PROCEEDINGS OF THE NINETEENTH INTERNATIONAL SYMPOSIUM ON ARTIFICIAL LIFE AND ROBOTICS

(AROB 19th 2014)

January 22– 24, 2014

B-Con Plaza, Beppu, Oita, JAPAN

Publication Date: January 22, 2014
ISBN 978-4-9907582-0-2
ISSN 2185-3797

International Society of Artificial Life and Robotics

Room B

OS16 System Sensing and Control

Chair: Kenji Sawada (The University of Electro-Communications, Japan)

Co-Chair: Tota Mizuno (The University of Electro-Communications, Japan)

OS16-1 *Force Sensation Affected by Visual Stimulation with a Hand-held Multimodal Vision-tactile-force Display*

Tota Mizuno (The University of Electro-Communications, Japan)

Jun Maeda, Yuichiro Kume (Tokyo Polytechnic University, Japan)

OS16-2 *Control of Body-Sway Using a Tactile Device*

Kunio Horiba, Masafumi Uchida (The University of Electro-Comunications, Japan)

OS16-3 *Data Processing of Hybrid Vehicle Driving using Cluster Analysis*

Kohei Komatsuzaki, Seiichi Shin, Kenji Sawada (The University of Electro-Communications, Japan)

OS16-4 *Air-conditioner adaptation for indoor environmental change via summational type adaptive observer*

Ryo Yanagawa, Kenji Sawada, Seiichi Shin (The University of Electro-Communications, Japan)

OS16-5 *On group motion control of multi-agent systems with obstacle avoidance -the case of column formation under input constraints-*

Naoaki Niizuma, Kazushi Nakano, Tetsuro Funato, Kohji Higuchi (The University of Electro-Communications, Japan)

OS16-6 *StRRT Based Path Planning with PSO-tuned Parameters for RoboCup Soccer*

Katsumichi Sameshima, Kazushi Nakano, Tetsuro Funato (The University of Electro-Communications, Japan)

Shu Hosokawa (Control System Security Center, Japan)

OS16-7 *Model following control for continuous-time discrete-valued input systems*

Kenji Sawada, Seiichi Shin (The University of Electro-Communications, Japan)

Room C

GS10 Intelligent control & modeling II

Chair: Takeshi Nishida (Kyushu Institute of Technology, Japan)

GS10-1 *Development of microwave sensor for measuring moisture in granulation process*

Shigeru Nakayama (Kagoshima University, Japan)

Koichi Kawaguchi, Tomoomi Segawa, Yoshikazu Yamada (Japan Atomic Energy Agency, Japan)

GS10-2 *Importance of the real-world properties in chasing task -Simulation and analysis of dragonfly's behavior-*

Naoyuki Sakuraba, Kazuyuki Ito (Hosei University, Japan)

GS10-3 *Intelligent Mobile Agents for Disaster Response: survivor search and simple communication support*

Assel Akzhalova, Dmitry Mukharsky (Kazakh-British Technical University, Kazakhstan)

Atsushi Inoue (Eastern Washington University, USA)

GS10-4 *Movement Imitation in a Humanoid Robot with Approximate Inference Control*

Toru Kadoya, Hideaki Itoh, Hisao Fukumoto, Hiroshi Wakuya, Tatsuya Furukawa (Saga University, Japan)

GS10-5 *Occupancy grid map of semi-static objects by mobile observer*

Chau V. Dang, Hiroshi Sato, Tomohiro Shirakawa, Akira Namatame (National Defense Academy, Japan)

The Nineteenth International Symposium on Artificial Life and Robotics 2014 (AROB 19th 2014),
B-Con Plaza, Beppu, Japan, January 22-24, 2014

Intelligent Mobile Agents for Disaster Response: survivor search and simple communication support

Assel Akzhalova[1], Atsushi Inoue[2], and Dmitry Mukharsky[3]

[1]Kazakh-British Technical University, Kazakhstan
[2]Eastern Washington University, USA
[3]Kazakh-British Technical University, Kazakhstan
(Tel: +7-7272-72-1502)

[1]assel.akzhalova@gmail.com
[2]inoueatsushij@gmail.com
[3]amiddd@rambler.ru

Abstract: Search and Rescue problem can be defined as the cooperative use of resources by mobile agents to accomplish their mission of finding and rescuing important assets that are either lost or in some way at risk. We consider autonomous intelligent mobile robot-agents that have several sensors and actuators as well as communication capabilities. All mobile agents have an identical architecture so that they autonomously perform their tasks. These robot-agents perform tasks which may be delegated to a single robot-agent or to a group of robot-agents to be implemented in a distributed way. In case of collective task execution the robot-agents should be able to cooperate to minimize resources and improve mission performance. The mobile robot-agents operate in an environment that is only partially accessible. We offer the control framework that uses behavioral model and hybrid approach based on PSO and Q-learning.

Keywords: Autonomous, intelligent, agent, search, Q-learning, PSO.

1 INTRODUCTION

Collective of autonomous mobile robot-agents can be applied to carrying out rescue and abnormal efforts in zones of ecological disasters and cataclysms when an exploration of terrain is impossible without automated vehicles [1], [2]. However, control of the group of mobile robot-agents in case of limited resources is complicated as there are several factors affecting quality such as indefinite and non-deterministic environment, incomplete information obtained via sensors with limited range, and frequent absence of communication between all robot-agents because of obstacles. Besides meeting rigid requirements for noise immunity and reliability support to the communications is crucial as it guarantees the maximum system performance to the whole group of robot-agents.

The one of the most effective solutions of the search and rescue problems in the conditions of dynamic-changing environment is the decision-making based on intelligent robot-agents [3]. Actions of each intelligent robot-agent of the group have to be directed on achieving the global target that is often can be divided on to the sequence of sub-targets.

Typically each member of the group of robot-agents is unitary one having small size and weak functional characteristics. This is very critical for restrictive resources, e.g. embedded systems with 1GHz CPU, 0.5GB RAM, few GB dist space. Although having small size and functional capabilities, however, the whole group of robot-agents is capable to solve the global problem effectively [4]. In case of failure of some robot-agents functioning in extreme conditions, the collective of robot-agents will be able to continue operation of the global goal. Moreover, collective control assumes redistribution of collective operations of robot-agents initially intending to some failed robot-agents. The redistribution has to be carried out depending on the current conditions to achieve the collective goal in shortest time and lowest expenses.

Assume that a large group of robot-agents is deployed to the low-studied region and has to find and rescue victims. Each robot-agent in the group possesses simple technical capabilities such as detection by means of sensors, signal transmission to other robot-agents in the group, being nearby, finding direction and distances to adjacent robot-agents, and both controls of the direction and speed of all movement. There is no initial data about the environment and, therefore, it is essential to avoid unknown obstacles, find victims and try to build a map of the overall field being monitored. The global task is set to the collective of robot-agents as follows: as many as possible robot-agents must find the maximum quantity of rescued victims spending minimum expenses within certain time of period or minimum time [5].

Among many necessary tasks such as communication, sensing, manipulation, etc., the most critical is collaboration [6]. Communication is generally complicated as sending signals may easily be obstructed. Therefore, mobile robot-agents operate in an environment that is only partially accessible, dynamic and non-uniform in terms of its configuration, presence of other robot-agents or level of intelligence. In addition to the robot-agents in the environment, several utilitarian agents are presented such as a knowledge base, a centralized intelligent controller for detection and resolution of possible conflicts between two or more agents that can also specify tasks and missions.

In this paper we offer an intelligent model of control of the autonomous robotic system in the conditions of dynamically changing environment, restrictions on reliability and resources that is based on Particle Swarm Optimization (PSO) and Q-learning. First section considers the architecture of the intelligent robot-agents control. Second section formulates the problem of collective control to achieve the global best performance. Third section presents an approach to solve the problem and intelligently maintain control of the group of robot-agents that can be observed from simulation results shown in the paper.

2 RELATED WORK

The one of most popular strategies of the autonomous control system design to attain the global target is to build an adaptive control for each robot-agent based on feedback from the adjacent robot-agents [7]. In spite of the fact that local interactions between robot-agents are simple scalable and reliable control systems provide the global behavior that intelligently manages the group of robot-agents. Certain success in this direction is achieved and series of research work is based on development of techniques related to the rules of group behavior control [8], [9]. [10] shows that the robot-agent movement in the appropriate changing environment can be realized by means of the table describing rules of movement and the distributed retroactive effect. Authors demonstrate that such approach allows the robot to bypass hindrances in the dynamic environment.

In this paper we introduce the table of movements that follow certain rules providing fast learning process and avoiding obstacles to the group of intelligent robot-agents.

[11] presents the controller that adapts to the different topology of modular robots. However, [11] does not consider strategy maintaining modular robots adaptation to the dynamic environment. [12] demonstrates distributed algorithms for movement through barriers.

In this paper we offer intelligent model of the collective control of autonomous robot-agents that employ an optimal control approach in the conditions of dynamically changing environment and constraints on reliability and resources of system. For the formulated problem we introduce a hybrid of PSO and Q-learning based on specified table of rules providing intelligent control in the applications of search and rescue problem.

3 AN ADAPTIVE CONTROL FRAMEWORK

3.1 An architecture of the control system

Traditional architecture of collective of autonomous robot-agents consists of communicated robots equipped by a series of sensors to obtain necessary information and detect victims within broad range, actuators, and transceivers operating through appropriate ports. Any task that is executed by some robot-agent can be considered as a sequence of actions. Meantime, the global task is a sequence of subtasks. The task can be solved by the target group of robot-agents chosen by the system to complete it within certain period of time or certain required amount of resources. During process of interactions between robot-agents it is often required that if some connection with a robot-agent is broken then it might be necessary to continue an execution of the intended target autonomously. Thus, it is necessary to develop a policy of intelligent decision-making on how to behave in the environment and achieve the set goal.

We introduce distributed system of a number of interacting intelligent robot-agents where each robot-agent is an autonomous control subsystem. The control subsystem realizes a local policy of decision-making and takes into account the global goal. Thus, in order to maintain scalability and robustness of the distributed system not only autonomous subsystem plays an important role but also a mechanism that can rebuild communication links between robot-agents and/or assignment of the robot-agent to new tasks on the basis of a current state of the environment and predicted future changes. We offer the architecture of control of autonomous robot-agents taking into account the structure of the intelligent robot-agent provided in **Fig. 1** which consists of the following components: Knowledge Base, Analyzer, Goals, Behavior, Scenario that are parts of the decision-making process.

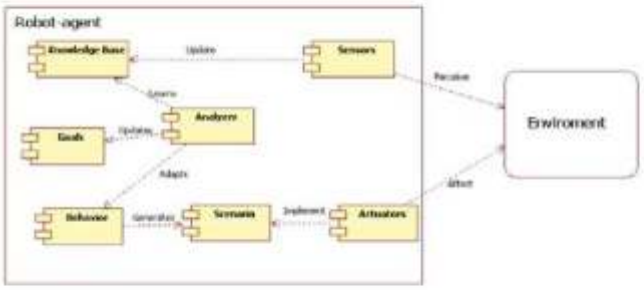

Fig.1 Architecture of the autonomous intelligent robot-agent

The intelligent control subsystem observes the non-deterministic environment via Sensors, enriches Knowledge Base and adapts its Behavioral model after specifying Goals by Analyzer, generates Scenarios that contain planning actions implemented by Actuators.

During execution of the tasks by the target groups of robot-agents it is also necessary to be convinced that the group completed all tasks. If some members of the group are not able to execute some tasks then the control system should redirect those tasks to available robot-agents. Analyzer provided in our control infrastructure allows fixing outstanding tasks and update Goals. Analyzer takes into account CostOfResources, CostOfObstacles, ReliabilityLevel and analyzes Goals. These goals (tasks) are distributed among best candidates among existing robot-agents. The process of the candidate assessment can demand time. The one of important feature of proposed model of control that can reduce time of such assessment is to use Scenarios based on Behavior policy (**Fig. 2**).

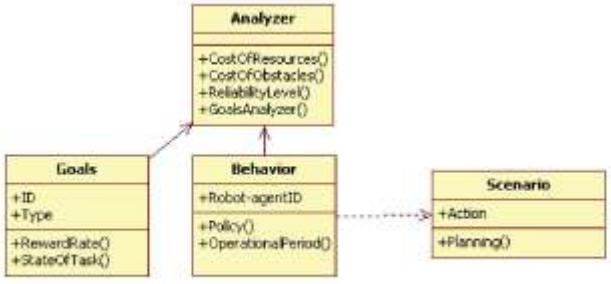

Fig.2 Behavior model for an intelligent decision-making process

Policy() specifies rules of adaptation and generates Scenarios that can be used if we have the same conditions while selecting candidates it is faster to load store Scenario, thereby reducing loads to the communications. Behavior model is also used to copy (imitate) the leader's actions if there are needs to continue operations without loading of communication network. All those features accelerate decision-making process.

Behavior model always is updating as the intellectual robot-agent is trained to act optimally in dynamically changing environment. We formulate an optimization problem to reach maximum group performance where factors of uncertainty of the environment and limited resources play roles of constraints. Next sub-section presents the optimization problem which will be solved by hybrid technique in the third section.

3.2 An optimal control problem

Let $\Re$ a group of N robot-agents operates in the environment E. The state of robot-agent at time t we describe as a vector:

$$\mathbf{R}_j(t) = [r_1(t), r_2(t), \ldots r_n(t)]^T,$$

where $r_i(t)$ are the values of the state parameters at time t. The state parameters $r_i(t)$ are determined as energy resources, speed and acceleration, position, the orientation.

In order to introduce group of robot-agents we apply a vector-function that defines the state of the group at the time step t:

$$\Re(t) = f_R(\mathbf{R}_1(t), \mathbf{R}_2(t), \ldots, \mathbf{R}_N(t))$$

The control system should able to execute a sequence of tasks T_j from the set of tasks $T = \{T_1, \ldots, T_m\}$ updated by Analyzer. Each robot-agent R_i is able to work on only one task. The task distribution among robot-agents is determined by Behavior model that produces a match $\langle R_i, T_j \rangle$. Let G is the sequence of tasks which is global Goal.

According to disaster scenario, the group of robot-agents operates in the non-deterministic and non-stationary environment that is determined by a vector-function:

$$\mathbf{E}_j(t) = [e_1(t), e_2(t), \ldots e_n(t)]^T,$$

where $e_i(t)$ are observed parameters of the environment. We want to provide fast and intelligent control framework of autonomous robot-agents system that brings maximum performance. An overall performance F_G to achieve the global task G by the group of robot-agents at certain period of time can be formalized as a subtraction of the cost associated with task failures from the successful task completion and excluding collisions.

It is necessary to find a control functions $\mathbf{u}_j$ that leads the group of robots to accomplish the goal with maximum performance.

We formulate an optimal control problem of the robot-agents group $R_j \in \Re$ can be formulated as one to identify such $\mathbf{u}_j$ control functions for each robot-agent in the range $[t_c, t_c + \Delta t]$ that provide maximum to the functional:

$$Y = \int_{t_c}^{t_c+\Delta t} F_G(t, \Re(t), \mathbf{E}(t), \mathbf{u}_1(t), \mathbf{u}_2(t), \ldots, \mathbf{u}_N(t))dt \quad (1)$$

which is subject to constraints:

$$\mathbf{u}_j(t) \in \{\mathbf{u}_e(t)\}, \quad (2)$$

$$\mathbf{u}_j(t) = f_u(\mathbf{u}_1(t), \ldots, \mathbf{u}_{j-1}(t), \mathbf{u}_{j+1}(t) \ldots, \mathbf{u}_N(t)) \quad (3)$$

where u_e the vector control must belong to the set of admissible controls which define a desirable level of reliability of each robot-agent; t_c is a current time step and t_f is the end of the operation of the group $\Re$; f_u means that the vector control $\mathbf{u}_j$ depends on other robot-agents control. Our functional (1) shows performance of the group control that has to reach the goal. We transform (1)-(3) in a compact form as follows:

The control system has to find $\mathbf{u}_j$ control functions for each robot-agent performing global task in non-deterministic and dynamically changing environment E during observing period of time $[t_c, t_c + \Delta t]$ that gives maximum to the functional (1):

$$Y \to \max \quad (4)$$

which is subject to constraints:

$$\mathbf{u}_j(t) \in \{\mathbf{u}_e\} \quad (j = \overline{1, N}) \quad (5)$$

$$\mathbf{u}_j(t) = f_u(\mathbf{u}_1(t), \ldots, \mathbf{u}_{j-1}(t), \mathbf{u}_{j+1}(t) \ldots, \mathbf{u}_N(t)) \quad (6)$$

Next section shows an approach to solve (4)-(6).

3 HYBRID CONTROL APPROACH

We employ both PSO and Q-learning approaches.

PSO method is based on collective behavior of socio-organized "living" groups. We apply Particle Swarm Optimization techniques (PSO) where robots-agents are particles. According to PSO approach, each iteration particles are searching an optimal solution for some position and velocity [13]. Q-learning method is used to provide intelligent control framework of robot-agents system [14]. At each step the evaluation function Y is stored in a special table where inputs are states and tasks. Next step evaluation function is attributed a certain way. In Fig. 3 we demonstrate the simulation tool that realizes a hybrid of the approaches as well as PSO and Q-learning by themselves.

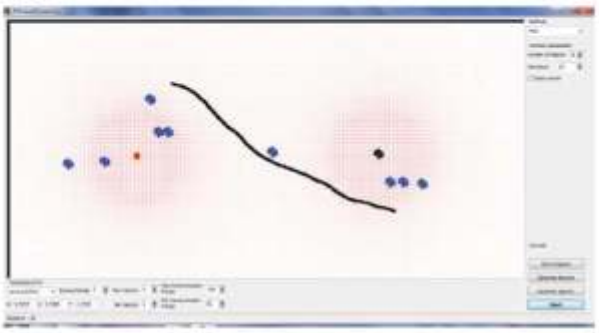

Fig.3 An interface of the simulation tool

The graphical user interface of the program you can see on Fig.1. We can add barriers and radiation sources on the field drawing different forms of obstacles on the screen. The radiation sources are represented as red circles. The sources has the intensity of the radiation sources. The simulation tool allows selection of the method of optimization control among: PSO, Q-Learning and hybrid of two algorithms PSO + Q-Learning. Choosing one of the methods changes the bar at the bottom of the main window. It is necessary initially specify parameters of the problem such as:

Numbers of Agents – quantity agents with participate in experiment;

Iterations – quantity of steps of the simulation;

State anthill – all agents bob up on work field not random places and in upper-left corner of the field.

For PSO implementation there is a special bar that has the following parameters:

Different modifications of PSO;

Sensing Range – radius of the sensors of the agent to find radiation source. When the agent approaches near to the radiation source it become green;

Max velocity – maximum speed of the agent;

Min velocity – minimum speed of the agent;

Max Communication Range – maximum distance between agents which they can translate messages. If the parameter are too big the agents will find only very far located sources;

Min Communication Range – minimum distance between agents for translate messages;

W, G, P – non-dimensional parameters of the model used to calculate velocity and position of robot-agents.

The Q-matrix is a main matrix of space solves of the agent. Q-matrix has columns that contain actions of the agent and rows containing states of the agent. At the initial state matrix is filled by zeros. In some realizations of the Q-learning the initial state of the matrix represents random numbers. When agent successful learns, the Q-matrix must

be successful full by non-zero values. Scheme of the Q-matrix is shown in **Table1**.

Table 1. Q-matrix

Action / State	0	1	2	3	..	n	Description		
1							Barrier is in the cell with number 1	Barriers	Priority of the state decreasing from top to bottom
2							Barrier is in the cell with number 2		
3							Barrier is in the cell with number 4		
…	…	..	..	..		…	…		
k							Maximum intensity in direction of the cell with number 1	Tasks	
k+1							Maximum intensity in direction of the cell with number 2		
k+2							And so on		
…	…	..	..	..	..	…	…		
m							The agent is in empty World		

Before the agent start turn it scans space around himself. Quantity of the state equals to the quantity of the cells which were scanned. Scanning has place in the following way: firstly, the scan radius takes minimum value that equals to 1 and the direction UP; next, the radius is increased with some fixed increment of the angle and the cells where agent checks a value of the cell. If the value of the cell is negative one (indicate presence of the border) then it returns this value of the cell as number of the state. Secondly, the radius increases to 1 and all process is repeated. For defined radius in ten unit (pixel) quality of the possible states are about 340. They all spread on an unwinding spiral from position of the agent. Identical process of scanning has place to find intensity of the radiation source. Searching of the cell with an intensity of the radiation source is successful if the value is maximum and this value returns as number of the state.

Most important in Q-learning algorithm is assignment of rewards of true and false moves. Truth assignment is guaranteed by fast and qualitative learning of the agent. The values can be positive and negative. In this simulator we use the following scheme of assignments: if the agent finds the radiation source receives one hundred points. If the agent remains backward from radiation source it receives negative five points. If the agent crashes with obstacle it receives negative ten points. In our experiments the agents often stopped in same position (action equal zero) and remained in it indefinitely. For this reason they receive negative reward for stopping that is equal to five points. LF, DF – non-dimensional parameters of the model Q-learning. They are presented in the formula the Q-matrix.

Quantity of the actions can vary depending on the choice of the user. Maximum quantity of the actions is 88 if maximum speed is five. Scheme of the agent movement according to the system of rewards is shown in **Fig. 4**.

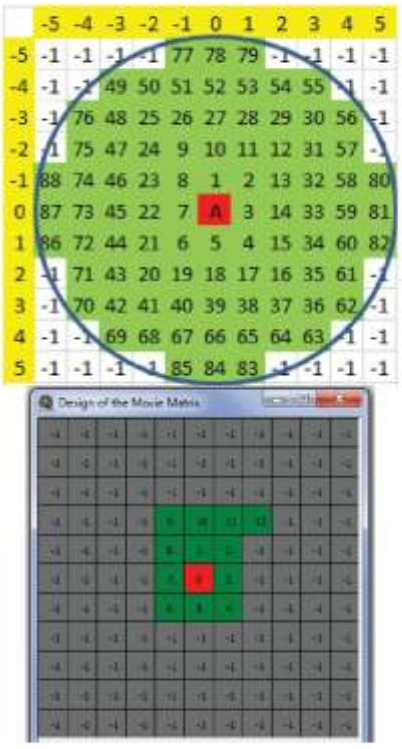

Fig. 4. Scheme of the agent learning: spiral scheme and Q-matrix

Red cell indicates position of the agent. In the matrix this cell coding by zero. The green cells indicate positions when the agent shall be able move from actual position if

its maximum speed is five. Yellow cells show array of indexes. These indexes are needed in order to coordinate real position of the agent receiving new allocation. Therefore, behavior of the agent is changing and learning process allows him chose best direction to find sources.

During learning process the simulator calculates percent of the learning of Q-matrix. It is a ratio of quality of the nonzero values of the matrix to the full quality of the elements of the matrix..

4 CONCLUSION

We have introduced the intelligent control framework for collective of autonomous robot-agents as well as hybrid approach of PSO and Q-learning to tackle uncertainty of environment that adapts behavior of the robot-agent and allows reach maximum performance. The proposed hybrid approach utilizes specific numbering of Q-matrix and allows fast decision-making. Future work will be directed towards applying mobile selection of leaders and extending behavior model of the robot-agents.

REFERENCES

[1] R. R. Murphy, S. Tadokoro, D. Nardi, A. Jacoff, P. Fiorini, H. Choset, and A., M. Erkmen (2008), Search and Rescue Robotics, Hand Books of Robotics, Springer, Germany, pp. 1151–1173

[2] M. Guarnieri, P. Debenest, T. Inoh, E. F. Fukushima, S. Hirose, (2004), Development of Helios VII: an arm equipped tracked vehicle for search and rescue operations. In Proc. of the International Conference Intelligent Robots and Systems, Sendai, Japan, , September 2004, pp. 39–45

[3] F. Amigoni, V. Caglioti (2010), An Information-based Exploration Strategy for Environment Mapping with Mobile Robots. In Robotics and Autonomous Systems, 58: 684-699

[4] Balajee Kannan, Lynne E. Parker, (2007). Metrics for quantifying system performance in intelligent, fault-tolerant multi-robot teams. In Proc. of IEEE International Conference on Intelligent Robots and Systems, San Diego, CA, 2007, pp.951-958

[5] W. Sheng, Q. Yang, and J. Tan, (2006). Distributed multi-robot coordination in area exploration. Robotic Autonomous Systems, 54:945-955

[6] Samuel Rutishauser, Nikolaus Correll, Alcherio Martinoli, (2009). Collaborative coverage using a swarm of networked miniature robots. Robotics and Autonomous Systems, 57(5):517-525

[7] A. J. Ijspeert, A. Martinoli, A. Billard, and L. M. Gambardella, (2001) Collaboration through the exploitation of local interactions in autonomous collective robotics: The stick pulling experiment. Autonomous Robots, 11:149–171

[8] S. Behnke, and R. A. Rojas, (2001) A hierarchy of reactive behaviors handles complexity. Balancing Reactivity and Social Deliberation in Multi-agent Systems, Springer, Berlin, Germany, 2001, pp. 239–248

[9] Chien-Ming Huang and Bilge Mutlu, (2012). Robot behavior toolkit: generating effective social behaviors for robots. In Proc. of the seventh annual ACM/IEEE international conference on Human-Robot Interaction (HRI '12). ACM, New York, NY, USA, 2012, pp.25-32

[10] Kurokawa H., Tomita K., Kamimura A., Yoshida E., Kokaji S., Murata S, (2005). Distributed self-reconfiguration control of modular robot M-TRAN. Mechatronics and Automation IEEE International Conference,1(29):254- 259

[11] Wei-Min Shen, Harris Chiu, Michael Rubenstein, Behnam Salemi, (2008). Rolling and Climbing by the Multifunctional SuperBot Reconfigurable Robotic System. In Proc. Space Technology and Applications Intl. Forum (STAIF-08), American Institute of Physics, 969: 839–848

[12] Keith Kotay, Daniela Rus, (2005). Efficient Locomotion for a Self-Reconfiguring Robot. In Proc. of IEEE Intl. Conf. on Robotics and Automation, Barcelona, Spain, 2005, pp.1127-1134

[13] J. Pugh, A. Martinoli, (2008), Distributed adaptation in multi-robot search using particle swarm optimization. In Lecture Notes in Computer Science, 10th International Conference on the Simulation of Adaptive Behavior, Osaka, Japan, 2008, pp.393–402

[14] R.S Sutton, A.G Barto, (1998) Reinforcement learning: An introduction. Cambridge Univ Press.