Automatic Synthesis Gait Scenarios for Reconfigurable Modular Robots Walking Platform Configuration

Sergey Manko and Evgeny Shestakov(✉)

Russian Technological University (MIREA), Moscow, Russia
shestakov.e.i@yandex.ru

Abstract. Reconfigurable mechatronic modular robots distinguished mainly by their ability to adapt their structure to specific tasks to be performed as well as to specific environments, are of great interest for a wide range of different applications. One of the key problems in motion control of this type of robots lies in the necessity to use original algorithms for each of the possible configurations whose variety is determined by the structure of mechatronic modules their number and the coupling option selected. Some standard configurations of mechatronic modular robots allow the possibility to develop motion control algorithms invariant to the number of modules in the kinematic structure. Yet, a promising approach to solving the problem is generally related to the development of self-learning means and methods to enable an automated synthesis of motion control algorithms for multi-unit mechatronic modular robots, taking into account the selected configuration. The present article discusses the results of exploratory research on using the apparatus of self-learning finite state machines for solving the problem of automated synthesis of gait scenarios in the walking plat-form configuration. The paper presents the results of model experiments confirming the workability and efficiency of the developed algorithms.

Keywords: Reconfigurable modular robots · Self-learning
Intelligent control systems · Finite state machine

1 Introduction

Attractiveness and functionality of modular reconfigurable robots conceptually developed by the end of the last century are wholly and totally determined by application of certain principles of modular design of complex technical systems. A combination of the mechanical structures modularity, hardware and software determines potential advantages of reconfigurable robots, such as a new class of electromechanical systems based on typical modules. Reconfigurable robots have a unique set of properties like multifunctionality, the adaptability of kinematic structure and its operational modifiability according to features of applied application and environmental conditions. Practical implementation of this similar approach related to the need for solving a number of key problems among which one of the most important are self-learning and automatic synthesis of multilink mechatronic modular robot control algorithms for its configuration synthesized based on the specifics of the current situation.

A. Ronzhin et al. (Eds.): ICR 2018, LNAI 11097, pp. 181–191, 2018.
https://doi.org/10.1007/978-3-319-99582-3_19

2 Features of Reconfigurable Mechatronic Modular Robot Functioning

The concept of construction mechatronic robots with adaptive kinematic structure means a presence of typical modules, combined into a single multilink structure. The typical mechatronic modules involve one or several motors with rotate joints and a simple mechanical transmissions, connectors for mechanical, electrical and information connection, controller, various sensors and autonomous power supply. Similar construction of typical modules provides their automatic docking and undocking for the operational formation of the necessary robot kinematic configuration depending on the goals and functional conditions.

Despite the diversity of the proposed variants of typical mechatronic modules, overwhelming majority of developers consider three main configurations of modular reconfigurable robots [1] for the targeted motion tasks, shown in Figs. 1 and 2:

- wheel configuration for moving on a flat surface;
- snake configuration for moving in a limited space;
- walking configuration for moving on a variable surface in complex scenes with numerous obstacles and irregularities.

Fig. 1. Examples of different configurations of modular robot PolyBot (PARK, Xerox, USA): wheel, snake and walking configuration.

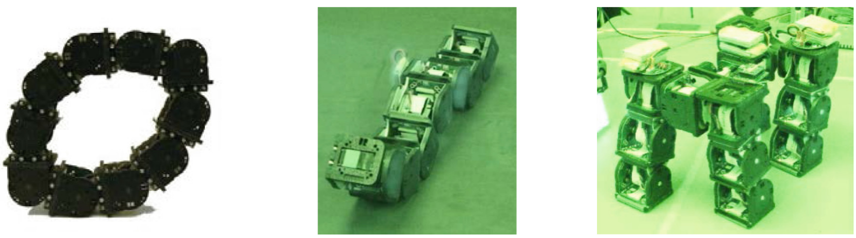

Fig. 2. Examples of different configurations of modular robot CkBot (Modlab, UPenn, USA): wheel, snake and walking configuration.

It should be noted, the control of this type of robots in snake and wheel configurations can be implemented using universal algorithms providing the wave-like recurrence of module movements in common kinematic chain for its targeted motion [2].

In common case the robot transformation requirement in a walking configuration imposes by complication of environment and permeability conditions. The number of legs and a number of joints in each of them must be determined based on the analysis of the actual situation considering the size of irregularities and obstacles, weight and size parameters of robot, payload and other factors.

The a priori uncertainty and plenty of possible options of modular robot kinematic scheme in a walking configuration don't allow for development required gait scenarios and according control algorithms. Consequently obviously the control problems of reconfigurable robots in walking configuration mainly related to the self-learning organization for automatic generation of gait scenarios for reasonably selected kinematic scheme with a fixed number of legs and their joints (Fig. 3).

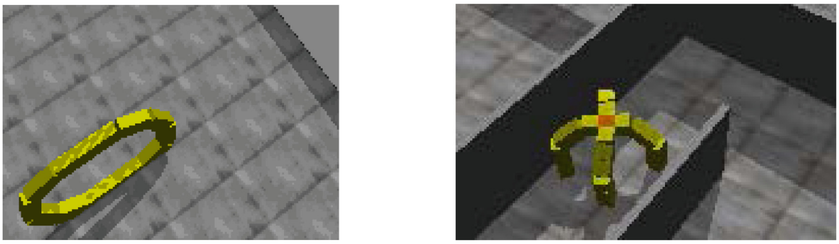

Fig. 3. Reconfigurable robot transformation in walking configuration because of complication of environment and permeability conditions.

3 Self-learning Methods and Technologies of in Intelligent Control Systems of Autonomous Robots

One of the key problems of construction intelligent control systems of autonomous robots and other types of complex dynamic objects operating in uncertainty conditions related to self-learning organization to acquire new knowledge about the surrounding world laws and behavior rules in certain situations.

Variety of self-learning tasks relevant to autonomous robotics [3] makes it necessary to find adequate methods of their solution. It should be noted that the theory of machine learning, as an independent subsection of artificial intelligence, has a lot of special tools and methods [4]. There are technologies based on clustering methods, reinforcement learning, evolutionary algorithms, regression analysis, Naive Bayes classifier, classification trees, particle swarm optimization, neural networks etc which are widely applied from medical and technical diagnostics to computer security and pattern recognition. The results of fundamental research in the field of intelligent control systems show some of these methods can be successfully used to solve some problems of self-learning autonomous robots [5–8]. In particular, the methods of evolutionary programming allows for automatic synthesis of algorithms for motion control of mobile platforms of various types [5, 6].

The methods of classification trees serve as an effective tool for autonomous robots self-learning, for example, to form knowledge about the patency of heterogeneous sections of the route in order to correct it quickly, taking into the minimization of the "cost" characteristics of the chosen trajectory.

Among the many well-known approaches to the self-learning organization, the specialized class of finite-state automata, the main principle of the construction and operation of which is associated with a change state depending on the current depth of its storage in memory, is of particular interest and perspective. Their peculiarity, realized in one way or another in automata of this kind, provides the solution of self-learning tasks aimed at identifying the conditions of the most effective interaction with the environment.

There is a number of characteristic representatives among of self-learning automata. One of them is Tsetlin machine (or linear tactics automata), whose state diagram is given by Fig. 4. For each action completed, the machine receives either negative or positive signals as a response from the environment.

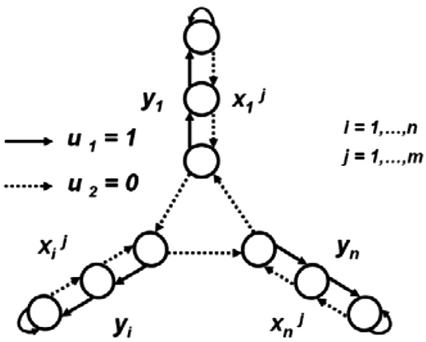

Fig. 4. Tsetlin machine state diagram.

In case of a positive response, the current state of the machine is restarted at the next higher memory level. A negative response causes a decrease in the depth of the current state storage or its cardinal change at the lowest memory level.

Thus, the finite state machine of this type can be interpreted as a dynamic system, which under the influence of some control command coming to the entrance at the time t, changes its current state x and the level j of its memory storage depth to a new one:

$$\begin{gathered} x(t), j(t) = f((x(t-1), j(t-1), u(t)) \\ y(t) = h(x(t)), \end{gathered} \tag{1}$$

where f, h are transition and output functions set by Table 1.

The depth of memory determines the inertial properties of the machine and allows to save the execution of the optimal action, even if there are single-piece negative responses. Herewith, it is proved that at sufficiently large values of the automata memory depth, its behavior tends to the optimum [9, 10].

Table 1. State transition table of Tsetlin machine.

Inputs\States	$x_i^j, \quad 1 < j < m$	$x_i^j, \quad j = 1$	$x_i^j, \quad j = m$
$u_1 = 1$	x_i^{j+1}/y_i	x_i^{j+1}/y_i	x_i^j/y_i
$u_2 = 0$	x_i^{j-1}/y_i	x_{i+1}^j/y_{i+1}	x_i^{j-1}/y_i

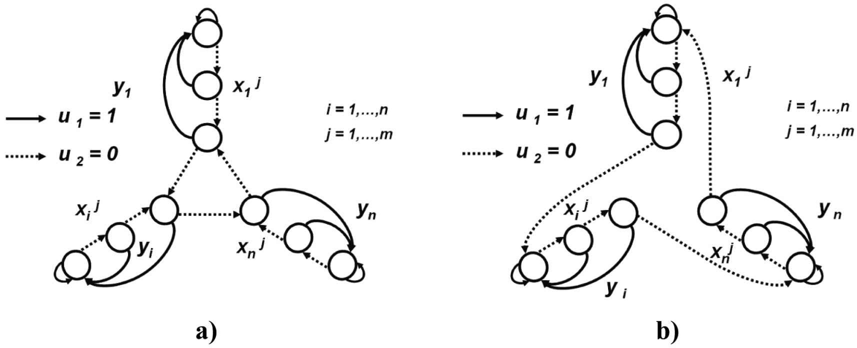

Fig. 5. Krinsky machine state diagram (a), Robbins machine state diagram (b).

Krinsky «trustful» machine is in mainly similar to the Tsetlin machine in principles of its construction and operation. As shown in Fig. 5(a), the main difference is in the current state transition of the machine to the deepest level of storage when receiving a positive response to the performed action.

As for Tsetlin machine, Krinsky and Robbins machines, whose transition and output functions are presented by Tables 2 and 3, it is strictly proved that their behavior in any stationary environment is rational.

Table 2. State transition table of Krinsky machine.

Inputs \ States	$x_i^j, \quad 1 < j < m$	$x_i^j, \quad j = 1$	$x_i^j, \quad j = m$
$u_1 = 1$	x_i^m/y_i	x_i^m/y_i	x_i^j/y_i
$u_2 = 0$	x_i^{j-1}/y_i	x_{i+1}^j/y_{i+1}	x_i^{j-1}/y_i

Table 3. State transition table of Robbins machine.

Inputs\States	$x_i^j, \quad 1 < j < m$	$x_i^j, \quad j = 1$	$x_i^j, \quad j = m$
$u_1 = 1$	x_i^m/y_i	x_i^m/y_i	x_i^j/y_i
$u_2 = 0$	x_i^{j-1}/y_i	x_{i+1}^m/y_{i+1}	x_i^{j-1}/y_i

4 Automatic Generation of a Reconfigurable Mechatronic Modular Robot Gait Scenarios in the Walking Configuration

One of the statement options of the considered problem on automatic synthesis of gait scenarios for walking configuration related with its interpretation from the point of view construction and functioning of self-learning automata. In this context, the set of interrelated states at different depths of the automata's memory is to be interpreted as a sequence of possible actions performed within a particular gait.

We assume the desired gait scenarios should be a cyclical process during which part of the legs is in motion, and the others serves as a static support. These requirements fully satisfys most simple and reliable (safe?) variant of the so-called "cautious" gait, when at each stage the motion is moved only one leg. In this case, the loss or save of the walking platform stability can be considered as criteria for selection of a suitable gait scenario in the process of its automated synthesis.

The simplification of the problem lies in its decomposition into two stages respectively, associated with the formation of a sequence of articulations for the rearrangement of a single leg and the order of the steps necessary for the robot transfer.

The use of such representations allows to totally determine the structure of automata which define the variety of gait scenarios as a set of combinations of possible actions for their implementation. A standard scenario of leg rearrangement in a new step interprets as a elementary sequence of joints rotations by value of the processed angle Δ.

Thus, the problem of automatic synthesis of scenario permutations of the leg rearrangement reduced to a combinatorial setting, allowing for the self-learning automata application to search for the necessary solutions. The general structure of a self-learning machine to search for a sequence of elementary rotations from start leg configuration to some target support state is shown in Fig. 6.

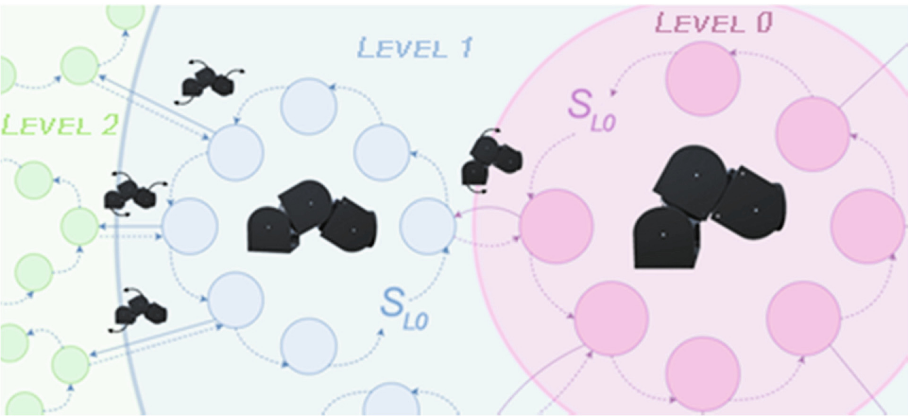

Fig. 6. General structure of self-learning machine for the scenario formation of the leg rearrangement.

Virtually this machine is divided into levels, in each of them one elementary action is generalized as leg joints angles change to the value Δ in the negative or positive direction either remain unchanged. Then the number of states S_{Li} in the i-th level to be determined by the following Eq. (2):

$$S_{Li} = \left(3^N\right)^{i+1}, \ (i = 0, 1, \ldots, L), \tag{2}$$

where N is the number of joints in extremities; L is the number of levels calculated by the Eq. (3):

$$L = \frac{|q_{min} - q_{max}|}{\Delta}, \tag{3}$$

where q_{min} and q_{max} are the minimum and maximum possible angles in the joint. Then, the total number of machine states does not exceed the value S_1:

$$S_1 = \sum\nolimits_{i=0}^{L-1} S_{Li}. \tag{4}$$

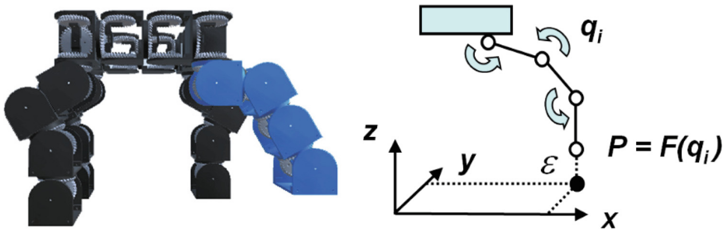

Fig. 7. Conformity assessment the current configuration of the leg to condition support.

As a criterion for desired solution selection, the condition for matching the leg joints current configuration with the support state can be used. In this case, as shown in Fig. 7, the current position of the leg P relative to the reference surface doesn't not exceed the specified level ε:

$$P \leq \varepsilon, \tag{5}$$

- $q_i = k_i\,\Delta$, ($i = 1, 2, \ldots, N$) are generalized coordinates of the leg;
- k_i is the conversion coefficient, determined by the self-learning machine in its working process;
- N is the number of joints in the leg;
- F is the transfer function between generalized and Cartesian coordinates.

Importantly the assignable elementary rotation value Δ selection and matching the current configuration to the support state ε condition essentially influence on search time of suitable scenarios, but also on the walking platform dynamics. In case of these parameters increasing should be a duration search reduction but to the dynamics impairment in the sense of amplitude vertical oscillations of robot center of mass increasing. Vice versa the elementary rotation value angle Δ and tolerance range ε decreasing should cause duration search increasing and dynamics motion improving.

The gait scenario is to regulate the rearrangement legs order of the walking platform during motion. As an example the structure of the self-learning machine, for a four-legged walking configuration is presented in Fig. 8. States S2 of this machine are determined by a set of rearrangement legs variants in accordance with the "cautious" gait concept:

$$S_2 = K!, \tag{6}$$

where K is the number of legs. To get all possible "cautious" gait variants one of the known generating permutations algorithms can be used. State transitions determinate by the self-learning automata idea with input negative or positive signals meaning loss or preservation of the robot stability. Stability assessment defines on software level by condition to entry projection of the center of platform gravity in the support area or by virtual physics simulation.

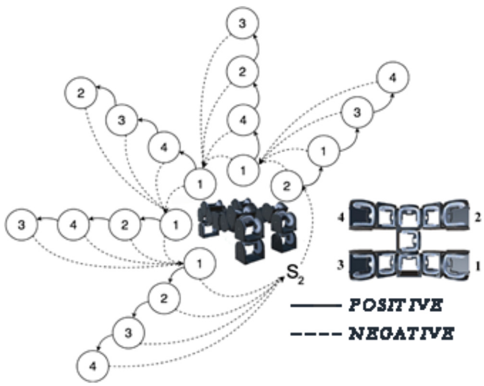

Fig. 8. General structure of self-learning machine for the sequence formation of the robot steps.

For the four-legged configuration, given for example in Fig. 9, not all cautious gait options are successful. As shown in Fig. 9 the scheme "1-2-3-4" changing the position of the center of robot gravity after first step, stability loss resulting. But the other scheme "4-3-1-2" safes the motion stability (Fig. 10).

A model experiments series convincingly testifies to possibility and efficiency of the offered approach use for automatic gait scenarios synthesis. The produced experimental results are shown in Tables 4 and 5 and Fig. 11. confirm expected and actual nature dependence of the duration search of gait scenarios and amplitude robot center of mass oscillations on the elementary rotation angle values Δ and tolerance leg position over the reference surface ε.

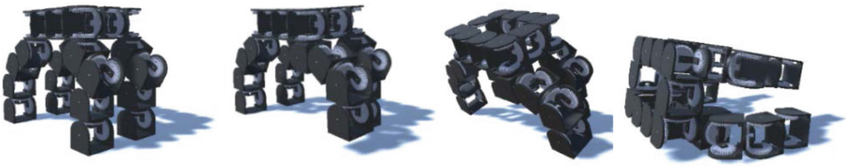

Fig. 9. Modeling of walking platform motion based on "cautious gait" scheme "1-2-3-4" where there is a loss of stability.

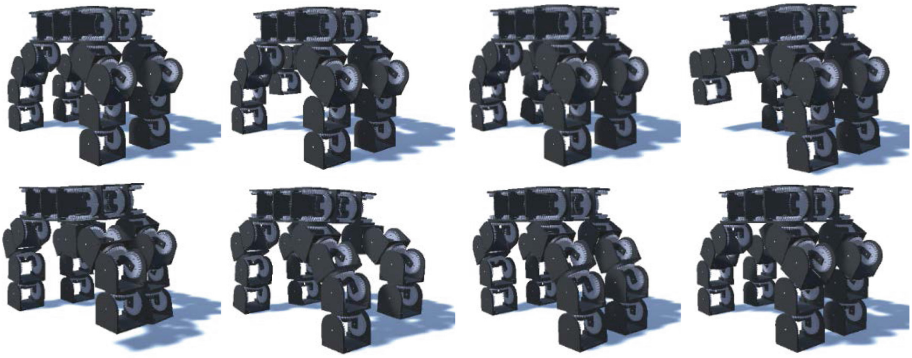

Fig. 10. Modeling of walking platform motion based on "cautious gait" scheme "4-3-2-1" where there isn't a loss of stability.

Table 4. Center of mass amplitude oscillations dependence on discretization angle and clearance.

	ε = 2.0 cm	ε = 2.5 cm	ε = 3.0 cm	ε = 3.5 cm
Δ = 30°	181,38	161,76	27,7	26,6
Δ = 20°	55,25	36,28	36,1	33,4

Table 5. The average learning time dependence on the angle discretization values and the clearance.

	ε = 2.0 cm	ε = 2.5 cm	ε = 3.0 cm	ε = 3.5 cm
Δ = 30°	181,38	161,76	27,7	26,6
Δ = 20°	55,25	36,28	36,1	33,4

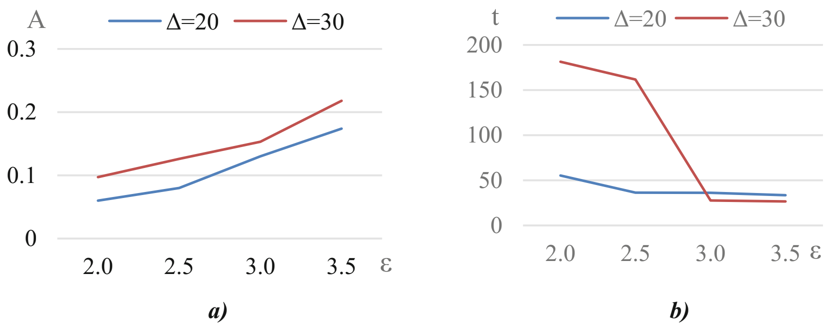

Fig. 11. Experimental results: a) dependence of the center of mass amplitude oscillations on angle of discretization and clearance b) dependence of the average learning time on the angle discretization values and the clearance.

5 Conclusions and Future Works

Self-learning opens up broad prospects for the automation of intelligent control systems synthesis and configuration, and to improve their functional and adaptive capabilities by the analysis and synthesis of the their working results. The present paper demonstrates development possibility and expediency attracting of self-learning automata for gait scenarios synthesis for modular robots in walking configuration. There are other approaches to solving this problem related, for example, to the use of genetic algorithms and evolutionary programming methods [5, 6]. Development of autonomous robot with advanced adaptive capacity including reconfigurability assumes necessity of their effective self-learning on-board equipment. In this regard, the future work related to a comparative analysis to effectiveness assessment of the genetic algorithms and self-learning automata use for automatic behavior and control algorithms generation.

References

1. Makarov, I.M., Lokhin, V.M., Manko, S.V.: Multilink mechatronic modular robots with adaptive kinematic structure. Mechatron. Autom. Control **11** (2006)
2. Makarov, I.M., Lokhin, V.M., Manko, S.V.: Motion control of mechatronic modular robots with adaptive kinematic structure. Mechatron. Autom. Control **3** (2008)
3. Lokhin, V.M., Manko, S.V., Diane, S.A.K., Panin, A.S., Alexandrova, R.I.: Mechanisms of intellectual feedbacks, knowledge processing and self-learning in control systems of autonomous robots and multiagent robotic groups. Mechatron. Autom. Control **16**(8) (2015)
4. Nikolayenko, S.N., Tulupyev, A.L.: Self-Learning Systems. MTSNMO Publ, Moscow (2009)
5. Makarov, I.M., Lokhin V.M., Manko S.V., Kadochnikov M.V., Sitnikov M.S.: Use of genetic algorithms for automatic knowledge base formation of intelligent control system for Autonomous mobile robot. Mechatron. Autom. Control **6** (2008)

6. Makarov, I.M., Lokhin, V.M., Manko, S.V., Kadochnikov, M.V., Vostrikov, G.S.: The use of genetic algorithms in the problems of automatic learning and self-organization of intelligent robotic systems. Mechatron. Autom. Control **9** (2008)
7. Makarov, I.M., Lokhin, V.M., Man'ko, S.V., Romanov, M.P.: Automation of the Synthesis and the Learning of Intelligent Control Systems. Nauka Publ, Moscow (2009)
8. Lokhin, V.M., Manko, S.V., Diane, S.A.K., Panin A.S., Aleksandrova R.I.: Self-learning mechanisms in multiagent robotic groups based on evolutionary tree forest classification. Mechatron. Autom. Control **18**(3) (2017)
9. Tsetlin, M.L.: Studies on the Theory of Automata and Modeling of Biological Systems. Science Publ, Moscow (1969)
10. Varshavsky, V.I., Pospelov, D.A.: The Orchestra Plays without a Conductor: Reflections on the Evolution and Control of Some Technical Systems. Science Publ, Moscow (1984)