Analysis of Semantic Probabilistic Inference Control Method in Multiagent Foraging Task

Vitaly Vorobiev, Maksim Rovbo
National Research Center «Kurchatov Institute»
Moscow, 123182, Russia
Vorobev_VV@rrcki.ru, rovboma@gmail.com

***Abstract*—Adaptation in robotics systems is often implemented as some form of learning. While much research is dedicated to studying policy and value approximation in reinforcement learning, some methods are based on rule inference and logical descriptions. One of these methods is based on a semantic probabilistic inference algorithm that has its roots in the theory of functional systems. In this article, the method is applied to a distributed multiagent foraging problem that has an important property of providing an environment that allows to study a decentralized system of individually learning agents. We compare the performance of this method to other methods: Q–learning and a random choice algorithm as a baseline. We also propose a modification of the algorithm that includes an exploration behavior. Experiments are carried out in a computer simulation system. The results show the performance of the algorithms with different parameters, as well as the effect of exploration on the performance.**

***Keywords*—adaptive control, robotics, semantic probabilistic inference, foraging, local interaction**

I. INTRODUCTION

Adaptive control system for robotics are of practical interest since they promise to increase robustness of existing systems, make the behavior closer to optimal as well as introduce the possibility to impart new behaviors to the robot by a system of rewards or examples. This may be especially important for multiagent systems as controlling them in a direct way to achieve a given goal is harder than single robots.

A lot of current research is dedicated to learning methods for virtual and robotic agents that is based on reinforcement learning methods using value and policy approximations, especially based on parametric descriptions of the functions and neural networks. Multiagent aspect introduces event more problems, like operating in a dynamic, non-markovian environment that makes even a static environment more challenging due to the activity of the agents themselves in relation to other agents. Robots often work in an environment, where only some information about the state is accessible, which means making decisions in a partially observable environment. Thus, seeking efficient ways to search the policy space for acceptable (and, preferably, optimal) observationaction mappings is important.

One of the ways to address this problem it to seek biologically inspired models of decision making or using different representations of the problem and policy space, such as logical. There are various approaches and methods that use logical descriptions that could be used for decision making, for example, semiotic networks [10], JSM method [2], semantic probabilistic inference [7], [9].

One of the main goals of this work was to study and compare capabilities of the SPI method and some reinforcement learning methods in a multiagent setting with physically distributed agents and also the effects of exploration and some other, problem-specific parameters, on the agents' performance. While SPI was used as a basis of a network composed of logic neurons and studied in a multiagent context in [1], in those works agents controlled tightly coupled (physically connected) elements of a robot, used a common reward from a centralized source and inferred the rules in a single system that could create rules specific to each agent, as well as general rules. In this work we emphasize that the studied system does not provide agents with a common reward (each reward is specific to the agent), the agents have only an indirect effect on each other's performance and they do not have to use a centralized rule-based learning system, but can learn separately from each other

The chosen problem environment is a foraging problem, where agents must gather food units in a grid world since it can be seen as a reasonably representative problem for some simple group robotics tasks and it satisfies the environment requirements described above. We also propose

a modification of the SPI algorithm that introduces exploration behavior into the system so that the agent is less susceptible to local minima of performance, especially in a stochastic environment.

On the other hand, an important issue is the question of the application of the logical system of adaptive control, which is based on the algorithm of semantic probabilistic inference, to a group of mobile robots that allows local interaction. In this regard, it is proposed to consider the possibility of using such a control system for a group of robots that solve some common task. In this case, the main emphasis is placed not on the solution of the common task but on resolving the problem of organizing communication and capabilities of separated and moving robots.

The work is structured as follows. Firstly, methods and algorithms used in the paper are presented, as well as the proposed modification. Then the model of the problem is described. After that, the experiment parameters, simulation results and the analysis follow. Then, in the section "Organization of a group of robots for collective application of the logical model of an adaptive control system" further research is described, detailing the problems that need to be solved and a proposed approach to adapt the semantic probabilistic inference for a physically distributed group of mobile robots. Finally, a conclusion sums up some key points about the article.

II. METHODS AND ALGORITHMS

The main algorithm that is studied in this work is the semantic probabilistic inference that is described in [9] but without the mechanism of new functional system formation as it was observed in the original work that for a foraging problem forming new functional systems is not required. The algorithm also uses the concept of a goal predicate, but in the later works [1] a reward was used as a prediction, which is what we use here, but without creating logic neurons described in the latter work. The reward predicted by the rules always equals to one, so it can be written as a goal predicate that states that the agent gets a reward of one.

We also propose a modification of the algorithm by introducing an exploration behavior into the system. The original algorithm chooses a random action only in cases where the situation was never encountered before and / or there were no suitable rules inferred from the experience. Instead, we also add a random possibility of choosing an action randomly with uniform probability that is governed by an exploration rate ϵ. This is similar to exploration done by the Q-learning algorithm and should help the agent gather information about alternative choices of actions in situations that already have a suitable rule. There are also cases where such exploration strategies were successfully applied to foraging problems [6], so it seems reasonable to try it for the SPI algorithm in a multiagent setting.

The following is a short version of the SPI algorithm as it is implemented in this work. The proposed modification is marked by an asterisk and is basically everything that uses ϵ.

1) Parameters of the algorithm basic_rule_depth *brd*, max_plan_length *mpl* are set, the environment and agents are initialized.

2) Agent receives an observation *obs* from the environment, which includes the reward r from the previous action

3) Agent updates its experience table (called here *spi_table*) by adding 1 to the record describing a combination of the last state s_{last}, sequence of last actions taken $asec_{last}$ and the resulting reward r (0 or 1):
$$spi\_table[s_{last}, asec_{last}, r] += 1$$

4) Rules for regularities detection are created by exploring a graph that has rule of the following form as its nodes:
$$P_1 \wedge P_2 \wedge \ldots \wedge A_1 \wedge \ldots \wedge A_{mpl} \rightarrow r$$
which contain state predicates (P_1 can be, for example, a fact "the left cell has food") and a sequence of actions A_i, in the precondition and a predicted reward r in the postcondition that always equals 1 (otherwise the rule would never be applied). Nodes are explored in two steps. Firstly, all possible rules with no more than *brd* predicates in the precondition including action predicates and no less than one action are built by expanding a node with a single new predicated added to the preconditions. During the second stage, only the rules that pass a positive rule regularity check are expanded. The first node is a $rule \rightarrow r$.

5) Positive regularity check for a rule means that its estimated probability to yield a reward r is higher than that of any subrule that can be formed with a subset of its preconditions. Only rules that pass a positive regularity check are added to the list of *regularities* for decision making. The probability check is the following inequality:
$$\frac{n(P_{rule} \wedge A_{rule} \wedge r)}{n(P_{rule} \wedge A_{rule})} > \frac{n(P_{subrule} \wedge A_{subrule} \wedge r)}{n(P_{subrule} \wedge A_{subrule})}$$
where P_{rule} — state preconditions of the rule, A_{rule} — action preconditions (planned actions) of the rule,

$P_{subrule}$ — state preconditions of the subset rule, $P_{subrule}$ — action preconditions (planned actions) of the subset rule, n(predicates) — number of times the predicates were applicable to agent's situation stored in its experience table *spi_table*.

6) (*) Exploration action is carried out with ϵ probability, which is a randomly chosen action with a uniform probability and step 9 is performed. Otherwise the usual SPI action selection is applied

7) If exploration action was not chosen, all discovered rules' applicability to current state from the regularities list are checked. If a rule's precondition is satisfied, its performance rule_performance (probability to get a reward following the precondition actions from the current state) is checked according to the formula:

$$rule\_performance(rule, state) = \\ = \frac{p(P_{state} \wedge A_{rule} \wedge r)}{p(P_{state} \wedge A_{rule})}$$

where P_{state} is all predicated describing the current observation (not just the predicated from the rule's precondition). A list of such rules is formed with their performances.

8) The applicable rule with the highest performance is chosen and its first action (if there are several in the rule action plan) is chosen to be performed:

$$chosen\_rule = \\ = \underset{rule}{argmax}(rule\_performance(rule, state)) \\ action = (A_{chosen\_rule})_1$$

9) The chosen *action* is stored as the last action performed, the sequence of length *mpl* of last actions is updated, the current state is remembered as the last state and the action is returned to the environment to be performed.

The SPI method was expected to show relatively high performance, surpassing some classical reinforcement learning algorithms at least on initial episodes, since it can aggregate states from observation space by only deciding on a few variables (predicates) from it. The original (with exploration rate $\epsilon = 0$) SPI also quickly adapts rewarding behavior, but it can be hypothesized that it can fall into a local minima because it does not seek new rules actively for a state that already has a rewarding regularity discovered. Hence the proposal to add an exploration coefficient to it, so that it can sometimes check other actions.

REFERENCES

References

[1] Demin A.V., Vityaev E.E. Adaptive Control of Modular Robots. Biologically Inspired Cognitive Architectures (BICA) for Young Scientists. BICA 2017. Advances in Intelligent Systems and Computing, 2018, vol. 636, pp. 204–212.

[2] Finn V.K. Plausible inferences and plausible reasoning. Journal of Soviet Mathematics, 1991, vol. 56, no 1, pp. 2201–2248.

[3] Karpov V., Karpova I. Leader election algorithms for static swarms. Biologically Inspired Cognitive Architectures, 2015, vol.12, pp. 54–64.

[4] Sutton R.S., Barto A.G. Reinforcement Learning: An Introduction, Cambridge, MA: The MIT Press, 2018, 552 p.

[5] Tokic M. Adaptive ϵ-greedy exploration in reinforcement learning based on value differences. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2010, vol.6359 LNAI, pp. 203–210.

[6] Yogeswaran M., Ponnambalam S.G. Reinforcement learning: Exploration–exploitation dilemma in multi–agent foraging task. Opsearch, 2012, vol. 49, no 3, pp. 223–236.

[7] Vityaev E.E. Printsipy raboty mozga, soderzhashchiesya v teorii funktsional'nykh sistem P.K. Anokhina i teorii emotsii P.V. Simonova [The principles of the brain from the Anokhin's theory of functional systems and P.V. Simov's theory of emotions]. Neiroinformatika [Neuroinformatics], 2008, vol. 3, no 1, pp. 25–78.

[8] Vorobiev V.V. Algoritmy vybora lidera i klasterizatsii v staticheskom roe robotov [Leader choice and clustering algorithms in a static swarm of robots]. Mekhatronika, avtomatizatsiya, Upravlenie [Mechatronics, automation, control], 2017, vol. 18, no 3., pp. 166–172.

[9] Demin A.V., Vityaev E.E. Logicheskaya model' adaptivnoi sistemy upravleniya [Logical model of the adaptive control system]. Neiroinformatika [Neuroinformatics], 2008, vol. 3, no 1, pp. 79–107.

[10] Osipov G.S., et al. Znakovaya kartina mira sub"ekta povedeniya [Symbolic worldview of a subject of behavior]. Moscow, FIZMATLIT, 2017, 259 p.