0
	кг/м	h	b	s	t	см ²	см ⁴	см ³	см ³	см ⁴	см ³	см
5	4,84	50	32	4,4	7	6,16	22,8	9,1	5,59	5,61	2,75	1,16
6,5	5,9	65	36	4,4	7,2	7,51	48,6	15	9,01	8,7	3,68	1,24
8	7,05	80	40	4,5	7,4	8,98	89,4	22,4	13,3	12,8	4,75	1,31
10	8,59	100	46	4,5	7,6	10,9	174	34,8	20,4	20,4	6,46	1,44
12	10,4	120	52	4,8	7,8	13,3	304	50,6	29,6	31,2	8,52	1,54
14	12,3	140	58	4,9	8,1	15,6	491	70,2	40,8	45,4	11,0	1,67
16	14,2	160	64	5	8,4	18,1	747	93,4	54,1	63,3	13,8	1,8
16а	15,3	160	68	5	9	19,5	823	103	59,4	78,8	16,4	2
18	16,3	180	70	5,1	8,7	20,7	1090	121	69,8	86	17	1,94
18а	17,4	180	74	5,1	9,3	22,2	1190	132	76,1	105	20	2,13
20	18,4	200	76	5,2	9	23,4	1520	152	87,7	113	20,5	2,07
22	21	220	82	5,4	9,5	26,7	2110	192	110	151	25,1	2,21
24	24	240	90	5,6	10	30,6	2900	242	139	208	31,6	2,42
27	27,7	270	95	6	10,5	35,2	4160	308	178	262	37,3	2,47
30	31,8	300	100	6,5	11	40,5	5810	387	224	327	43,6	2,52
33	36,5	330	105	7	11,7	46,5	7980	484	281	410	51,8	2,59
36	41,9	360	110	7,5	12,6	53,4	10820	601	350	513	61,7	2,68
40	48,3	400	115	8	13,5	61,5	15220	761	444	642	73,4	2,75

Уголки стальные горячекатаные, ГОСТ 8509-86

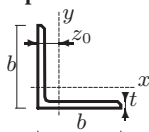


Таблица 11

№	ρ	b	t	F	J_x	J_{max}	$ J_{xy} $	z_0
	кг/м							
5	3,05	50	4	3,89	9,21	14,6	5,42	1,38
	3,77		5	4,80	11,2	17,8	6,57	1,42
5,6	3,44	56	4	4,38	13,1	20,8	7,69	1,52
	4,25		5	5,41	16,0	25,4	9,41	1,57
6,3	3,90	63	4	4,96	18,9	29,9	11	1,69
	4,80		5	6,13	23,1	36,8	13,7	1,74
	5,70		6	7,28	27,1	42,9	15,9	1,78
7	5,38	70	5	6,86	31,9	50,7	18,7	1,90
	6,39		6	8,15	37,6	59,6	22,1	1,94
7,5	5,80	75	5	7,39	39,5	62,6	23,1	2,02
	6,89		6	8,78	46,6	73,9	27,3	2,06
	7,96		7	10,1	53,3	84,6	31,2	2,1
8	6,78	80	5,5	8,63	52,7	83,6	30,9	2,17
	7,36		6	9,38	57,0	90,4	33,4	2,19
	8,51		7	10,8	65,3	104	38,3	2,23
9	8,33	90	6	10,6	82,1	130	48,1	2,43
	9,64		7	12,3	94,3	150	55,4	2,47
	10,9		8	13,9	106,0	168	62,3	2,51
10	10,8	100	7	13,8	131,0	207	76,4	2,71
	12,2		8	15,6	147,0	233	86,3	2,75
	15,1		10	19,2	179,0	284	110	2,83
	17,9		12	22,8	209,0	331	122	2,91
11	11,9	110	7	15,2	176,0	279	106	2,96
	13,5		8	17,2	198,0	315	116	3,00
12,5	15,5	125	8	19,7	294,0	467	172	3,36
	17,3		9	22,0	327,0	520	192	3,40
	19,1		10	24,3	360,0	571	211	3,45
	22,7		12	28,9	422,0	670	248	3,53
14	19,4	140	9	24,7	466,0	739	274	3,78
	21,5		10	27,3	512,0	814	301	3,82
	25,5		12	32,5	602,0	957	354	3,90
16	24,7	160	10	31,4	774,0	1229	455	4,30
	27,0		11	34,4	844,0	1340	496	4,35
	29,4		12	37,4	913,0	1450	537	4,39
	34,0		14	43,6	1046,0	1662	615	4,47
	38,5		16	49,1	1175,0	1866	690	4,55
18	30,5	180	11	38,8	1216,0	1933	716	4,85
	33,1		12	42,2	1317,0	2093	776	4,89

Приложение 2. Допускаемые напряжения и коэффициент продольного изгиба φ

Допускаемые напряжения, МПа

Таблица 12

	Сталь 2	Сталь 3	Сталь 5	Дюралю- миний Д16Т	Чугун серый	Дерево сосна вдоль волокон
Сжатие	140	160	240	100	120–150	10–12
Растяжение	140	160	240	100	28–80	7–10

Коэффициент продольного изгиба φ

Таблица 13

Гиб- кость λ	Ст. 2 Ст. 3 Ст. 4	Ст.5	Д16Т	Чугун	Дерево
0	1,00	1,00	1,00	1,00	1,00
10	0,99	0,98	1,00	0,97	0,99
20	0,96	0,96	1,00	0,91	0,97
30	0,94	0,93	0,84	0,81	0,93
40	0,92	0,89	0,70	0,69	0,87
50	0,89	0,85	0,57	0,57	0,80
60	0,86	0,80	0,46	0,44	0,71
70	0,81	0,74	0,35	0,34	0,61
80	0,75	0,67	0,27	0,26	0,48
90	0,69	0,59	0,21	0,20	0,38
100	0,60	0,50	0,17	0,16	0,31
110	0,52	0,43	0,14	–	0,26
120	0,45	0,37	0,12	–	0,22
130	0,40	0,32	0,10	–	0,18
140	0,36	0,28	0,09	–	0,16
150	0,32	0,25	0,08	–	0,14
160	0,29	0,23	–	–	0,12
170	0,26	0,21	–	–	0,11
180	0,23	0,19	–	–	0,10
190	0,21	0,17	–	–	0,09
200	0,19	0,15	–	–	0,08

Список литературы

1. *Авраамова О. Д.* Язык VRML. Практическое руководство. — М.: Диалог-МИФИ, 2000.
2. *Александров А. В., Потапов В. Д., Державин Б. П.* Сопротивление материалов: Учебник для вузов. — М.: Высш. шк., 1995.
3. *Алексеев Е. Р., Чеснокова О. В.* Решение задач вычислительной математики в пакете Mathcad 12, MATLAB 7, Maple 9. — М.: ИТ Пресс, 2006.
4. *Амосов А. А., Дубинский Ю. А., Копченова Н. В.* Вычислительные методы. — М.: Издательский дом МЭИ, 2008. — 672 с.
5. *Бать М. И., Джанелидзе Г. Ю., Кельзон А. С.* Теоретическая механика в примерах и задачах. Т.2. — СПб.: Лань, 2009.
6. *Бутенин Н. В., Луңц Я. Л., Меркин Д. Р.* Курс теоретической механики. — СПб.: Лань, 2009.
7. *Васильев А. Н.* Maple 8. Самоучитель. — М.: Изд. дом "Вильямс", 2003.
8. *Вильке В. Г.* Теоретическая механика. — СПб.: Лань, 2003.
9. *Гантмахер Ф. Р.* Теория матриц. — М.: Наука, 1966.
10. *Говорухин В. Н., Цибулин В. Г.* Компьютер в математическом исследовании. Учебный курс. — СПб.: Питер, 2001.
11. *Голоскоков Д. П.* Уравнения математической физики. Решение задач в системе Maple. Учебник для вузов — СПб.: Питер, 2004.
12. *Голоскоков Д. П.* Практический курс математической физики в системе Maple. — СПб.: ООО «ПаркКом», 2010.
13. *Голубев Ю. Ф.* Основы теоретической механики. — М.: Изд-во МГУ, 2000.
14. *Горшков А. Г., Трошин В. Н., Шалашилин В. И.* Сопротивление материалов. — М.: ФИЗМАТЛИТ, 2002.
15. *Дарков А. В., Шапошников Н. Н.* Строительная механика. — СПб.: Лань, 2008.
16. *Дробот Ю. Б., Мазалов В. Н.* Решение дифференциальных уравнений в системе Maple. — Хабаровск: Изд-во ДВГУПС, 2004.
17. *Дробот Ю. Б.* Введение в систему Maple 10. — Хабаровск: Изд-во ДВГУПС, 2006.
18. *Дьяконов В. П.* Maple 10/11/12/13/14 в математических расчетах. — М.: ДМК-Пресс, 2011.
19. *Дьяконов В. П.* Mathematica 5/6/7. Полное руководство. — М.: ДМК-Пресс, 2009.
20. *Журавлев В. Ф.* Основы теоретической механики. — М.: Физматлит, 2008.



21. *Журов А. И., Карпов И. И., Шингарева И. К.* Основы Maple. Применение в механике. Препринт 536. — М.: Институт проблем механики, 1995.
22. *Зубов В. С., Шевченко И. В.* Структуры и методы обработки данных. Практикум в среде Delphi. — М.: Филинь, 2004.
23. *Зими́на О. В., Кириллов А. И., Сальникова Т. А.* Решебник. Высшая математика. — М.: ФИЗМАТЛИТ, 2006.
24. *Игнатъев Ю. Г.* Релятивистская кинетика неравновесных процессов в гравитационных полях. — Казань: ООО «Фолиантъ», 2010.
25. *Ке́пе О. Э., Виба Я. А., Гранис О. П.* и др. Сборник коротких задач по теоретической механике / Под ред. О. Э. Ке́пе. — СПб.: Лань, 2009.
26. *Кийко И. А.* О форме пластического слоя, сжимаемого параллельными плоскостями // ПММ, 2011, №1. с. 15–24.
27. *Кирсанов М. Н.* Решебник. Теоретическая механика / Под ред. А. И. Кириллова. — М.: ФИЗМАТЛИТ, 2008.
28. *Кирсанов М. Н.* Задачи по теоретической механике с решениями в Maple 11. — М.: ФИЗМАТЛИТ, 2010.
29. *Кирсанов М. Н.* Графы в Maple. — М.: ФИЗМАТЛИТ, 2007.
30. *Кирсанов М. Н.* Сборник экзаменационных задач по динамике. — М.: МЭИ, 2005.
31. *Клюшников В. Д.* Лекции по устойчивости деформируемых систем. — М.: МГУ, 1986.
32. *Колесников Н. Н., Сальникова Т. В., Якимова К. Е.* Классическая и аналитическая механика. Задачник / Под ред. В. В. Козлова. — М.: МГУ, 1998.
33. *Коробов В. И., Очков В. Ф.* Химическая кинетика: введение с Mathcad/Maple/MCS. — М.: Горячая линия. — Телеком, 2009.
34. *Кристалинский Р. Е., Шапошников Н. Н.* Вычисление геометрических характеристик плоских и пространственных фигур // Сборник научных трудов по строительной механике и теории надежности конструкций. — М.: Изд-во МАДИ, 2010. С. 186–189.
35. *Кристалинский Р. Е., Шапошников Н. Н.* Решение вариационных задач строительной механики в системе MATHEMATICA. — СПб.: Лань, 2010.
36. *Макаров Е. Г.* Сопротивление материалов на базе Mathcad. — СПб.: БХВ-Петербург, 2004.
37. *Манзон Б. М.* Maple V Power Edition. — М.: ИИД «Филинь», 1998.
38. *Маркеев А. П.* Теоретическая механика. — М.: Регулярная и хаотическая динамика, 2007.
39. *Маркус П., Минк Х.* Обзор по теории матриц и матричных неравенств. — М.: Наука, 1972.
40. *Мартыненко Ю. Г.* Движение твердого тела в электрических и магнитных полях. — М.: Наука, 1988.

41. *Мартынова Е. Д.* Определение свойств вязкоупругих материалов из экспериментов на внедрение сферического индентора в вязкоупругий образец // Упругость и неупругость / под ред. И. А. Кийко, Г. Л. Бровка, Р. А. Васина. — М.: Изд-во Моск. ун-та, 2011. с. 394-398.
42. *Матросов А. В.* Марле 6. Решение задач высшей математики и механики. — СПб.: БХВ-Петербург, 2001.
43. *Меркурьев И. В., Подалков В. В.* Динамика микромеханического и волнового твердотельного гироскопов. — М.: ФИЗМАТЛИТ, 2009.
44. *Мещерский И. В.* Задачи по теоретической механике. — СПб.: Лань, 2010.
45. *Минин Л. С., Хроматов В. Е., Самсонов Ю. П.* Расчетные и тестовые задания по сопротивлению материалов / Под ред. В. Е. Хроматова. — М.: Высш.шк., 2008.
46. *Окопный Ю. А., Радин В. П., Хроматов В. Е., Чирков В. П.* Механика материалов и конструкций. Сборник задач / Под ред. В. П. Чиркова. — М.: Машиностроение, 2004.
47. *Очков В. Ф.* Физические и экономические величины в Mathcad и Marle. — М.: Финансы и статистика, 2002.
48. *Новожилов И. В., Зацепин М. Ф.* Типовые расчеты по теоретической механике на базе ЭВМ. — М.: Высш. шк., 1986.
49. *Потапов В. Д., Александров А. В., Косицын С. Б., Долотказин Д. Б.* Строительная механика: Учебник для вузов. Кн. 1. Статика упругих систем / Под ред. В. Д. Потапова. — М.: Высш. шк., 2007.
50. *Пятницкий Е. С., Трухан Н. М., Ханукаев Ю. И., Яковенко Г. Н.* Сборник задач по аналитической механике. — М.: Физматлит, 1996.
51. Сборник заданий для курсовых работ по теоретической механике: Учебное пособие для техн. вузов / А. А. Яблонский, С. С. Норейко, С. А. Вольфсон и др. / Под ред. А. А. Яблонского. — 16-е изд. — М.: Интеграл-Пресс, 2008.
52. *Сдвижков О. А.* Математика на компьютере: Marle 8. — М.: СОЛОН-Пресс, 2003.
53. *Седов Л. И.* Механика сплошной среды. В 2-х тт. — СПб.: Лань, 2004.
54. *Соловьев М. Е.* Теория и задачи механики сплошных сред с пакетом Marle. Учебное пособие. — Ярославль: Изд-во ЯГТУ, 2006.
55. *Старовойтов Э. И.* Сопротивление материалов. — М.: Физматлит, 2008.
56. *Тарг С. М.* Краткий курс теоретической механики. — М.: Высш. шк., 2007.
57. *Швядкий В. С., Дзюзер В. Я., Мощинский А. В.* Инженерные расчеты в интегрированной среде Marle: Учебное пособие / Под ред. В. Я. Дзюзера — Екатеринбург: Изд-во АМБ, 2010.
58. *Яблонский А. А., Норейко С. С.* Курс теории колебаний. — СПб.: Лань, 2003.
59. *Kirsanov M. N.* Singular points of the creep deformation and buckling of a column // International Journal Eng. Science, 1997, V5, N3, pp. 221–227.

Предметный и именной указатель

- \$ 363
- &x 247
- _Envdiffopdomain 298
- __EnvExplicit 477
- __MaxSols 479
- 'error' 329
- 'stop' 323
- yellow 266
- Abel 302
- Abel Niels Henrik 297
- Abel2A, Abel2C 302
- abelsol 297
- above 278
- abs 444
- Action 454
- Action Script (язык) 363
- Add 244
- add 233, 353, 361, 393, 408, 435
- Adjoint(A) 244
- Adobe Photoshop 269
- AlertDialog 335
- algsols 239, 377
- ALIGNRIGHT 442, 469
- All Files 333
- always 309, 310
- ambientlight 288
- animate 271, 324
- animate3d 271
- animatecurve 271
- antisymmetric 246
- APPEND 240
- appendline 347, 426
- apply 232
- applyrule 377
- approvecaption 332
- approvecheck 328
- arc 279
- arccos 388
- arctan 341, 396
- array 239
- ArrayTools 349
- arrow 279, 346, 353, 358, 404
- as_needed 309, 310, 321
- assign 339, 367, 374, 403, 445, 479
- Assistants 274, 296
- assume 235, 368, 369, 406
- Assumed Variables 369
- asterisk 265, 290
- AudioTools 270
- autonomous 297
- axes 265, 346, 353, 367, 412, 442
- axesfont 265, 353
- background 307
- backslash 312, 327
- BackwardSubstitute 244
- band 246
- BandMatrix 244
- below 278
- Bernoulli 302
- bernoullisol 297
- Bessel 303
- Betti Enrico 118
- BilinearForm 245
- bipolar 266
- bipolarcylindrical 288
- bispherical 288
- black 353
- Blue 277
- blue 353, 356
- BlueViolet 277
- BOLD 292, 345, 416
- bold 308
- border 308, 419
- bottom 309, 315
- BOX 292, 412
- box 265, 290
- BoxCell 307, 320
- BoxColumn, BoxRow 307, 309
- boxed 265
- breakwidth 325
- Brown 266
- Button 310, 315, 348, 418, 427
- ButtonGroup 320
- bvp 295
- C 373
- capped 280



- caption 266, 308, 313, 417
cardioid 266
cardioidal 288
cardioidcylindrical 288
casscylindrical 288
cassinian 266
cat 462
center 308, 309, 314, 315
changecoords 272
CharacteristicMatrix 245
CharacteristicPolynomial 245, 256
CheckBox 311, 317
CheckBoxMenuItem 311
Chini 302
chinisol 297
Cholesky 253, 255
CIRCLE 292, 385
circle 264, 279, 282, 290, 356, 385
circle, geometry 70
Clairaut 302
Classic Worksheet 226, 243, 253
classical 295
closeonapprove 333
CloseWindow 329
codegen 373
collect 237, 375, 383, 481
COLOR 346, 385, 404
color 266, 346, 356, 358, 404
ColorDialog 330, 331
Column 245
ColumnDimension 245
ColumnOperation 245
ColumnSpace 245
combine 374
ComboBox 326
comet 299
CompanionMatrix 246
complex 255
complexplot 272
Concatenate 349
ConditionNumber 246
cone 280
ConfirmDialog 335
confocalellip 288
confocalparab 288
conical 288
conjugate 245, 248
constant 246
ConstantMatrix 246
ConstantVector 246
constrained 346, 353, 359, 367, 385
continuous 323, 324
contour 290
contourplot 272
contours 272, 288
convert 247, 270, 327, 346, 417
convertAlg 297
convertsys 297
coordinates 70
coords 266, 288
Copy 247
cortona 286
cos 363, 412
cosh 412
cot 232
Coulomb C. A. 32
Courier 308
Create 270
CROSS 292
cross 265, 282, 290
crossprod 9, 359
CrossProduct 9, 247, 353, 401, 403
csgn 368
cuboid 280, 293, 351
curve 280, 299
CurveFitting 476
CURVES 291, 346, 356, 416, 445
cutin,cutout 280
cylinder 280, 293
cylindrical 288
cylindrical arrow 351
D 375
dAlembert 302
DarkRed, DarkTurquoise 277
dash 267, 342
dashdot 267, 341
dchange 300
Dchangevar 300
dcoeffs 299
de2diffop 299
declare 294, 375, 383, 420, 477, 481
default 277
defaultbutton 310
delay 323
DeleteColumn 247
DeleteRow 247
denom 479, 483



- DEnormal 300
 Determinant 247, 480, 482
 dfieldplot 299
 Diagonal 247
 diagonal 246
 diagonalcross 265, 290
 DiagonalMatrix 248
 Dialog 308
 dialog 331
 DialogInput 308
 DIAMOND 292
 diamond 265, 282, 290
 diff 376, 479
 diff_table 300
 diffop2de 298
 Digits 261, 398, 401
 Dimension 248
 Dimensions 248
 directoriesonly 333
 directory 328, 333
 DirectX, рендерер 287
 disk 280
 display 272, 353, 367, 375, 427, 445
 displayprecision 250, 402, 460
 divide 256
 dodecahedron 277, 280
 done 324
 dot 267, 341
 dotprod 359
 DotProduct 248, 353
 draw 70
 DropDownBox 325
 dsolve 295, 375, 421
 dsolve[interactive] 295
 dualaxisplot 272
 Duffing 303
 dverk78 295

 editable 315, 347, 397, 426, 454
 Ei 236
 Eigenvalues 120, 248, 401
 Eigenvectors 248
 elif 231, 473
 eliminate 179, 390
 elcylindrical 288
 ellipse 281
 ellipsoidal 288, 303
 elliptic 266, 303
 ellipticArc 281

 else 481
 Emden 303
 enabled 313
 Encapsulated PostScript 266, 268
 end proc 481
 end use 347, 426, 453
 erf 303
 Error
 — (in Matrix) initializer 245
 — attempt to assign.. 246
 — cannot determine 230
 — differential algebra not specified 298
 — in LinearAlgebra 253
 — invalid input 232, 233, 377
 — ocured while running maplet 337
 Euclidean 350, 353, 403, 405
 EulerLagrange 374
 eval 233, 377
 evalb 254
 evalf 356, 435
 evalindent 377
 evalm 346, 353, 358
 Evaluate 312, 329, 348, 418, 427
 exact 302
 exact_linear 303
 exact_nonlinear 303
 exactsol 300
 expand 236, 373, 386
 export 424
 Export As 11
 exprseq 327, 407, 423

 factor 121, 238, 374, 383, 482
 false 231, 347
 fclose 239, 327, 334
 fieldplot 273
 File (меню) 11, 369
 FileDialog 332
 filefilter 333
 filename 333
 filesanddirectories 333
 fileselectionmode 333
 filesonly 333
 FileTools 270
 filled 266, 288, 321, 454
 filledregions 272, 289
 Filter 269
 filterdescription 333

- fisheye 289
- float 232, 349, 416, 473
- FONT 292, 345, 416
- Font 318, 347
- Font, font 308
- fontsize 325
- fopen 327, 334
- foreground 314
- Fortran 373
- fprintf 327
- frame 265
- frame_backwards 323
- frame_forward 323
- frames 323
- free 253
- Frobenius 256
- FrobeniusForm 248, 258
- from 293
- fscanf 432
- fsolve 243
- function 419, 427
- GAMMA 236
- gamma 412
- GaussianEliminationTutor 253
- gear 295
- Gegenbauer 303
- GenerateEquations 249
- GenerateMatrix 249, 483
- genfunc 122
- geometry 70
- Get 416, 424, 453
- GetFile 328
- GIF 474
- gif 314
- GivensRotationMatrix 249
- global 356, 416, 440, 480
- glossiness 289
- gold 266
- Goldenrod 277
- gradplot 273
- GramSchmidt 249
- Graphics 277
- gray 266
- green 266, 454
- Green George 391
- grey 266
- gridlines 266
- GridRow 320
- group 426
- halign 308, 314
- Halm 303
- hamilton_eqs 301, 387
- HankelMatrix 249
- head_length 353
- Height 270
- height 311, 314, 347, 418, 426, 427
- HELVETICA 353
- helvetica 308
- hemisphere 281
- Hermite 303
- hermitian 246
- Hessenberg 246
- Hessenberg Karl 250
- HessenbergForm 250
- hexahedron 281
- hidden 290
- Hilbert David 251
- HilbertMatrix 251
- hint 315
- homogeneous 302
- homogeneousB (C, D, G) 302
- homothety 281
- horizontal 322
- HorizontalGlue 309
- hscroll 309
- HTML 11, 366
- HUE 346, 385, 407, 417
- hybrid 253
- hyperbola 282
- hyperbolic 266
- hypercylindrical 288
- hypergeom 236
- icosahedron 282
- identity 246
- IdentityMatrix 251
- ifactor 237
- Image 314, 474
- image 314
- ImageTools 270
- implicit 248
- implicitplot 225, 267, 273, 480, 482
- in 453
- inequal 273
- infinity 166
- information 329
- initialdata 301



- inplace 254, 257, 261, 262
 InputDialog 331
 insequence 272, 354, 359, 364, 385
 inset 309
 Int 237
 int 454, 465
 integer 255
 interactive 273
 interactiveparams 274
 interface 228, 250, 368, 369, 402, 460
 intersect 234
 intersection 70
 intersectplot 275
 intfactor 301
 invcasscylindrical 288
 invcassinian 266
 invellcylindrical 288
 invelliptic 266
 inverse 459
 invoblspheroidal 288
 invyprospheroidal 288
 IsDefinite 251
 isolate 238
 solve 243
 IsOrthogonal 251
 IsSimilar 251
 italic 308
 Jacobi 303
 Jahne B. 270
 JordanBlockMatrix 251
 JordanForm 251
 jpg 314
 kernelopts 415
 Khaki 277
 khaki 266
 known 300
 Korkel S. 270
 Kovacic J. 301
 kovacicsols 301
 KroneckerProduct 252
 Kroneker L. 252
 Label 313
 labeldirections 289
 labelfont 266
 labels 266
 Lagerstrom 303
 Laguerre 303
 large 299
 LaTeX 11, 366
 LeastSquares 252
 left 278, 308, 314
 legend 267, 384, 470
 length 327, 399
 lexorder 399
 lhs 299
 light 289
 light1 275
 light2 412
 lightmodel 275, 413
 LimeGreen 277
 Limit 237
 limit 166
 linalg 9, 352, 359, 360, 459
 line 267, 282, 290, 299, 351
 linear 302, 465, 476
 linear_ODEs 303
 linear_sym 303
 LinearAlgebra 9, 345, 352, 400, 479
 linearsol 301
 LinearSolve 253
 LIFESTYLE 291
 linestyle 267, 384
 Liouville 303
 list 416, 444
 ListBox 326
 ListBoxSplit 327
 listplot 275
 ListTools 401
 ln 236
 local 356, 416, 440, 480
 location 122, 342
 logarithmic 266
 logcosh 266
 logcoshcylindrical 288
 logcylindrical 288
 loglogplot 275
 logplot 275
 longdash 267, 341
 lower 321
 lsode 295
 LU 253, 255
 LUdecomposition 252
 LyapunovSolve 252
 macro 351
 magenta 266
 majorticks 321



- Map 254
- map 232, 358, 443, 444, 483
- Map2 254
- Maple 11 355
- Maple 15 304
- Maple 9 491
- Maple V Release 4 228
- Maplets:-Display 427
- Maplets:-Elements 426
- Maplets[Examples] 328
- Maplets[Tools] 416, 424, 453
- maroon 266
- Mathcad 436, 491, 493
- Mathematica 28, 391, 491
- MathML 325, 373
- MathMLEditor 325
- MathMLViewer 325
- MATLAB 491
- Matrix 242, 400, 403
- matrix 239, 396
- Matrix Builder 253
- Matrix shape incompatible 253
- MatrixAdd 254
- MatrixBuilder 243
- matrixDE 301
- MatrixExponential 254
- MatrixFunction 255
- MatrixInverse 255
- MatrixMatrixMultiply 255
- MatrixNorm 255, 443
- matrixplot 275
- MatrixPower 256
- MatrixScalarMultiply 256
- MatrixVectorMultiply 256
- matrixview 278
- max 444
- maxdigits 415
- maxwell 266
- maxwellicylindrical 288
- mebdfi 295
- MediaShell 8
- medium 299
- mediumfill 299
- MediumOrchid 277
- Menu 311
- MenuBar 311
- menubar 311, 427
- MenuItem 311, 427
- MessageDialog 329
- method 253
- MinimalPolynomial 256
- minimize 122, 342
- Minor 257
- minorticks 321
- minus 453, 477, 478
- missing 303
- mnemonic key 311
- mod 444
- modular 253
- module() 424
- Mohr O. 118
- Monospaced 308
- yandex.ru 8
- msolve 243
- mtaylor 378
- muchange 302
- Multiply 257
- mutest 302
- navy 266
- never 309, 310
- No Annotation (режим) 369
- NONE 367, 412, 442
- none 255, 265, 289, 299, 346, 351
- nops 234, 343, 345, 361, 416
- Norm 353, 355, 400
- norm 359, 360
- normal 265, 289, 353
- Normalize 257, 350, 403, 405
- normalized 249
- NullSpace 257
- numer 480
- numeric 234, 375, 384
- numpoints 267, 289, 384, 480, 482
- oblatespheroidal 288
- OBLIQUE 265
- octahedron 282
- ODE Analyzer 296
- ode_y1 302
- odeadvisor 296, 302
- odeplot 275, 375, 384
- onapprove 329, 331
- onchange 315, 317
- onclick 311, 427
- op 327, 409, 443
- OpenGL, рендерер 287
- operator 375, 384
- optionsexcluded 273

- optionsfeasible 273
- orange 266
- Order 378, 380, 382
- orientation 281, 289, 322, 353, 413
- orthogonal 289
- OuterProductMatrix 257
- output 375, 384
- Painleve 303
- parabolic 266
- paraboloidal 288
- paracylindrical 288
- parallelepiped 282
- pareto 276
- parse 331, 332, 433, 448
- Pascal 11, 231
- PATCH 385
- patch 290
- patchcontour 290
- patchnograd 290
- patmatch 377
- patterns 303
- pause 323
- PDEtools 294, 375, 383, 420, 477
- Permanent 257
- Phrase (режим) 369
- Pi 355, 408
- pi 356
- piecewise 421, 425, 454, 465, 469
- pieslice 282
- pink 266
- plain 329
- PLOT 291, 346, 356, 416
- Plot Builder 274
- Plot Color Names 277
- PLOT3D 293
- plot3d 265, 275, 412
- plot3d/option 275
- plotcompare 276
- plots 345, 352, 366, 482
- Plotter 323, 348, 418, 427, 471
- plottools 279, 345
- plum 266
- png 314
- POINT 292
- point 265, 267, 282, 290, 308
- point, geometry 70
- pointplot 276
- pointplot3d 276
- POINTS 292, 385
- Poisson Simeon 134
- polar 236, 266
- polarplot 276
- polygon 267, 282
- polygonoutline 267
- polygonplot 277, 393, 396
- polygonplot3d 277
- POLYGONS 291, 385, 407
- polyhedra_supported() 277
- polyhedraplot 277
- polynom 255
- PolynomialInterpolation 476
- popupmenu 316
- posint 453
- powsubs 377
- Prandtl Ludwig 411
- Preference (меню) 369
- prime 294, 300, 483
- print 230
- printf 433, 442
- proc 232, 233, 346, 356, 440, 480
- project 283
- projection 289
- prolatespheroidal 288
- prompt 228
- pseudo 255
- QR 253
- QR-разложение матрицы 257
- QRDecomposition 257
- quadrature 303
- QuestionDialog 335
- quit 324
- R98, рендерер 287
- radical 236
- RadioButton 311, 322, 426, 474
- RadioButtonMenuItem 311
- rand 258
- RandomMatrix 258
- RandomVector 258
- Rank 258
- rational 255, 303
- RationalCanonicalForm 249, 258
- READ 240
- Read 270
- read 363, 366, 403, 433
- readbytes 334
- readdata 349



- readline 433
real 406
rectangle 283
Red 277
red 266, 346, 353, 404
redode 303
reducible 303
reference 312
reflect 283
resizable 312
resolution 267
restart 229, 338
resultant 260
return 417
RGB 385
RGBtoGray 270
rgf_findrecur 122
rhs 299, 380
Riccati 303
riccatisol 303
right 278, 308, 314
rkf45 295
rkf45_dae 295
rose 266
rosecylindrical 288
rosenbrock 295
rosenbrock_dae 295
Rotate 401
rotate 283, 294
Row 258
RowDimension 258
RowOperation 258
RowSpace 258
rsolve 123, 243
rtable 246
RTF 11
RunDialog 331, 332, 334
RunWindow 311, 329, 427
same_box 276
SansSerif 308
save 357
scalar 246
ScalarMatrix 259
ScalarVector 259
scale 284
scaling 267, 346, 353, 359, 385, 445
scan 400
Scharr H. 270
semilogplot 277
semitorus 284
separable 303
seq 346
series 382
Serif 308
Set 396
setcolors 277
SetOption 310, 329, 454
setoptions 277
shading 289, 351, 412
shape 400
showassumed 236, 368, 369
showlabels 322
showticks 322
Shutdown 313, 329, 348, 427, 455
sienna 266
simplify 480, 482
sin 363
SingularValues 259
sixsphere 288
size 300, 308
Slider 321
small 299
smalltwo 299
snapticks 322
Sockets 270
solid 267, 342
solidbox 265
soliddiamond 265
solidsphere 290
solve 11, 253, 339, 367, 479
sort 238, 327, 399, 401
spacecurve 278
spacedash 267, 341
spacedot 267, 341
spacing 309, 397
SparseDirect 253
SparseIterative 253
SparseLU 253
sparsematrixplot 278
sphere 284, 290
spherical 288
spline 465, 476
sprintf 317
sqrt 178, 232, 236, 356, 357
SquareAntiprism 277
Standard Worksheet 227, 243, 253, 274
stellate 284



- stop 324
- StronglyConnectedBlocks 259
- Student 243
- STYLE 385
- style 264, 289, 351
- Stylize 269
- SubMatrix 259, 270
- subs 238, 253, 255, 377, 480, 482
- subsindent 377
- subsop 377
- SubVector 260
- Sum 237
- sum 121, 367, 412
- surface 290
- surfacecontour 290
- surfacewireframe 290
- surfdata 278
- swf 363
- SylvesterMatrix 260
- SylvesterSolve 260
- sym_Fx 303
- sym_implicit 303
- SYMBOL 265, 292, 385
- Symbol 308
- symbol 282, 290, 346, 417, 442
- symbolsize 267, 278, 282, 290
- symmetric 246, 400
- Table 320
- TableHeader 321
- TableItem 321
- TableRow 321
- tan 232, 266
- tangen 266
- tangencylindrical 288
- tangentsphere 288
- taylor 378
- taylorseries 295
- terminal 239
- Tetmajer L. 213
- tetrahedron 285
- TEXT 292, 346, 358
- TextBox 318, 334, 347, 427
- TextField 310, 315, 347, 418, 426
- textplot 278
- textplot3d 279
- then 481
- THICKNESS 291, 404, 416
- thickness 267, 355, 364, 384, 482
- tickmarks 267, 290, 353
- time 412
- TIMES 292, 345, 416
- Times 308
- Titchmarsh 303
- title 268, 312, 359, 367, 445
- titlefont 268
- to_end 323
- to_start 323
- ToeplitzMatrix 260
- ToggleButton 311, 319
- ToolBar 312
- toolbar 312
- ToolBarSeparator 312
- Tools 243, 253, 274, 296
- tooltip 347, 426
- top 309, 315
- toroida 288
- torus 285
- Trace 261
- Trace Contour 269
- Trailing Tildes (режим) 369
- transform 285
- translate 285, 294
- transparency 290
- Transpose 261, 270, 351
- triangular 246, 400
- TriangularPrism 277
- TridiagonalForm 261
- trig 374
- trigsub 377
- true 231, 364
- tubeplot 279
- tubepoints 279
- turquoise 266
- Tutors 243, 253
- type 329, 375, 384
- unapply 234
- UNCONSTRAINED 445
- uneval 233
- union 234
- UnitVector 261
- univar 255
- UNIX 226
- unprotect 362
- upper 322, 400
- use 347, 424, 453
- Utilities 328

- valign 309, 315
value 317, 347, 426
Van_der_Pol 303
VandermondeMatrix 261
VariationalCalculus 374
Vector 242, 403
vector 243, 358, 360
VectorAdd 261
VectorAngle 262
VectorMatrixMultiply 262
VectorNorm 262, 352, 403
VectorScalarMultiply 262
vertical 322
VerticalGlue 309
view 268
violet 266
visible 309, 315
vrml 285
vscroll 310
vuz.exponenta.ru 28, 228, 301, 304,
349, 363, 416
warning 329
whattype 327, 423
wheat 266
white 266
Width 270
width 311, 315, 347, 353, 418, 426
Window 310
WinEdt 306
wireframe 290
wireframeopaque 290
with 345
WRITE 327
Write 271
writedata 239, 349
wrl 285
www.cortona3d.com 286
xcoord 313
xy, xyz 289
ycoord 313
Yellow 396
z 289
zero 246
ZeroMatrix, ZeroVector 262
zgrayscale 289
zhue 289
Zip 262
Абель Н. (Abel Niels) 297
Автономность 297
Аксиома статики 44
Активные силы 89
Алгебраические функции 301
Алгоритм
— J.Kovacic 301
— Горячева В., Резунова А. 343
Алюминий 155
Амперсанд & 311
Амплитуда колебаний 119
Аннулирующий полином 256
Антипризма 277
Арка 206, 467
— круговая 206
— параболическая 205
Архимед 272
Базисные вектора 245, 258
Бернулли Д. (Daniel Bernoulli) 297
Бетти Э. (Betti Enrico) 118
Бимомент 420
Бимомент инерции 143
Бояршинов С. В. 149
Браузер 286
Вандермонд А. (Vandermonde A.) 261
Ввод и вывод 239
Вейерштрасс К. (Weierstrass K.) 256
Вектор
— нулевой 262
— свободный 10
— скользящий 10
— случайный 258
— строка, столбец 243
Векторное произведение 247, 358
Виртуальное перемещение 89
Власов В.З. 140
Внешнее произведение векторов 257
Временная нагрузка 189
Время
— критическое 216
— на «размышление» 296
— счета 363, 412, 479
ГПа (гигапаскаль) 136, 412
Гамильтон У. (Hamilton W.) 116, 256
Ганкель Г. (Hankel H.) 249
Гаусс К. Ф. (Gauss C. F.) 244, 252, 258
Геннеберг Э. (Henneberg E.) 38
Гибкость 211, 474
Гильберт Давид (Hilbert David) 251

- Гипербола 275, 282
 Гипотеза плоских сечений 221
 Гипотезы прочности 161
 Главная диагональ 247
 Главные
 — деформации 164
 — касательные напряжения 164
 Главный
 — вектор 47, 353
 — момент 48, 353
 — момент инерции 144
 Глобальная переменная 477
 Горячие клавиши 311
 Грам Й. (Gram J.) 249
 Граф 55
 Граф (процедура) 361, 379
 Граф двудольный 257
 Графические структуры 382
 Графический редактор 269
 Грин Дж. (George Green) 134
 Двадцатигранник 282
 Двенадцатигранник 277
 Движение
 — вращательное 50, 86
 — плоское 50, 86
 — поступательное 50, 86
 Двусвязная фигура 395
 Двухместная операция 262
 Двухшарнирная арка 206
 Делимость 256
 Демонстрационная версия Maple 228
 Депланация 140
 Дерево 215, 476, 490
 Деформации главные 164
 Диагональные блоки 259
 Диаграмма
 — Максвелла–Кремоны 35
 — Парето 276
 Динама 49
 Дитто операция 233, 483
 Дифф (процедура) 378, 383
 Дифференциальный оператор 298
 Додекаэдр 277, 280
 Дуга эллипса 281
 Дюралюминий 490
 Евклидова норма матрицы 256
 Единичная матрица 245
 Жесткость 118, 191
 Жесткость на изгиб 432
 Жорданова форма 251
 Жуковский Н. Е. 77
 Задача Коши 381, 384
 Заделка скользящая 431
 Закон
 — Гука 164, 167, 172, 179, 222
 — парности 158
 — сухого трения 32
 Замена переменной 300, 302
 Замок арки 202
 Зацепин М. Ф. 28, 493
 Зимина О. В. 249
 Знак
 — кривизны 181
 — момента 10, 49
 — момента инерции 132
 — угловых скоростей 67
 — умножения матриц 343
 Золотое сечение 125
 Идеальная связь 42
 Изгибно-крутильная характеристика 420
 Икосаэдр 282
 Импульс обобщенный 383, 386
 Инвариант
 — векторный 47
 — скалярный 48, 49
 Инварианты тензора напряжения 162
 Интеграл
 — Максвелла–Мора 195, 403
 — двойной 127
 — криволинейный 141, 391
 Интегрирование по контуру 390
 Интегрирующий множитель 301, 303
 Интерактивный оператор dsolve 295
 Интерфейс 11, 344, 354, 421, 472
 Инфиксная форма 247
 Канонические уравнения
 — динамики 117
 — метода сил 175
 Капустина О. М. 28
 Квазистатический процесс 216
 Кийко И. А. 276, 492, 493
 Кинематическая
 — изменяемость 197
 — неизменяемость 38
 — проверка 465

- Кинетическая энергия 119
 Кинетический потенциал 108, 112
 Кириллов А. И. 249, 492
 Клавиша
 — Ctrl 327
 — F3, F4 228
 — Shift 327
 — Пуск 354
 Ключ арки 202
 Ключников В. Д. 216
 Коммутативность 217
 Комплексная
 — область 243
 — плоскость 272
 Комплексное решение 477
 Конвертация
 — документа 366
 — программы 373
 Конкатенация матриц 349
 Константа Эйлера 412
 Константы формулы Ясинского 215
 Конус 275, 280
 Координаты
 — бисферические 288
 — декартовы 54
 — материальные 221
 — обобщенные 92
 — полярные 53
 — тороидальные 279
 Копирование матрицы 247
 Кориолис Г. (G. G. Coriolis) 76
 Коэффициент
 — α , β , γ 411
 — Пуассона 158, 161, 399
 — Сен-Венана 410
 — запаса прочности 158, 161
 — запаса устойчивости 215
 — податливости 119
 — поправочный 410
 — продольного изгиба 490
 — температурный α 179
 — трения качения δ 30
 — трения скольжения f 32
 — формы 476
 Коэффициенты
 — Сен-Венана 413
 — канонических уравнений 463
 Кратность шарнира 196, 439
 Крэмона А. (Cremona A.) 35
 Кривые неустойчивости 220, 225
 Критическое напряжение 474
 Кронекер Л. 252
 Круговая перестановка 401
 Куб 281
 Кулон Ш. (Coulomb C. A.) 32
 Кутта М. В. (Kutta M. W.) 295
 Кэли А. (Cayley A.) 256
 Лагранж Ж. (Lagrange J.) 91, 118
 Лаплас П. (Laplace P.) 300
 Латунь 179
 Лемандр А. (Legendre A.) 115
 Лемниската 264
 Линеаризация уравнения 217
 Линейная комбинация
 — векторов 261
 — матриц 254
 — столбцов 245
 — строк 258
 Линейная упругость 222
 Линия
 — влияния 189, 450
 — цепная 206
 Лиувилль 301
 Логарифмическая спираль 264
 Ляпунов А. М. 252
 МЦС 41, 56, 60, 66, 70, 72, 91, 101
 Максвелл Д. (Maxwell J.) 35, 118, 168
 Мартынова Е. Д. 263, 493
 Матрица
 — Вандермонда 261
 — Ганкеля 249, 261
 — Гильберта 251
 — Жорданова 251
 — Фробениуса 248, 258
 — Хессенберга 250
 — блочно-диагональная 251
 — верхняя треугольная 244, 252, 257
 — взаимная 244
 — диагональная 248
 — единичная 245
 — заполнение 278
 — квазидиагональная 248
 — ленточная 244
 — неустойчивости 477
 — нижняя треугольная 252
 — нулевая 262

- обратная 244, 255, 343, 459, 464
- ортогональная 250, 251, 257
- перестановок 252
- поворота 249, 400
- подобная 251
- положительно определенная 251
- почти треугольная 250
- присоединенная 244
- псевдообратная 255
- расширенная 249, 253
- симметричная 400
- системы уравнений 249
- слияния 349
- случайная 258
- смежности 257
- сопровождающая 246
- союзная 244
- теплицева 250, 260
- транспонированная 261
- треугольная 246
- трехдиагональная 261
- характеристическая 245
- Матричная функция 254
- Мгновенный центр скоростей 56
- Медь 161
- Метод
 - Head and tail 73
 - Гаусса 244
 - Геннеберга 37, 38
 - МЦС 91
 - Ньютона 162
 - Риттера 34
 - геометрический 176
 - графов 55, 60
 - деформаций 176
 - замены стержней 37
 - индуктивный 122
 - интегрирования по контуру 391
 - исключения Гаусса 258
 - кинематический 41, 176, 404
 - координатный 57
 - обратной матрицы 242
 - последовательных приближений 213
 - сил 168, 176
- Минимальный полином 256
- Минимум 341
- Минор 257
- Многогранник 277
- Многопролетная балка 188
- Модуль
 - Юнга 134
 - сдвига 134, 135, 156, 407
 - упругости 118, 158, 399
 - стали 195
- Модульная арифметика 243
- Момент
 - инерции 222
 - секториально-линейный 143
 - секториально-статический 143
 - сопротивления
 - полярный 136
 - при кручении 408
 - статический 221
 - трения качения 29, 31
- Момент инерции
 - главный 129, 133, 144, 150
 - двутавра 487
 - обобщенный 98
 - обода 104, 112
 - полярный 135, 138
 - секториальный 143
 - стержня 92
 - трапеции 486
 - треугольника 486
 - уголка 489
 - центральный 130
 - центробежный 127, 129, 392
 - цилиндра 86, 99
 - четверти круга 486
 - швеллера 488
- Момент силы
 - относительно точки 9
 - относительно оси 10
- Моноцикл 103
- Мор О. Х. (Mohr O.) 118
- Мощность 92, 94, 136
- Нагрузка
 - временная 189
 - постоянная 189
- Наддиагональ 247
- Направляющие косинусы 163
- Напряжение
 - максимальное касательное 135, 155
 - октаэдрическое 401
- Напряжения

- главные 401
- главные касательные 401
- допускаемые 490
- Насыщенность (saturation) 331
- Неподвижный шарнир 11
- Николаи Е. Л. 123
- Новожилов И. В. 493
- Норма
 - вектора 262, 400
 - евклидова 256
- Нормальная форма матрицы 258
- Нормальное сопротивление 158
- Обобщенные
 - импульсы 115, 383, 386
 - координаты 92
 - силы 386
 - скорости 114, 377, 383
- Объединение множеств 234
- Октаэдр 282, 283
- Оператор
 - выбора 231
 - вывода 230
 - вычитания множеств 453
 - минимизации 341
 - снятия скобок 341
 - суммирования add 234
 - условный 231
 - цикла 230
- Операция
 - отражения 283
 - параллельного переноса 285
 - поворота 283
 - растяжения (сжатия) 284
- Определитель матрицы 247
- Ортогонализация Грама–Шмидта 249
- Открытый профиль 154
- Относительное изменение объема 164
- Относительный угол закручивания 155
- Очков В. Ф. 493
- Па 407
- Паллогграф 123
- Пара сил 10
- Параболоид
 - вращения 283
- Параметр волнообразования 222
- Парето В. (Pareto V.) 276
- Паросочетание 257
- Пауза 323
- Перевалов В. (EPURA.swf) 180
- Переменная окружения 298
- Пересечение
 - множеств 234
 - поверхностей 275
- Перманент 257
- Плагин 286
- План скоростей 67
- Плечо 9, 47
- Площадь
 - главная секториальная 143
 - параболической эюры 199
 - фигуры 391
- Поддиагональ 247
- Подматрица 260
- Позиционная координата 381
- Поле направлений 299
- Полином
 - минимальный 256
 - характеристический 245, 256
- Положение эюры моментов 443
- Полоса прокрутки 320, 334
- Полусфера 281
- Полюс 55
- Поля
 - внешние 36
 - внутренние 36
- Порядок
 - матрицы 246
 - особой точки 478
 - остаточного члена 378
 - рекуррентного уравнения 122
- Постоянная нагрузка 189
- Потапов В. Д. 38, 457, 467, 493
- Потенциальная энергия 112, 383
- Пояс
 - верхний 119
 - нижний 119
- Правило
 - Верещагина 142, 199, 415
 - Жуковского 77
 - дождя 181
 - знаков 162, 193
 - перерезывающие силы 429
- Прандтль 411
- Предел 166
- Преобразование
 - Лежандра 115

- подобия 281
- Приведенные массы 371
- Принцип возможных
 - перемещений 89
 - скоростей 41
- Проверка
 - деформационная 201
 - кинематическая 465
 - ортогональности 251
 - подобия 251
 - положительной определенности 251
 - статическая 201
 - типа 327
- Проекция
 - силы 9
 - фигуры 283
- Произведение
 - Кронекера 252
 - векторное 247, 358
 - внешнее 257
 - матриц 257
 - скалярное 248, 359
- Пролет фермы 121
- Профиль замкнутый 155
- Процедура 232, 256
- Процедуры для рисунков 354
- Прочность материала 161
- Прямоугольник 283
- Псевдобиуркация 216
- Пуассон С. (Poisson S.) 134, 158, 411
- Пята арки 202
- Пятиугольник 277
- Работнов Ю. Н. 225
- Радиус
 - вектор 89
 - инерции 86
 - полярный 53
- Разложение Холецкого 252
- Разложение на множители 121, 237
- Размер
 - графа 360
 - окна 311
 - символа 290
 - файла 270
 - шрифта 308
- Ранг матрицы 258
- Раскосы 119
- Распор 207
- Распорные конструкции 202
- Растяжение (сжатие) 281
- Раус Э. Д. (Routh E. J.) 113
- Редактор формул
 - MathMLEditor 325
 - WinEdt 306
- Режим командной строки 324
- Результат 260
- Рекуррентное уравнение 122
- Рекуррентные формулы 225
- Рендерер 287
- Решение
 - в целых числах 243
 - системы уравнений 11, 242
 - точное 303
 - тривиальное 301
- Ривальс 72
- Ригель 456
- Риккати Я. Ф. (Riccati J. F.) 303
- Риттер А. (August Ritter) 34
- Рунге К. Д. (Runge D.) 295
- Ряд Тейлора 378
- Свечение краев 269
- Связь
 - идеальная 89
 - неудерживающая 28
 - стационарная 92
- Седов Л. И. 180
- Сектор 282
- Семейства шрифтов 308
- Сен-Венан (Sen-Venan V.) 410
- Силовой винт 49
- Сильвестр Д. Д. (Sylvester J. J.) 260
- Сильно связанные блоки 259
- Символ
 - \ (backslash) 312
 - cross, circle и др. 290
 - блока 229
 - одиночный (печать) 240
 - переноса 240, 333
 - подчеркивания 298
 - размер 267, 290
- Симпсон 142
- Сингулярные числа матрицы 259
- Система
 - линейных уравнений 249
 - основная 168

- эквивалентная 168
- Скалярное произведение 248
- Скользкая заделка 11, 15, 431, 437
- Скорость
 - при плоском движении 50
 - угловая 50
 - центра масс 86
 - циклическая 115
- След матрицы 261
- Слияние матриц 349
- Собственные
 - векторы 248
 - значения 119, 120, 399
 - частоты 118
 - числа 246, 248, 259
- Сортамент 487
- Спектральная норма матрицы 256
- Спираль
 - Архимеда 272
 - логарифмическая 264
- Сплав алюминиевый 155
- Сплайны кубические 476
- Сравнение по модулю 243
- Сталь 215, 476, 490
- Сталь, модуль сдвига 135
- Статический момент 391
- Стилизация 269
- Стиль линии 341
- Строка матрицы 258
- Суммирование в Maple 121
- Сфера 284
- Сферическая опора 45
- Тангенциальное ускорение 358
- Тейлор Б. (Taylor B.) 378
- Теорема
 - Гамильтона–Кэли 256
 - Кориолиса 76
 - взаимности Бетти 118
 - о проекциях векторов скоростей 57, 72
 - о трех силах 43
 - трапеции 59
- Теплиц Отто (Toeplitz Otto) 260
- Тетмайер Л. (Tetmajer L.) 213
- Тетраэдр 285
- Тимошенко С. П. 411
- Толщина линии 341
- Тор 285
- Точка
 - моментная 404
 - неустойчивости 218
 - приведения 48
- Точность 250, 382, 401, 415
- Трехшарнирная арка 202
- Труба 280
- Угол
 - закручивания
 - абсолютный 136
 - относительный 134, 155
 - полярный 53
- Угол между векторами 262
- Удельное изменение объема 160
- Умножение матриц 255
- Умножение матрицы на вектор 464
- Уравнение
 - Абеля 297
 - Бернулли 297
 - Лапласа 300
 - Ляпунова 252, 260
 - Пуассона 411
 - Риккати 243, 303
 - Сильвестра 260
 - Чини 297
 - в частных производных 219, 300, 301, 481
 - кубическое 162
 - линеаризованное 216
 - матричное 252, 260
 - рекуррентное 122
 - совместности 171
- Уравнения
 - Гамильтона 117, 301
 - Лагранжа 118
 - Лагранжа 2-го рода 108
 - канонические
 - динамики 117
 - метода сил 175
 - связи 125
 - трех угловых скоростей 73
 - трех угловых ускорений 59, 366
- Ускорение
 - Кориолиса 76
 - нормальное 54, 358
 - при плоском движении 55
 - тангенциальное 54
 - угловое 50



- Условие неустойчивости 216, 482
 Условный оператор 444
 Устойчивость 472
 Фаза колебаний начальная 119
 Ферма 32, 35, 37, 39, 41, 43, 164, 342
 Фильтры Собеля и Шарра 270
 Фихтенгольц Г. М. 391
 Флаг вывода 240
 Форма
 — Жорданова 251
 — инфиксная 247
 — первая нормальная 248
 Формат вывода 240, 327, 442
 Формула
 — Грина 134, 391
 — Кулона 32
 — Максвелла–Мора 118, 168, 199
 — Ривальса 55, 72
 — Симпсона 142
 — Тетмайера 213
 — Эйлера 222
 — трапеций 208
 — ускорения при вращении 72
 Фробениус Ф. Г. (Frobenius F. G.) 256
 Функция 234
 — Гамильтона 117
 — Лагранжа 108, 112, 114, 117, 374
 — Рауса 115
 — алгебраическая 301
 — знака 368
 — комплекснозначная 276
 — кусочно-заданная 423
 — кусочно-линейная 264
 — напряжений 411
 — ошибка 263
 Характеристический полином 245
 Хессенберг К. (Hessenberg K.) 250
 Холецкий А.-Л. (Cholesky A.-L.) 252
 Цвет
 — кривых графика 277
 — полный список 277
 — фона 307
 — шрифта 314
 Цветовая модель 280
 Цветовой режим 382
 Центральная винтовая ось 49
 Цепная линия 206
 Циклический интеграл 115
 Циклоида 264
 Частота
 — кадров 287
 — малых колебаний 113, 125
 Чини 297
 Число
 — статической неопределимости 196
 — вершин графа 360
 — вещественное 232
 — дуг 360
 — знаков вывода 402
 — обусловленности матрицы 246
 — паросочетаний 257
 — полуволн 222
 — сингулярное матрицы 259
 — собственное 248
 — стержней и узлов фермы 45
 — столбцов матрицы 245
 — строк матрицы 258
 — угловых точек 396
 Чтение файла 241, 332
 Чугун 215, 476, 490
 Шаг винта 49, 353
 Шестериков С. А. 225
 Шлик О. 123
 Шмидт Э. (Erhard Schmidt) 249
 Шрифт 265, 308
 Штрих (производная) 483
 Эйлер Л. (Euler L.) 223, 473
 Эллипс 281, 284, 410
 Эллипс инерции 134
 Энергия кинетическая 119
 Эпюра
 — касательных напряжений 135
 — криволинейная 142
 — моментов 136
 — в балке 181
 — в раме 458
 — параболическая 199
 — перерезывающих сил 182, 197, 428, 443, 450
 — процедура построения 415, 421
 — секториальных площадей 142
 Ядро матрицы 257
 Яркость (brightness) 331
 Ясинский Ф. С. 211

