2015 15th International Conference on Control, Automation and Systems (ICCAS 2015)
Oct. 13-16, 2015 in BEXCO, Busan, Korea

Efficient Algorithm for Evacuation Problem Solving

Amirgaliyev Yedilkhan[1], Kovalenko Alexey [2], Kalizhanova Aliya[3],

Kozbakova Ainur[4], Amirgaliyeva Zhazira[5]

[1]Institute of Information and Computing technologies, Kazakhstan, amir_ed@mail.ru,
[2]Samara State University, Russia, alexey.gavrilovich.kovalenko@rambler.ru,
[3]Kazakh National Technical University after K.I.Satpaev, Kazakhstan, kalizhanova_aliya@mail.ru,
[4]Kazakh National Technical University after K.I.Satpaev, Kazakhstan,ainur79@mail.ru
[53]Kazakh National University named after al-Farabi, zh.amirgaliyeva@gmail.com

Abstract: The article considers the problem of people evacuation of from the school, as well as mathematical models and methods for solving the problem of evacuation. To find the optimal solution of the problem of maximum flow in a network using game-theoretic approach and the various methods of optimization

Keywords: Model, graph, method, algorithm, maximum flow, evacuation.

1. INTRODUCTION

Evacuation is one of population protection means. It is taking out or withdrawal of people from hazardous areas. It could take place both in peacetime and wartime. Evacuation as a means of population protection used long time ago.

Actuality of evacuation as the means of population protection in wartime and peacetime during recent years not only decreased but also increased. Contemporary life experience says that the population increasingly runs into danger in the result of natural calamities, accidents and disasters in industry and transport. Think for instance of natural calamities, earthquakes, floods, snow slides, mud streams and earth falls, wild scale forest fires. In such cases, evacuation is usually unavoidable. Evacuation measures are taken at accidents at atomic power stations, emissions and flood of hazardous chemicals and biologically damaging substances, at vast fires at petrochemical and oil refineries.

In the posted task, we consider people evacuation from educational institution in the emergency situation. The main peculiarity of educational institution's buildings is instability of people distribution along internal premises connected with the lectures timetable. It requires assessing the lessons schedule with regard to organizing unobstructed movement of people. As announced earlier the topic herein is and will be acute as, unfortunately, emergencies happen increasingly frequently. To solve the given problem there used mathematical methods and model of people flows motion inside the building.

2. MATHEMATICAL STATEMENT OF THE PROBLEM

2.1 Algorithm of solving the task on evacuation

Let us suppose that an emergency has happened in an educational institution bringing to the necessity to evacuate people. There are 24 classrooms with 30 students in each, and 8 stair wells and 2 exits. It is necessary to calculate the time, speed and direction of students' evacuation from the educational institution. Let us specify a graph $G=<E, V, H>$, in which direction of every arc $v \epsilon V$ identifies direction of flow motion, flow capacity of each arc equals to dv. Auditoriums are in E vertexes multiple. There identified two vertexes "start" and 'end' in E vertexes multiple. Vertex 0 is the stream source, 35 flowing. For i from E there given 2 numbers: amount of people sitting there and amount of people rushing out of there per time unit. Arcs are corridors and stir wells between the nodes.

As every arc has limited flowing capacity, the check of existing permissible flows along with their search can be fulfilled by means of the task on maximum flow and solving it with Ford-Fulkerson algorithm [1].

In the task on maximum flow, the flow is passed from one initial vertex to one final. All arcs have prescribed flowing capacity. To arrange that type of the task let us add two dummy vertexes ii and kk. Let us connect ii with stream source i_0. Its flowing capacity equals to Flowing c is connected with arcs to the vertex kk. Capacity of those arcs is accordingly. We obtain the task on standard maximum flow and apply any known algorithm for its solving. If it turned out that maximum flow is less than then initial task of one layer and accordingly the whole task has no solution. In that case, the minimal cut is beyond additional arcs [2].

If it turned out that maximum flow equals to we obtain permissible flow, which is transferred to the state of equilibrium by invariant transformations.

Let us describe people's flow movement along a corridor and staircase by means of Grindshiels formula. Let us introduce following designations: L – network section length, T – time of moving along the section, x – flow having passed the road section for time unit, P – flow density, S – number of lanes, W – speed of flow, λ – average corridor length.

As defined, the density is $\rho=1/\lambda$. Assume W – student's speed, W_{max} – maximum speed. Time, which gets a man to travel a route section of λ length equals to $\tau=\lambda/v$. Amount of students per time unit will be equal to $\kappa=1/\tau$. Therefore, We'll consider that the flow speed and density are interconnected due to linear dependence (Grindshiels formula) [3].

Therefrom or Let us insert it, and obtain Obtained function is a parabola with branches downward directed, maximum is achieved at and accordingly

Thus, we obtained the magnitude of maximum flow, which can be passed through.

Let us insert instead of ρ the expression and receive the following formula

According to Viete formula we obtain taking into account that every member strives to maximize own speed.

From here we receive that the time of travelling along the network section is expressed with following dependence: where T_{min} – minimal travelling time along the section in case the flow along it equals to zero. Let us consider evacuation movement route. Based on investigation data the width can be accepted as 0,6 м, with supposition of its small reduction for the roads with the width in several flows. Apart from that, in view of necessity, irrespective of the road width, in case of possibility of occasional opposing traffic or overdrive at traffic delay, the path with width of one flow should be accepted with some width reserve. Considering this and the necessity of flow number at existing and adaptable evacuation routes, we can give a table for defining flow number per width both of horizontal route and of staircases.

Table 1. Determination of flows number

Number of elementary flows	Width of evacuation route		
	Normal	Minimal	Maximum
1	0,9	0,9	1,2
2	1,2	1,2	1,7
3	1, 8	1,7	2,3
4	2,4	2,3	3

In practice, mass movement speed fluctuates from 5 to 75 m per minute. At sustained motion, density cannot reach physically maximum amount, therefore it is rational to accept the length of the route as calculation basis. At that, speed specified values are defined for horizontal path as 16 meters per minute, for descent down staircase as 10 meters and for ascent 20% less, that is as 8 meters per minute [4].

Flowing capacity of elementary stream per minute is defined as fraction of speed division by flow density. Total capability is defined by multiplying the obtained value by flows number at route width and by number per minute, making up evacuation duration. It is evident hereof, that such product, depending on evacuation motion factors total, cannot be constant value, as it is recommended by existing norms, but it is, to a significant extent, a variable value, depending on local conditions and increasing proportionally to increase of evacuation permissible duration. Time allowance directly influences at permissible route length.

For the first stage, the route length characterizes ultimate moving away from the exits and has importance mainly for big buildings. For the sum of the first and second stages the norms herein determine laying out of separate floors in ratio of number and location of exits to outside or to the staircases. For the sum of three stages, the same norms influence at laying out in whole limiting number of floors, and prescribing premises grouping per floors in such a way, that the first and second stages could decrease in proportion to the increase of the third one [5].

2.2 Problem on maximum flow in the network

In many network problems, it is meaningful to consider the arcs as certain communication having definite flowing capacity. In this case, as a rule, there considered the task of some flow maximization, directed from the selected vertex (source) to some other vertex (outflow). Such type of task is called the problem of maximum flow.

Let us assume that there is an orient graph $G=\langle E,V,H\rangle$, in which direction of every arc $v\in V$ denotes the flow motion direction, flowing capacity of each arc equals to dv.

At vertexes of multiple E there distinguished two vertexes: start and end.

Vertex $н$ is the source of the flow, κ – is the outflow. It requires maximum flow, which can pass from vertex $н$ to κ.

Let us denote as x_v flow level passing along the arc v. It is obvious, that

$0\le x_v \le d_v$, $v\in V$ (1) In every vertex the incoming flow level equals to outgoing flow level. That is, following congruence is true

(2)

or

(3)

Accordingly to vertexes $н$ and κ there executed

(4)

(5)

Magnitude Q is value of the flow, outgoing from vertex $н$ and incoming into vertex κ.

Problem. Define:

$Q \to \max$ (6)

at delimitations (1) – (5).

Values (Q, xv , $v\in V$) satisfying delimitations (1) – (5) will be named as flow in the network, and if they maximize the magnitude Q, then as maximum flow. It is easy to see that values $Q=0$, $xv=0$, $v\in V$, is the flow in the network. Problem (1) – (5) is the task of linear programming and can be solved applying simplex algorithm.

Let us break multiple of vertex E into two nonintersecting parts $E1$ and $E2$ in such a way, that $н \in E1$, $к \in E2$. Crosscut $R(E1,E2)$, separating $н$ and $к$ we will name such multiple $R(E1,E2) \subset V$, that for every arc $v \in R(E1,E2)$ or $h1(v) \in E1$ and $h2(v) \in E2$, or $h1(v) \in E2$ and $h2(v) \in E1$.

Fig. 1. Search for crosscut

There is multiple $E1=\{1,4,7\}$ *on Fig.1*, these vertexes have dark filling. $E2=\{2,3,5,6,8,9\}$. Crosscut $R(E1,E2)$ represent arcs, which dotted line went through.

Let us break multiple $R(E1,E2)$ into two parts as follows:

$R+(E1,E2)=\{v \in R(E1,E2) | \ h1(v) \in E1 \text{ and } h2(v) \in E2\}$,

$R–(E1,E2)=\{v \in R(E1,E2) | \ h2(v) \in E1 \text{ and } h1(v) \in E2\}$.

Elements of the multiple $R+(E1,E2)$ we will name straight arcs, they lead from multiple $E1$ to $E2$. Elements of the multiple $R–(E1,E2)$ are backward arcs, they lead from multiple $E2$ to $E1$. Flow through the crosscut we will name the value

Crosscut flowing capacity we will name the value

It is obvious that $0 \le X(E1,E2) \le D(E1,E2)$. Next theorem is true.

Theorem 1. On maximum flow and minimal crosscut.

In any network the magnitude of maximum flow Q from the source $н$ to overflow $к$ equals the minimal flowing capacity $D(E1,E2)$ amongst all crosscuts $R(E1,E2)$, separating vertexes $н$ and $к$.

Crosscut $R(\bar{E}_1, \bar{E}_2)$, with $Q=D(\bar{E}_1,\bar{E}_2)$ we will name constraining. At constraining crosscut, there is executed

Let us assume, that $(Q, xv, v \in V)$ is a flow in the network, and succession $н=i0, v1, i1 , v2, i2,,vK, iK=к$ is a circuit connecting vertexes $н$ and $к$. Define on that circuit motion direction from vertex $н$ to $к$. Arc vj from that circuit is called straight, if its direction coincides with motion direction from $н$ to $к$, and backward, if not. Circuit will be called flow increasing circuit, if for straight arcs of the circuit v $(dv– xv)>0$ and for backward $xv>0$. Through the circuit thereof it is possible to pass additional flow q from $н$ to $к$ with value $q=min(q_1,q_2)$, where $q_1=min(dv – xv)$, minimum is taken from all straight arcs of the circuit, $q_2=min(xv)$, minimum is taken from all backward arcs of the circuit.

Theorem 2. Flow $(Q, xv, v \in V)$, is maximum, then and only then, there is no way to increase the flow. Offered algorithm for solving the problem of maximum flow in the network is based on searching an increasing flow in the circuit from $н$ to $к$. The search, in its turn, is based on the process of vertexes marks disposition similar to Dejkstra algorithm.

Let us add mark $P_i=[gi,vi,\theta]$ to every vertex i, where g_i – value of additional flow entered the vertex i, v_i – arc through which the flow entered, θ – sign «+», if the flow entered along the arc v_i , directed to i (along straight arc); θ – sign «–», if the flow entered along the arc v_i , directed from i (along backward arc),

Let us say that vertex i:

- is not labelled, if the additional flow does not reach it, the label will have the form $P_i=[0,–,\theta]$,
- is labelled, but not viewed, if the flow has reached it, but has not been allowed to go further, the label will have the form $P_i=[g_i,v_i,\theta]$, where $g_i>0$,
- labelled and viewed, if the flow reached it and allowed to go further, label will have the form $P_i=[g_i,v_i,\theta]$.

Let us consider solution algorithm.

0. For all $v \in V$ assume that $xv=0$, assume that $Q=0$.

1. All vertexes are unlabeled. Vertex $н$ is labelled, but not viewed with a label $P_н=[\infty,–,–]$. It means that the unlimited volume flow enters that vertex. 2. Search labelled but not viewed vertex. If it is not available, then the found flow $Q, xv, v \in V$ is maximum and algorithm completes its function. If such vertex is found, i – is its number, then pass on to 3.

3. View vertex i:

- for all assume $j=h_2(v)$. If vertex j is unlabeled and $(dv–xv)>0$, then mark it with label $P_j=[q,v,+]$, where $q=min(q_i, (dv–xv))$, if $j=к$, then pass on to point 4.
- for all assume $j=h_1(v)$. If vertex j is unlabeled and $xv>0$, then mark it with a label $P_j=[q,v,–]$, where $q=min(q_i, xv)$, if $j=к$, then pass on to point 4.
- label vertex i as viewed and pass on to point 2.

4. Pass additional flow. Let us assume that $j=к$, $q=gк$ and $v=vj$.

- if θ=«+», then it is necessary to fulfill: Let us assume that $xv=xv+q$, $i=h_1(v)$, if $i=н$, then pass on to point 1, otherwise put $j=i$ and pass on to $v=vj$),
- if θ=«–», then it is necessary to fulfill: Let us assume that $xv=xv–q$, $i=h_2(v)$, if $i=н$, then pass to point 1, otherwise put $j=i$ and pass on to $v=vj$).

Because of the algorithm execution there will be obtained the flow $(Q, xv, v \in V)$. To search the crosscut with minimal flowing capacity part of vertexes should be labelled and viewed at the final stage of algorithm operation in point 2, we include these vertexes into multiple $\bar{E}_1$, $\bar{E}_2=\bar{E} \setminus \bar{E}_1$. Cross cut $R(\bar{E}_1, \bar{E}_2)$ will be the sought for [7].

2.3 Method of potentials and criterion of optimality

Method of potentials for solving the problem thereof supposes, that at initial graph, in some way, there defined initial radix tree $G'=\langle E,V',H \rangle$, (lets name its arcs as basic), along which transportation is performed in such a way that:

We can use the following algorithm for defining values $x_v \in V'$.

0 – step. $E' := E$.

K –step. If $E' \neq \varnothing$, find the vertex for which $(|V'^{+}_{i}|+|V'^{-}_{i}|=1)$, that is, the vertex i is final at $G'=\langle E',V',H \rangle$.

If $|V'^{+}_{i}|=1$, then for the arc $v \in V'^{+}_{i}$ perform: $x_v:=b_i$, If $|V'^{-}_i|=1$, then for the arc $v\in V'^{-}_i$ perform: $x_v:=-b_i$,

Move to the next step. We can obtain an infeasible solution in the result of algorithm operation, that is, for some $v\in V$ $x_v<0$. In that case, initial radix tree shall be changed. However, it does not guarantee that we will obtain feasible solution for it. Further there will be described the algorithm allowing either define basic solution or demonstrate that it does not exist.

Let us consider that initial radix tree G' is found and for $v\in V'$ defined $x_v>0$. In order to determine whether the obtained solution is optimal, we will make use of optimality criterion.

Let x_v, $v\in V$ is such solution of the problem that for $v\in V\backslash V'$ $x_v=0$ and for $v\in V'$ $x_v\geq 0$. The solution herein is optimal, then and only then, when there exists numbers u_i, $i\in E$, called potentials, such, that

(7)

(8)

For potentials calculation there applied next algorithm.

0–step. For certain (only one) vertex $i\in E$ we assume $u_i:=0$.

k–step. Find arc $v\in V'$, for which potential is known, of only one of its vertexes. If there is no such arc, it is the end of operation, otherwise, using dependence we define potential in the vertex, in which it is unknown, and pass on to step$(k+1)$.

Let us consider next algorithm.

1. Amongst all arcs $v\in V\backslash V'$ we search an arc v_0 such that ;

2. If there is no such arc, then initial problem is solved, otherwise it is needed to accomplish algorithm of transfer to a new radix tree.

$V':=V'\cup\{v_0\}$, where v_0 the arc, found in previous algorithm. Now graph $G'=\langle E,V',H\rangle$ contains exactly one cycle $G''=\langle E'',V'',H\rangle$, at that, $v_0\in V'$. At subgraph G'' we define girdle direction coinciding with direction of arc v_0, and pass along subgraph G'' an additional flow in girdle direction of value θ. Every arc $v\in V''$ is ascribed a symbol «$+\theta$», if direction of arc v coincides with girdle direction, and a symbol «$-\theta$», if not. Assume that θ=min x_v amongst $v\in V''$, which are ascribed a symbol «$-\theta$».

Assume that , for all $v\in V''\backslash\{v_0\}$:

$x_v:=x_v+\theta$, if an arc v is ascribed a sign $+\theta$,

$x_v:=x_v-\theta$, if an arc v is ascribed a sign $-\theta$.

From multiple $V'\backslash\{v_0\}$ we exclude the arc, for which $x_v=0$. If there are several such arcs (degenerate case), we remove only one, in such a way, that graph connectivity $G'=\langle E,V',H\rangle$ was not broken.

For obtained solution we over again calculate potentials with the second algorithm and study it, concerning optimality with the third algorithm, while optimal solution is found.

Let us consider the algorithm of initial radix tree search. Upon initial radix tree search there is applied described above method for the next "toy" problem.

Construct graph $G=\langle E,V,H\rangle$, in which $E=E\cup\{i_0\}$, where i_0 – additional value, $V=V\cup V_1\cup V_2$, где V_1 – multiple of additional arcs, directed from vertexes – points of production to additional vertex i_0, and V_2 – multiple of additional arcs, directed from i_0 intermediate vertexes and vertexes–points of consumption:

- for $v\in V$ assume c_v=0,
- for $v\in V_1$ assume c_v=1,
- for $v\in V_2$ assume c_v=1.

Assume . Initial radix tree is subgraph $G=\langle E,V_1\cup V_2, H\rangle$. If in the result of solving this problem it turned out, that optimal value of a functional strictly greater than zero, then the initial problem has no solution, otherwise, there will be obtained radix tree of the initial problem[8].

2.4 Ford and Fulkerson algorithm

Let us assume that some permissible flow has been already found. Let us ask two questions: how, having permissible flow, to define, whether it is optimal, and how to obtain permissible flow greater by value if the permissible flow thereof is not optimal.

For that purpose, it is necessary to identify, what of given below properties owns every arc in the circuit. For the first, the flow along an arc (i, j) is less than flowing capacity of an arc, (i, j), which naturally means that the flow along the arc can be increased. Let us denote multiple of such arcs in the circuit as i. For the second the flow along an arc (i, j) is positive, which means that it can be reduced. Let us denote the multiple of such arcs as R. Let us describe the procedure of Ford and Fulkerson method for labels disposition to construct the greater flow.

Step 1. Assign label to the source (vertex 1).

Step 2. Assign other labels to the vertexes and arcs proceeding from the next rules. If vertex x has a label, and vertex y has no mark and the arc $(x ,y)\epsilon I$, then label the arc (x ,y) and vertex y. In this case, the arc (x, y) is the straight direction arc. If vertex x has a label, and vertex y is unmarked and the arc $(y, x)\ \epsilon R$, then label the arc(y, x) and vertex y. In this case, the arc(y, x) is a backward direction one.

Step 3. Continue procedure of labels disposition until the outflow is labeled, or there are no unlabeled arcs left.

If in case of the given procedure implementation the outflow turned out to be labelled, we can say that there exists sequence of labelled arcs (name it C) from the source to outflow. Changing arcs flows entering C, we can construct the flow of greater value comparing to initial. In order to be sure, let us consider two cases: succession C contains only arcs of straight direction and succession C contains both straight and backward direction arcs.

In every case we can say how to obtain the flow of greater value comparing to the given one.

Let us consider case 1. Let $i\ (x, y)$ – a maximum value, the flow along the arc can be increased without violation of delimitation on flowing capacity. Assume that

Then $k>0$. In order to modify the flow upwards, let us increase values of flows on all arcs from C per value k. In this case, not a single delimitation of flowing capacity will be violated. It is easy to note, the flow preservation conditions for all vertexes will be satisfied. It follows that a new flow, on the one hand is permissible, and on the other hand, it has the value for k greater than the initial one.

Let us consider case 2. In this case, succession C contains both straight direction and backwards direction arcs. Let $r\ (x, y)$ – maximum value, the flow can be decreased along the arc (x, y). Assume that

Both values k_1 and k_2, and, accordingly, $min\ (k_1, k_2)>0$. In order to modify the flow upwards, let us increase flows values along all straight direction arcs from C for the value $min\ (k_1, k_2)$, and at all backward directions arcs from C decrease for the same value $min\ (k_1, k_2)$. In this case, not a single delimitation of flowing capacity will be violated. It is easy to note, the flow preservation conditions for all vertexes will be satisfied as well. Accordingly, a new flow, on the one hand, is permissible; on the other hand, it has the value less for $min\ (k_1, k_2)$ comparing to initial one.

If outflow cannot be labeled, it means that the flow is maximum. To ground this consideration, let us study the crosscut notion.

Let us select any multiple V, containing an outflow, but without the source. Then multiple of arcs (x, y), for which x does not belong to V, and $y \in V$ is called a circuit crosscut. In other words, crosscut is multiple of arcs, excluding which out of the circuit we would separate the source from the outflow. Crosscut value is the sum of flowing capacities of the arcs entering the crosscut. Crosscut is multiple of arcs removal, which brings to impossibility to pass from the source to the outflow along the remained arcs. There are several crosscuts in the circuit. Lemma 1 and lemma 2 establish connection between crosscuts and maximum flow. Lemma 1 concludes, that the value of any permissible flow from the source to the overflow is not greater than the value of any crosscut. Let us consider any crosscut, defined by multiple V. Assume W – all other circuit vertexes, not included into the multiple V. Let x_{ij} – value of flow for the arc (i, j), and z – overall value of the flow from the source to outflow. If to summarize conditions of flow preservation for all vertexes from the multiple W, then values of flows for arcs (i, j), for which vertex i and vertex j belong to the multiple W, will reduce, then in the result remains

Taking into account that the first sum from the given ratio is not bigger than the crosscut value, it can be concluded that Lemma 1 is true [8].

Lemma 2 lies in the fact that if the outflow cannot be labelled, then value of some crosscut equals to the flow's value. Let V is multiple of unmarked vertexes, and W is multiple of labelled vertexes. Let us consider arcs (i, j), for which $i \in W$, $j \in V$, then for them $x_{ij}=c_{ij}$ is true. It follows because, in the contrary case we could mark vertex j from the multiple V (as the arc (i, j) is the straight direction arc), which would contradict to determination of the multiple V.

Let us consider arcs (i, j), for which $i \in V$, a $j \in W$, then for them $x_{ij}=0$ is true. It follows because in the contrary case we could label vertex i from the multiple V (as the arc(i, j) is the backward direction arc), which would contradict to determination of the multiple V. Thus, it is seen from the ratio, that crosscut value equals to the flow value.

2.5 Nash equilibrium

Nash equilibrium is the situation, upon which none of the players can increase own bending of the game, changing, on a unilateral basis, own decision. In other ways, it is the situation, at which the strategy of both players is the best reaction at opponent's actions.

Rational approach to finding the game solution supposes, that any player i forms an opinion on other players actions and selects as own best answer. Situation in the game is called Nash equilibrium, if for any player i and for his any strategy there is fulfilled inequation Put it otherwise, is the best reply for every player i. The given situation is such, that it is not beneficial for anybody to deviate from it. If others confine themselves to it.

Nash equilibrium is the main concept for solving in no cooperative case. Notion of equilibrium connects two hypotheses on players' behavior. The first – if the situation is unbalanced, it cannot be considered as stable state. That is, if a player sees that deviation from will bring the bigger bending game, then he/she, most likely, will deviate. It matches to rationality hypothesis. However, the player surely understands that his deviation can arouse unpredictable chain of responses from other players, final consequences of which is difficult to overestimate. Such deviation is justified only in case if there is confidence that other players keep unchanged their strategies.

The second hypothesis– if every player sees that deviations from bring no improvement, he will maintain that strategy. Equilibrium bending of the game cannot be less than guaranteed level α_i.

Lemma 1 lies in the fact that if - Nash equilibrium, then for any player- i.

Lemma 2 supposes that for every player there prescribed subtotals . Suppose, that - equilibrium in the game, and for any i. Then is equilibrium in the game.

If is a game, obtained after iterated elimination of strongly dominated strategies, then We can show that any equilibrium in the game is equilibrium in the initial game , that is, we can record The given congruence explains the sense of elimination of heavily dominated strategies. If after sequential exclusion there is one profile remained, it is in equilibrium in the initial game, but if there several profiles remained, then it is necessary to find the balanced one among them.

Nash theorem. Let us assume that in the game all multiples are convex, and functions of bending of the game are persistent and hill-shaped per variable, then there exists at least one Nash equilibrium.

2.6 **Analysis of flow mption along the arc**

It is obvious, that in order the movement along total length was minimal, it is necessary, that the motion speed of every participant on the arc was maximum. However, other participants involuntarily affect at the speed of a certain participant. They as well strive to speed maximization, selecting own motion parameters. Flow increase leads to motion speed decrease of a considered driver, which results in time increase.

Let us consider motion only along one arc, and omit all indices concerning the arcs. Let us introduce following designations: L – length of circuit section, T – time of movement along the section, x – the flow, having passed through a road section per time unit, ρ - flow density, s – number of lanes in the corridor, w – speed of the flow, λ - average length of the corridor.

According to determination the density is $\rho=1/\lambda$. Let w – speed of a student, w_{max} – maximum speed. Time, spent by a person to travel a section of length λ equals to $\tau=\lambda/v$. Amount of students per time unit will equal to $\kappa=1/\tau$. Therefore, We'll consider that speed and density of flow are interconnected with linear dependence (Grindshils formula), here from or Substituting it into the flow, we obtain Obtained function is parabola with branches downward directed, maximum is reached at and accordingly Thus, we received the value of maximum flow, which can be passed through the road.

Let us substitute instead of ρ its expression, we'll obtain Using Viete formula we obtain with account that every participant strives to maximize own speed. Herefrom, we receive that motion time along circuit section is expressed with following dependence: where τ_{min} – minimal motion time along the section in case when the flow along it equals to zero. For descriptive reasons of that function we provide the function graph

Fig. 2. Diagram of function for time calculation

2.7 **Balance ratio of the layer**

Let $G=<E, V, H>$ is an oriented graph, E and V - final multiples,

H – mapping $H: V \rightarrow E \times E$. Let us name elements of multiple E the graph vertexes, elements of multiple V -arcs. For every arc $v \in V$ mapping $H(v)=(h_1(v), h_2(v))$, $h_1(v)$ – start of the arc v, $h_2(v)$ – end. Denote - multiple arcs, entering the vertex i, - multiple of arcs, outgoing from vertex i.

For every pair (i, j) of vertexes there prescribed numbers Q_{ij}, defining the value of the flow from the vertex - source i to the vertex - outflow j. These flows split up into separate currents and distributed along the circuit, in the result for every arc $v \in V$ we receive - flow along the arc v, moving from the source i to outflow j.

Let us take the vertex i_0, which is the source of the flow to other vertexes. Flows entering vertexes $i \in E$ we denote as $q_i(i_0)$, for vertex $i_0 \in E$ it will be . Total we will name a layer i_0. For every vertex

is true.

For vertex i_0

is true

Ratio is the First Kirchhoff rule for the circuit. Let us denote through the total flow moving along the arc v.

Accordingly,

Without restricting the generality we will assume that every vertex forms a layer, if for some vertex i_0 it is not available, then it means [9].

Single-layer systems, in themselves, have big practical significance. The simplest examples of such tasks are problems of entering some institution, evacuation from the buildings, stadiums, etc.

As we consider one layer, then index of layer i_0 we'll omit. Idea of equilibrium search algorithm consists in searching permissible initial flows and their subsequent transformation into balanced state. As every arc has limited flowing capacity, the check of existing permissible flows along with search can be fulfilled by means of the task on maximum flow and solving it by means of Ford-Fulkerson algorithm.

In the task of maximum flow, the latter passes from initial vertex to one final. All arcs have prescribed flowing capacity. In order to deduce the task to such form we will add two fictitious vertexes ii and kk. Let us connect ii to the source of the flow i_0. We connect outflows with vertex kk by arcs. Flowing capacity of those arcs equals accordingly $q_i(i)$. We obtain the task on maximum flow in standard form, for its solution, we apply any of the known algorithms. If it turned out that maximum flow is less than , then initial task of one layer, and accordingly, the whole task has no solution. In that case, minimal crosscut is out of additional arcs.

If it turned out that maximum flow equals , then we obtain permissible flow, which we transfer into balance state by means of invariant transformations. Let us consider any cycle C. Let us define any direction of by-pass, coinciding with some arc direction from cycle u. Let us construct generic function :

Let xv, $v \in V$ satisfy ratio. Let us take any number θ, for all $v \in V$ suppose , that is for cycle arcs, direction of which coincides with by-pass direction, the value of flow xv is added θ, for cycle arcs, direction of which is contrarily to by-pass direction, the value of the flow xv is deduced θ. Then satisfies the ratio.

Conclusion

In the work herein we studied models of flows distribution along the circuit using game-theoretical approach. We executed following tasks:

- given descriptive setting of the problem, with consideration of emergencies classification, norms, stages and principles of evacuation rating;

- given mathematical setting of the problem, with consideration of problem solving algorithm, task on maximum flow, method of potentials and criterion of optimality, Ford-Fulkerson algorithm, second Kirchhoff convention, Nash equilibrium and contour interrelationship;

-fulfilled search of permissible solutions of the task based on the problem on maximum flow;

-search of minimal evacuation time based on game-theoretical approach to people's flow motion modeling;

-search of the shortest path using the algorithm of finding balanced state in describing of people's flow motion model.

REFERENCES

[1] Kovalenko A.G., Vlasova I.A., Borisova S.P. Games theory and operations research. Samara: Publishing house «Samarsky University», 2006. – 147p.

[2] Volkov I.K., Zagoruiko Ye.A. Operations research. Moscow: Publishing house of MSTU named after Bauman N.E., 2000. – 435p.

[3] Vasin A.A., Morozov V.V. Games theory and econometric models. Moscow: MAKC Press, 2005. – 237p.

[4] Gorlach B.A. Operations research. S-Petersburg: Publishing house «Lanj», 2013. – 448p.

[5] Kosorukov O.A., Mischenko A.V. Operations research. Moscow: Publishing house «Ekzamen», 2003. – 448p.

[6] Belyayev S.V. Evacuation of mass public buildings. Moscow: Publishing house of «All-Union Academy of architecture », 1938. – 257p.

[7] Germeiyer Yu.B. Introduction into the theory of operations research. Moscow.: Nauka, 1971. – 358p.

[8] Neiman J., Morgenshtern O. Games theory and economic behavior. Moscow.: Nauka, 1970. – 244p.

[9] Zamkov O.O., Tolstopyatenko A.V., Cheremnykh Yu.N. Mathematical methods in economics . Moscow: MSU, 1997. – 366p.

[10] Kuznetsov A.I., Sakovich V.A., Kholod N.I. Mathematical Programming. Minsk: Higher School,2001. – 352p.