

CONTEXT-AWARE TASK SEQUENCE PLANNING FOR AUTONOMOUS INTELLIGENT SYSTEMS

*Botchkaryov Alexey**Lviv Polytechnic National University, 12, S. Bandery Str., Lviv, 79013, Ukraine
Computer Engineering Department**Submitted on 01.12.2018*

© Botchkaryov A., 2018

Abstract. The problem of context-aware task sequence planning by an autonomous intelligent system (intelligent agent) for a case of independent or loosely coupled tasks is considered. The principle of matching the task to the context was analyzed, the structure of the task sequence planning module and the algorithm of its work were proposed. The paper also proposes an algorithm for calculating the dynamic priority of a task, an algorithm for determining whether a task meets context, and an algorithm for adapting a set of rules for matching tasks to a context based on reinforcement learning in a stationary random environment with context dependence (contextual multi-armed bandit problem). The outline of the reinforcement learning procedure, implemented in the prototype of the task sequence planning module, has been proposed.

Index Terms: context awareness, task sequence planning, autonomous intelligent system

I. INTRODUCTION

One of the main directions in developing the ideas and technologies of artificial intelligence is the creation of autonomous intelligent systems (intelligent agents) capable to solve the tasks assigned to them independently without human intervention [1–3]. Examples of intelligent agents are autonomous robots, unmanned aerial vehicles, unmanned underwater vehicles, software agents, intelligent personal assistants (virtual assistants), etc. The development and use of intelligent agents is the key area in the research and development of cyber-physical systems [4–6]. Modern advances in computer technology, wireless technology, mobile computing and other areas make it possible to implement more and more complex architectures of intelligent agents and to increase their functionality more and more. As a result, situations often arise when an intelligent agent is faced with several different tasks at the same time, and he needs to choose the sequence of their execution. One of the first areas in which researchers encountered this problem was the development of autonomous robots for assembly production [7–10]. At the same time, depending on the specifics of the tasks and the nature of the relationships between them, different tasks of optimizing their sequence can be considered. Among them, task sequence planning, taking into account the state of the agent's environment and the observed dynamics of its changes, is particularly relevant. In this case, it is assumed that the agent is able to determine which of the tasks in these conditions is most advantageous to perform before the

others, and to form the order in which the tasks are carried out accordingly. In other words, planning in this case is context-aware and may be the subject of research within the framework of a more general concept of context-aware computing [11–16].

This paper considers the problem of context-aware task sequence planning by an autonomous intelligent system (intelligent agent) for a case of independent or loosely coupled tasks. In particular, the principle of matching the task to the context was analyzed, the structure of the task sequence planning module and the algorithm of its work were proposed. The paper also proposes an algorithm for calculating the dynamic priority of a task, an algorithm for determining whether a task meets context, and an algorithm for adapting a set of rules for matching tasks to a context based on reinforcement learning in a stationary random environment with context dependence (contextual multi-armed bandit problem). The outline of the reinforcement learning procedure, implemented in the prototype of the task sequence planning module, has been proposed. It should be noted that the results obtained can be generalized to the case of task sequence planning to be performed by a multi-agent system, including taking into account the distribution of individual tasks between agents.

II. THE PROBLEM OF CONTEXT-AWARE PLANNING

Consider the set of independent tasks $Q=\{q\}_n$ that need to be performed by the intelligent agent. We assume that each task q is characterized by a non-empty set of attributes from the set of all attributes $X=\{x\}_h$. Examples of attributes include the type of resource needed to complete the task, the estimate of the time spent on the task, the need for user interaction to complete the task, etc. Based on the concept of context-aware computing, we consider the context model as a set of context parameters $C=\{c\}_m$. Each context parameter can take one of several specified values. Examples of context parameters can be indications of hardware and software sensors, information about the location of an intelligent agent in space, current time, information about the availability of a resource, etc. In this case, depending on the field of application of an intelligent agent, context models different in structure and content can be used [11,

13, 15]. Taking into account the fact that at each moment of time all parameters $\{c\}_m$ take some values, we will consider the context space (S, d) , where $S = \{s\}$ is the set of context vectors, and d is the metric defined on this set $d(s_i, s_j) = \|s_i - s_j\|$ (in general, Euclidean metric). In the case of a multi-agent system [3], one can additionally consider the joint context of several agents, for the determination of which it is necessary to provide the possibility of information exchange between agents.

The basis of the context-aware planning is the principle of matching the task to the context. It is understood that between the attributes of the tasks and the values of the context parameters there is some connection that allows one to distinguish the tasks in terms of their greater or lesser accordance with the conditions in which the intelligent agent is currently located. It is convenient to present this connection as a set of rules for matching tasks to the context $U = \{u\}_k$. Each rule u describes a separate aspect of matching tasks to context as a combination of the values of context parameters from C and the corresponding combination of attributes of tasks from X . The more rules indicate that the task q corresponds to the context vector $s(t)$, the more advantageous it is to perform this task before others under the given conditions. In the baseline scenario, the developer sets the set of rules U to the agent before starting its work. In this case, it is advantageous to provide the possibility of subsequent adaptation of the set of rules U to the specifics of the tasks performed by the agent in some unknown to the developer work conditions. The adapted set of rules for matching tasks to the context in step t will be denoted as $U_t \subseteq U$. Additionally, one can consider an interesting opportunity to let the intelligent agent to form the rules $\{u\}$ by itself (in autonomous mode) and check them in the process of completing tasks. That is, it is possible to formulate the problem of autonomous exploration of the correspondence of tasks to the context, including in the mode of multi-agent interaction. To solve this problem, intelligent data collection technologies and methods for solving problems of distributed autonomous explorations [17-21] can be successfully applied.

As a result of task sequence planning, we will consider an ordered list (set) of relevant tasks $Q_A = \{q\}_z$, $1 \leq z \leq n$, $Q_A \subseteq Q$. The higher the task in the Q_A list, the more profitable it is to perform it in this conditions. Thus, the problem of context-aware planning is to form a list of relevant tasks Q_A from the set Q according to an adapted set of rules U_t and in accordance with the context vector $s(t)$.

III. THE TASK SEQUENCE PLANNING MODULE

To solve the stated problem, it is advantageous to single out a separate task sequence planning module in the structure of an intelligent agent. In the structure of the task sequence planning module itself (Fig. 1), there are three main groups of blocks: 1) management of tasks, including the addition/removal of tasks, the formation of new tasks, the calculation of dynamic priorities of tasks and, in fact, planning the task sequence; 2) management

of the context, including the determination of the values of the context parameters using sensor subsystems, as well as by combining information from sensors (sensor fusion), and the formation of the context vector $s(t)$, based on a given context model C ; 3) matching tasks to the context, including the addition/removal of rules for matching tasks to the context, and the formation of an adapted set of rules U_t by evaluating the performance of an intelligent agent for a given list of relevant tasks Q_A and execution of the reinforcement learning procedure.

IV. THE TASK SEQUENCE PLANNING

The task sequence planning module performs the algorithm of planning (Fig. 2) taking as input the set of tasks and the context model. The idea of the algorithm is that for each task q its dynamic priority $P_d(q)$, $P_d(q) \in [0, 1]$ is calculated, after which the tasks are sorted in descending order of dynamic priority, and the first z of them form the list of relevant tasks Q_A . In this case, the value of dynamic priority depends on two values: 1) the static priority of the task $P_s(q) = \text{const}$, $P_s(q) \in [0, 1]$, assigned at the moment of adding the task q to the set Q , 2) the degree of relevance of the task to the context $R(q) \geq 0$ taking into account the current context vector $s(t)$. In this case, two modes of starting the algorithm are allowed: 1) upon the request of an intelligent agent for a new task, 2) upon changing the context vector. Also from the point of view of forming a list of relevant tasks Q_A , one can consider two options: 1) the first z tasks with the highest dynamic priority are added to the Q_A list; 2) the tasks whose dynamic priority is greater than the specified threshold value are added to the Q_A list. There are three options for the agent to use the list of relevant tasks: 1) the agent selects the task to be performed from Q_A with probability $1/z$, 2) the agent selects a task from Q_A with a maximum dynamic priority (the first task in the list), 3) the combination of the first and second option.

V. CALCULATION OF THE DYNAMIC PRIORITY OF THE TASK

The dynamic priority of the task q is calculated by the formula:

$$P_d(q) = (1 - \lambda(s, n))P_s(q) + \lambda(s, n)[R(q)/(R(q)+1)],$$

where $\lambda(s, n) \in [0, 1]$ is the degree of influence of contextual dependence on dynamic priority as a function of the context vector $s(t)$ and the current number of tasks n . The value of $\lambda(s, n)$ allows one to adjust the influence of the degree of relevance of the task to the context $R(q)$ on the place of the task in the Q_A list. For example, in calculating the value of $\lambda(s, n)$, you can implement the following principle: the more tasks in Q , the greater the influence of contextual dependence on the dynamic priority of tasks. In addition, for specific context vectors that are equivalent to some extreme external conditions, one can set $\lambda(s, n)$ to the maximum value. Since $R(q)$ can take the values $R(q) > 1$, in the second term of the formula its normalization is performed.

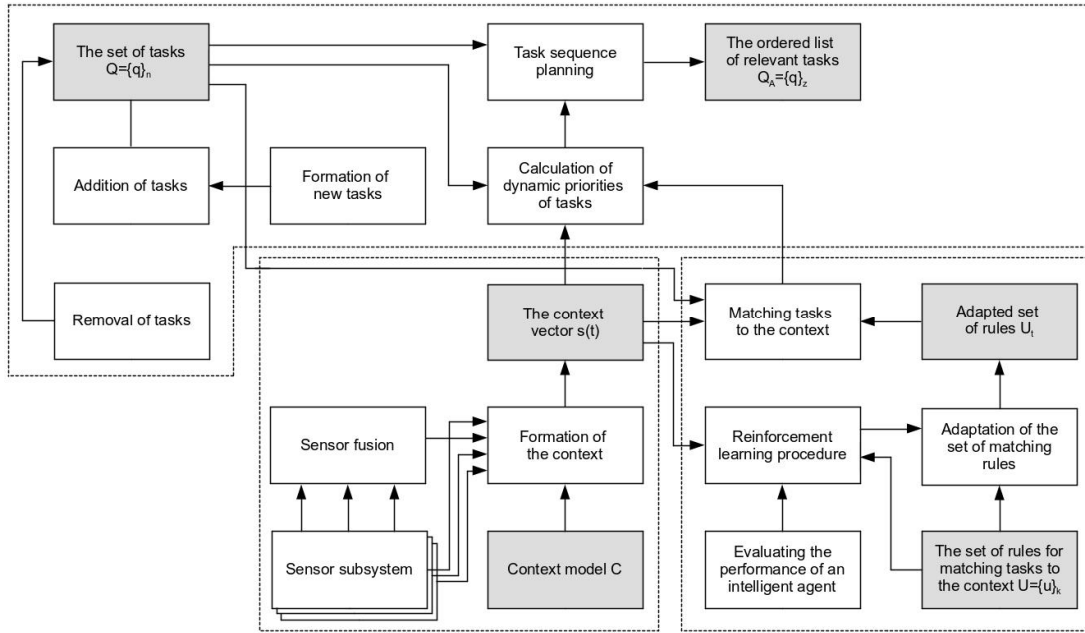


Fig. 1. The structure of the task sequence planning module

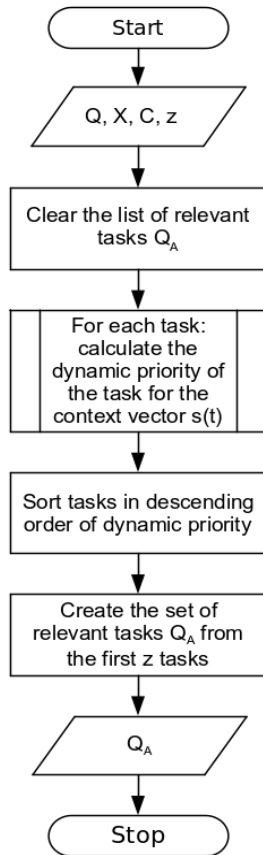


Fig. 2. Algorithm of context-aware task sequence planning (A1)

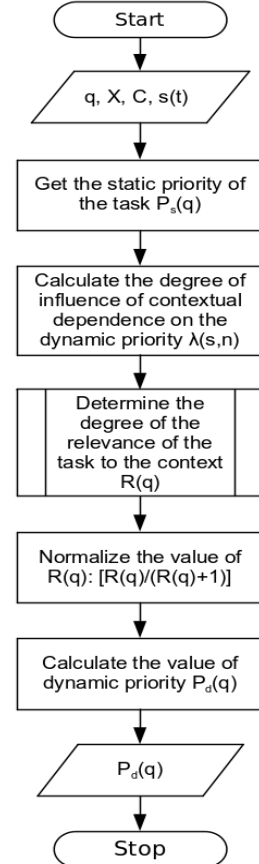


Fig. 3. Algorithm for calculating the dynamic priority of a task (A2)

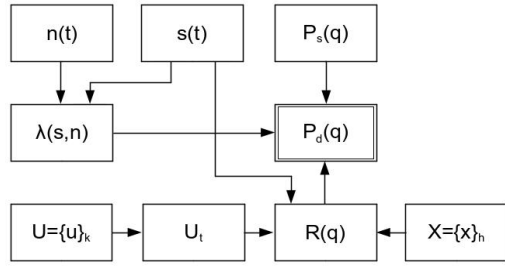


Fig. 4. The scheme for calculating the dynamic priority of the task

The algorithm for calculating the dynamic priority (Fig. 3) is performed independently for each task q of Q . Thus, its implementation as a program has a great potential for parallelization. The scheme for calculating the dynamic priority of the task (Fig. 4) also shows the possibility of parallelization inside the single dynamic priority calculation using the algorithm A2.

VI. DETERMINING THE RELEVANCE OF TASKS TO THE CONTEXT

To determine the degree of the relevance of tasks to the context, the set of all rules $U=\{u\}_k$ for matching tasks to the context is used. Every rule u has the following form

u: if $T_s(c)$ then for all $q \in Q$ with $T_a(x)$ do $R(q)=R(q)+1$,

where $T_s(c)$ is a Boolean expression, whose elements are pairs of values of the form $c_i=v$; $T_a(x)$ is a Boolean expression whose elements are x . Thus, if the combination of the values of the context parameters $T_s(c)$ is true, then for all tasks for which the combination of task attributes $T_a(x)$ is also true, the degree of relevance of the task to the context will be increased by one. The use of Boolean expressions in $T_s(c)$ and $T_a(x)$ makes it possible to map virtually any aspect of matching tasks to context. Using the set of attributes X allows one to specify the semantics of the dependence of tasks on the context. The choice of attributes of tasks and the method of forming the set X is a separate interesting issue.

The idea of the algorithm for determining the relevance of tasks to the context (Fig. 5) is that the more rules u “worked” for a task q , the greater will be its degree of relevance to the context. To determine whether the tasks correspond to the context in step t , an adapted set of matching rules $U_t \subseteq U$ is used. The result of the algorithm A3 is the degree of relevance of the task to the context $R(q)$ for each q .

VII. ADAPTING A SET OF RULES FOR MATCHING TASKS TO A CONTEXT

The problem in consideration is characterized by the fact that in most cases several uncertainty factors are present in the work of an intelligent agent [1]. This is, first of all, the unknown to the developer in advance external conditions of the agent's operation, in which it performs task sequence planning. Additional factors of uncertainty

are: 1) the dynamics of change of the set Q (deletion of some and the addition of other tasks at “random” points in time); 2) the unknown in advance parameters and attributes of new tasks that are added to Q ; 3) the dynamics of changes in the set of matching rules U from the side of entities external to the agent and a number of other factors. The influence of uncertainty factors can be reduced by adapting the set of matching rules U to the external conditions and partially to “internal” changes, based on the performance metric of an intelligent agent. One of the most promising approaches in this case is the use of machine learning, and in particular the methods of reinforcement learning [22–26].

In this paper, to adapt the set of rules for matching tasks to a context, we used a method of reinforcement learning in a stationary random environment with contextual dependence (contextual multi-armed bandit problem) [22, 27–30]. This method is applied separately for each rule u , that is, for each rule u , its own instance of the reinforcement learning procedure is executed (a total of k instances). As context for all instances of the reinforcement learning procedure, the general context C and the corresponding context space (S, d) are considered.

Using the metric d , the context space is divided into domains $\{S_d\}_M$. The two context vectors s_i and s_j fall into the same domain, provided that

$$d(s_i, s_j) = \|s_i - s_j\| < \mu.$$

The value of μ determines the size of the domain and the total number of domains M . The reinforcement learning procedure of each rule u is further divided into M instances by the number of domains. Thus, for each domain S_d and rule u , its own instance of the reinforcement learning procedure is executed (a total of kM instances).

The reinforcement learning method works as follows (Fig. 6). For each matching rule u , the set of actions $A=\{a\}_L$ is defined. Each action a_i determines the certain probability of applying the rule u in step t (that is, the rule u from the set U is selected to the set U_t at step t with probability $p_i(u)$):

$$a_i: p_i(u) = (i-1)/(L-1), i=1, \dots, L.$$

The performance metric of an intelligent agent obtained for the list of relevant tasks Q_A formed on the basis of the adapted set of rules U_t is used as a reinforcement r_t for selecting the action a_t at step t . The choice of the action a_i determines the frequency with which the rule u is applied in the corresponding domain of the context space S_d . The use of the reinforcement learning procedure allows us to find the frequency of applying the rule u in the domain S_d , which maximizes the average value of the agent's performance metric for the pair (u, S_d) .

In each instance of the reinforcement learning procedure, for each selected action a , its value is recalculated:

$$V_{t+1}(a) = V_t(a) + \alpha[r_t - V_t(a)],$$

where $\alpha = \text{const}$ is a step-size parameter, $\alpha \in (0, 1]$. To select the next action, the Upper-Confidence-Bound (UCB) method [22, 31, 32] is used:

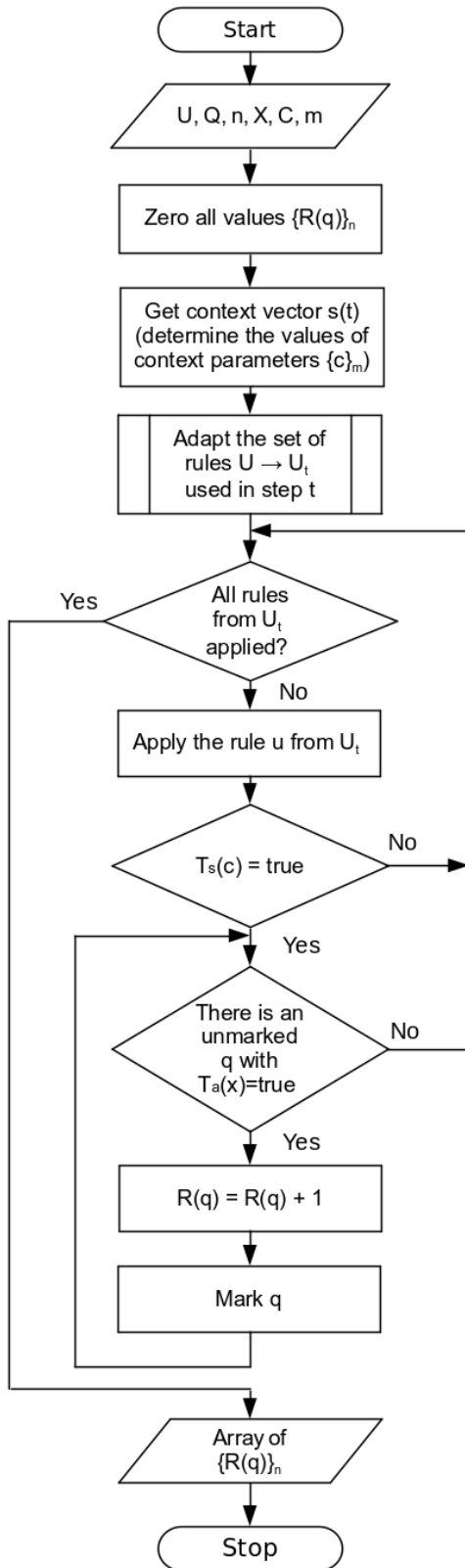


Fig. 5. Algorithm for determining the relevance of tasks to the context (A3)

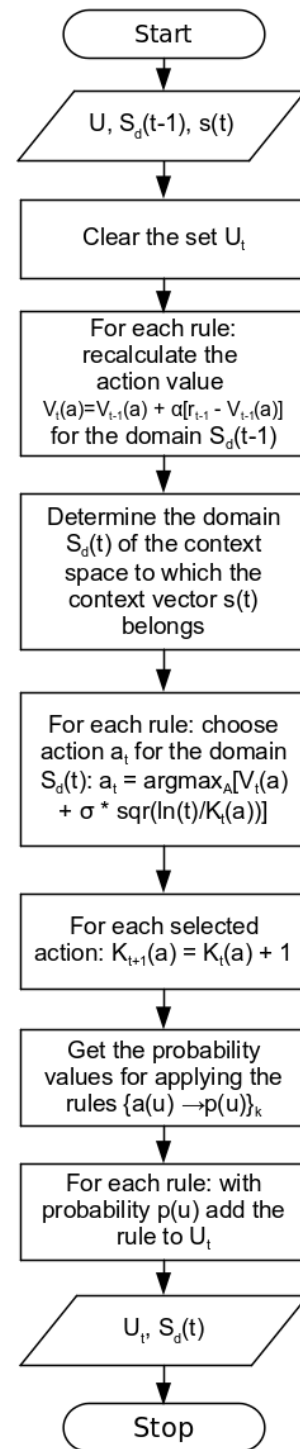


Fig. 6. Algorithm of adapting a set of rules for matching tasks to a context (A4)

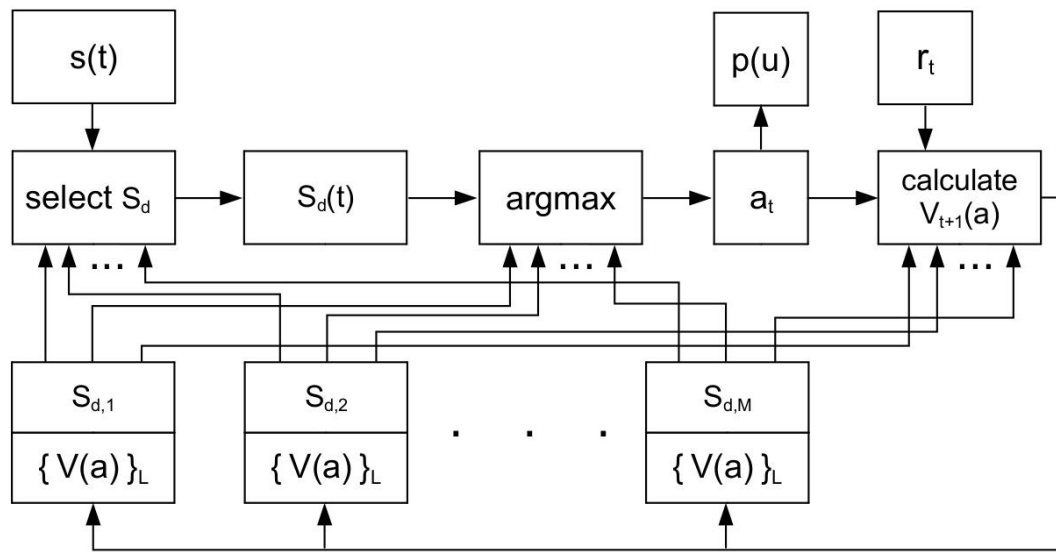


Fig. 7. The outline of the reinforcement learning procedure

$$a_t = \operatorname{argmax}_A [V_t(a) + \sigma \cdot \operatorname{sqr}(\ln(t) / K_t(a))],$$

where $\sigma > 0$ is the parameter that defines the degree of exploration, $K_t(a)$ is the number of times the action a was selected prior to time t .

The outline of the reinforcement learning procedure (Fig. 7), implemented in the prototype of the task sequence planning module, allows one to use data decomposition ($S_{d,i}; \{V(a)\}_L$) and functional decomposition (select S_d , argmax , calculate $V(a)$) to parallelize the corresponding calculations.

VIII. CONCLUSIONS

The problem of context-aware task sequence planning by an autonomous intelligent system (intelligent agent) for a case of independent or loosely coupled tasks is considered. In particular, the principle of matching the task to the context was analyzed, the structure of the task sequence planning module and the algorithm of its work were proposed. The paper also proposes an algorithm for calculating the dynamic priority of a task, an algorithm for determining whether a task meets context, and an algorithm for adapting a set of rules for matching tasks to a context based on reinforcement learning in a stationary random environment with context dependence (contextual multi-armed bandit problem). The outline of the reinforcement learning procedure, implemented in the prototype of the task sequence planning module, has been proposed.

Part of the results presented in this paper were obtained within the framework of the research project DB/CIBER (registration number 0115U000446), 01.01.2015 – 31.12.2017, funded by the Ministry of Education and Science of Ukraine.

REFERENCES

[1] Stuart Russell, Peter Norvig, *Artificial Intelligence: A Modern Approach*, 3rd Ed., Pearson, 2009. – 1152 p.

[2] David L. Poole, Alan K. Mackworth, *Artificial Intelligence: Foundations of Computational Agents*, 2nd Ed., Cambridge University Press, 2017. – 820 p.

[3] *Multiaгент Systems*, by Gerhard Weiss (Editor), 2nd Ed., The MIT Press, 2013. – 920 p.

[4] Melnyk A., *Cyber-physical systems: problems of creation and directions of development*, Transactions on Computer systems and networks, Lviv Polytechnic National University Press, No. 806, 2014. – pp. 154–161 (in Ukrainian)

[5] Melnyk A., Integration of the levels of the cyber-physical system, Transactions on Computer systems and networks, Lviv Polytechnic National University Press, No. 830, 2015. – pp. 61–67 (in Ukrainian)

[6] Golemb V., Botchkaryov A., *Approaches to the construction of conceptual models of cyber-physical systems*, Transactions on Computer Science and Information Technology, Lviv Polytechnic National University Press, No. 864, 2017. – pp. 168–178 (in Ukrainian)

[7] Tierney K., Browne J., *Task sequence planning*, In: Bernhardt R., Dillman R., Hörmann K., Tierney K. (eds) *Integration of Robots into CIM*. Springer, 1992. – pp. 36–44

[8] Tiehua Cao, Arthur C. Sanderson, *Intelligent Task Planning Using Fuzzy Petri Nets*, World Scientific, 1996. – 192 p.

[9] J. Rosell, N. Munoz and A. Gambin, “Robot tasks sequence planning using Petri nets”, Proceedings of the IEEE International Symposium on Assembly and Task Planning, 2003., Besancon, France, 2003, pp. 24–29.

[10] L. Zhang et al., “Adaptive quantum genetic algorithm for task sequence planning of complex assembly systems”, in *Electronics Letters*, vol. 54, no. 14, 2018. – pp. 870–872.

[11] Schilit B., Adams N., Want R., *Context-aware computing applications*, in Proceedings of the IEEE Workshop on “Mobile Computing Systems and Applications”, IEEE Computer Society, 1994. – pp. 85–90.

[12] Abowd G.D., Dey A.K., Brown P.J., Davies N., Smith M., Steggle P. (1999) *Towards a Better Understanding of Context and Context-Awareness*. In: Gellersen H.W. (eds) *Handheld and Ubiquitous Computing*. HUC 1999. Lecture Notes in Computer Science, vol 1707. Springer, Berlin, Heidelberg. – pp. 304–307

[13] Cristiana Bolchini, Carlo A. Curino, Elisa Quintarelli, Fabio A. Schreiber, and Letizia Tanca. 2007. *A data-oriented survey of context models*. *ACM SIGMOD Record*, 36, 4 (December 2007), pp. 19–26.

[14] C. Perera, A. Zaslavsky, P. Christen and D. Georgakopoulos, “Context Aware Computing for The Internet of Things: A Survey”, in *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, First Quarter 2014, pp. 414–454.

- [15] Grifoni P., D'Ulizia A., Ferri F., *Context-Awareness in Location Based Services in the Big Data Era*, In: Skourletopoulos G., Mastorakis G., Mavromoustakis C., Dobre C., Pallis E. (eds) *Mobile Big Data. Lecture Notes on Data Engineering and Communications Technologies*, Springer, vol 10, 2018. – pp. 85–127
- [16] Nicholas Capurso, Bo Mei, Tianyi Song, Xiuzhen Cheng, *A survey on key fields of context awareness for mobile devices*. *Journal of Network and Computer Applications*, Volume 118, 2018. – pp. 44–60.
- [17] Botchkaryov A., Golembo V., *Applying intelligent technologies of data collection to autonomous cyber-physical systems*, *Transactions on Computer systems and networks*, Lviv Polytechnic National University Press, No. 830, 2015. – pp. 7–11 (in Ukrainian)
- [18] Melnyk A., Golembo V., Botchkaryov A., *The new principles of designing configurable smart sensor networks based on intelligent agents*, *Transactions on Computer systems and networks*, Lviv Polytechnic National University Press, No. 492, 2003. – pp. 100–107 (in Ukrainian)
- [19] Botchkaryov A., *Collective behavior of mobile intelligent agents solving the autonomous distributed exploration task*, *Transactions on Computer systems and networks*, Lviv Polytechnic National University Press, No. 546, 2005. – pp. 12–17 (in Ukrainian)
- [20] Botchkaryov A., *Structural adaptation of the autonomous distributed sensing and computing systems*, *Transactions on Computer systems and networks*, Lviv Polytechnic National University Press, No. 688, 2010. – pp. 16–22 (in Ukrainian)
- [21] Botchkaryov A., *The problem of organizing adaptive sensing and computing processes in autonomous distributed systems*, *Transactions on Computer systems and networks*, Lviv Polytechnic National University Press, No. 745, 2012. – pp. 20–26 (in Ukrainian)
- [22] Richard S. Sutton, Andrew G. Barto, *Reinforcement Learning: An Introduction*, 2nd Ed., A Bradford Book, 2018. – 532 p.
- [23] L.P. Kaelbling, Michael L. Littman, and Andrew W. Moore, *Reinforcement learning: A survey*. *Journal of AI Research*, N 4, 1996. – pp. 237–285.
- [24] Csaba Szepesvari, *Algorithms for Reinforcement Learning*, Morgan and Claypool Publishers, 2010. – 104 p.
- [25] Maxim Lapan, *Deep Reinforcement Learning Hands-On*, Packt Publishing, 2018. – 546 p.
- [26] H. M. Schwartz, *Multi-Agent Machine Learning: A Reinforcement Approach*, Wiley, 2014. – 256 p.
- [27] Langford, John; Zhang, Tong (2008), "The Epoch-Greedy Algorithm for Contextual Multi-armed Bandits", *Advances in Neural Information Processing Systems* 20, 2008. – pp. 817–824.
- [28] Lu T, Pál D, Pál M., *Contextual multi-armed bandits*, *In Proceedings of the Thirteenth international conference on Artificial Intelligence and Statistics*, 2010 Mar 31. – pp. 485–492.
- [29] Djallel Bouneffouf, Amel Bouzeghoub, Alda Lopes Gançarski, "A Contextual-Bandit Algorithm for Mobile Context-Aware Recommender System", *Neural Information Processing – 19th International Conference, ICONIP 2012, Doha, Qatar, November 12–15, 2012, Proceedings, Part III, Lecture Notes in Computer Science*, 7665, Springer. – pp. 324–331.
- [30] Kumpati S. Narendra, Mandayam A. L. Thathachar, *Learning Automata: An Introduction*, Dover Publications, 2012. – 496 p.
- [31] Auer, P. *Using upper confidence bounds for online learning*. *Proceedings 41st Annual Symposium on Foundations of Computer Science*. IEEE Comput. Soc., 2000. – pp. 270–279.
- [32] Auer, P., Cesa-Bianchi, N., Fischer, P., *Finite-time analysis of the multiarmed bandit problem*. *Machine learning*, 47(2-3), 2002. – pp. 235–256.



Alexey Botchkaryov was born in 1975 in Lviv, Ukraine. He received the B.S. and the M.S. degrees in computer engineering from the Lviv Polytechnic National University in 1996 and 1998. He has been doing scientific and research

work since 1994. His work resulted in 79 publications and master thesis. Currently, he is a Senior Lecturer with the Computer Engineering Department and a Research Assistant with the Intelligent Systems Laboratory, Lviv Polytechnic National University. His research interests include intelligent data collection technologies and multi-agent systems.