Multi-agent control framework for multi-wheeled mobile platforms

Uladzimir Dziomin [1], Vladimir Golovko [1] Ralf Stetter [2]
1) Brest State Technical university, Brest, Belarus. gva@bstu.by
2) Hochschule Ravensburg-Weingarten, Weingarten, Germany. stetter@hs-weingarten.de

Abstract: *This paper presents control framework based on multi-agent reinforcement approach for building intellectual steering software for multi-wheeled mobile robots. The framework uses modified reinforcement learning approach based on special multi-agent structure with virtual leader. The framework application example will be shown with real multi-wheeled mobile platform. The experiments were performed in simulation environment with accurate virtual model.*

***Keywords*:** multi-agent systems, reinforcement learning, self-learning systems, robotics, control.

1. INTRODUCTION

A major achievement for any vehicle or mobile robot is the ability to move freely in any direction and performing rotation at any arbitrary position and orientation. This can be realized by obtaining a locomotion system with the ability to perform a holonomic and omnidirectional motion [1]. Both abilities are necessary for realizing better maneuver in a crowded area or restricted space. A non-holonomic omnidirectional mechanism requires preliminary maneuvering for reorientation of steering wheel which may affect the time-travel cost and the spaces needed. The holonomic omnidirectional mechanism has better mobility and also less complexity in design compared with a non-holonomic omnidirectional mechanism [2].

During the last decades several approaches towards better maneuverability, usually in the form of omnidirectional mobile system were presented; Wada provides a good overview [3]. Notable is additionally the ongoing long-term research project "Omni" concerning similar systems [4]. In spite of the fact that several systems were proposed none are today available on the market. One of the causes of this can be the high complexity of the solutions.

The paper stars with a description of the multi-agent control framework architecture. Then the details of learning approach are described. Section after presents the results of the experiments and discuss the further improvements.

2. MULTI AGENT CONTROL FRAMEWORK

The multi-agent control framework provides the *control system* of high quality. The design of multi agent system consists of two agent types: *virtual agent* and *control agent*. *Control agent* – an agent that communicate with hardware of the robot, provides control and receives feedback from the sensors. In the framework architecture control agent is responsible only for one wheel but can combine multiple sensors. *Virtual agent* – an agent that coordinate team of the control agents to provide desirable control. Virtual agent is responsible for learning his agents. The desirable control generates by control system goal.

As an example we took mobile wheelchair platform with three holonomic wheels. The proposed multi-agent architecture for the platform consists of four agents where three of them are the representation of each driving module and the fourth agent is the *virtual head agent*. The *virtual head agent* is the agent that achieves the desired coordinates from the user and sends the set of cooperating commands to the *control agents*. The decomposition of the wheelchair is shown on the Figure 1. As another example the decomposition of the four wheeled mobile robot with holonomic wheels is shown on Figure 2.

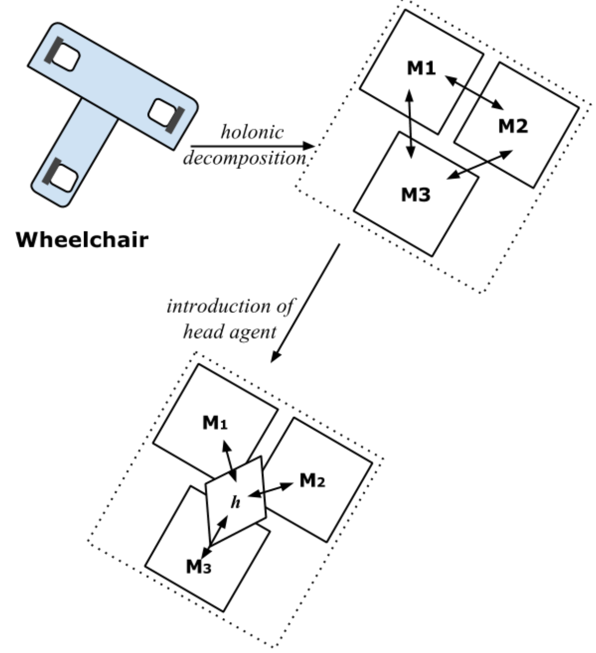

Figure 1. Three wheel-based wheelchair decomposition to multi-agent system with head agent.

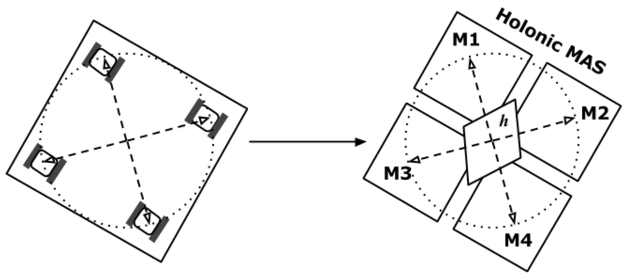

Figure 2. Four-wheeled mobile robot decomposition to multi-agent system with head agent.

To connect agents to real wheel controllers, sensors and other actuators we use special framework Facades of controllers. Controller Façade provides three program interfaces: *actions*, *sensors* and *state*. The software architecture of the control systems with three control agent is shown on Figure 3.

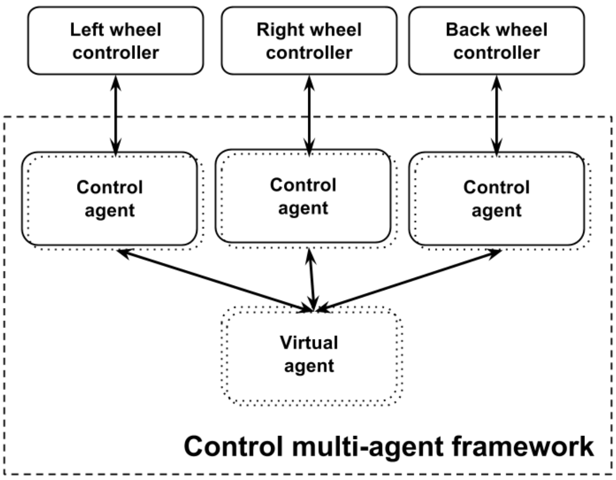

Figure 3. Framework based control architecture for wheelchair mobile robot

The current framework is working on the wheelchair mobile platform (Figure 4). In the next parts all examples and experiments will describe the framework implementation with this robot.

Figure 4. Wheelchair mobile robot

3. REINFORCEMENT LEARNING FOR MULTI-AGENT CONTROL

In previous researches we showed the multi-agent approach mixed with reinforcement learning (RL). The approach improves the control of the mobile robot based on complicated steering scheme [5, 6]. The control was done either for simulation or for real robot hardware. The main problem of such software transfer is that real hardware is not ready to be controlled by agents and needs a lot of development to use it.

To describe multi-agent systems, we need to start from the agent definition. An **agent** is an instance that can be viewed as one that perceives its **environment** through the **sensors** and **acts** upon that environment through the **actuators** [7]. In multi-agent systems the most important receive/perceive ability of the agent is the communication ability. The agents of the system should send messages and answer to each other. This is the main criterion to create multi-agency.

The second important criterion is the intelligence of the agent. An **intelligent** or **rational** agent for each possible percept sequence should select an action that is expected to maximize its performance measure, given the evidence provided by the percept sequence and whatever built-in knowledge the agent has [7]. Our intelligent agents contain reinforcement learning decision-making system. In reinforcement learning agents improve their behavior by exploration of the environment. To become really intelligent agent should make a lot of epochs (couple of iterations). Each epoch the agent starts with a random state and explores the world by making actions. Every action of the agent returns a feedback from the environment as a reward. The reward value the bigger the better. The reward value is saved by the agent to the knowledge base using one of the reinforcement learning rules. This knowledge base is used to help in making choice of the next action in the next epochs. The goal of the agent not to only to use maximum valuable steps from the base but also to investigate not visited ones. In our approach we used the improved multi-agent reinforcement learning algorithm with *shared state*. The description and the framework will be revealed at the end of this chapter.

As it was mentioned above the wheelchair provides two driving modes: *fast mode* as common driving one and the *unlimited maneuvering mode* used when the wheelchair needs to drive along complicated trajectories or in the limited space. So we developed the multi-agent system for working in these two modes.

The general architecture after decomposition step for multi-agent control system with head agent is shown on Figure 5. Every module provides control $P = \{p_i | i = 1..n\}$ of n input parameters. For every control parameter agent is using policy π_i and after every step achieves the reward R_i from environment. The Environment/Planning subsystem provides information about the desired speed of the platform v^d and the global state of the agents $S = \{U_{i=1}^{N} s_i\} \cup \{s_h\}$, where $s_i \in s$ is the state of the i-th module and s_h is the state of the head agent.

The intelligence of our agents is made by reinforcement learning approach that is extended to the multi-agent system. The standard reinforcement learning approaches are not suitable for the multi-agent systems. The curse of dimension becomes a huge problem when we need to deal with more than one agent. In our approach we use an alternative way to learn many agents in one time. The approach uses sharing of the agents' state. The sharing procedure is provided by head agent that sends synchronization command to the other agents. During the process of synchronization agents update their knowledge base due to the Q-Learning reinforcement learning rule.

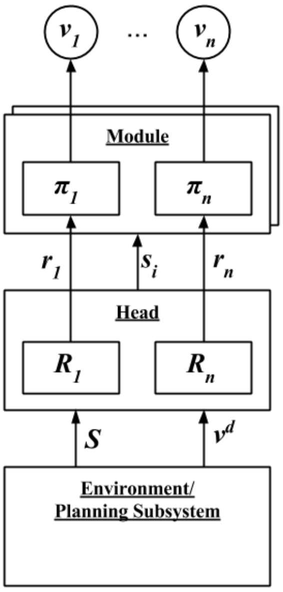

Figure 5. The architecture of the control system.

While reinforcement learning the i^{th} agent executes an action a_i at the current state s_i. Then it goes to the next state s'_i and receives a numerical reward r as the feedback for the recent action [8], where $s_i, s_i' \in s, a_i' \in a, r \in R$. Ideally agents should explore state space (interact with environment) to build an optimal policy π^*.

Let $Q(s, a)$ – represent a Q-function that reflects the quality of the specified action a in state s. Optimal policy can be expressed in the terms of optimal Q-function Q^*:

$$\pi^*(s) = \operatorname{argmax}_{a \in \alpha(s)} Q^*(s, a) \quad (1)$$

The initial values of Q-function are unknown and equal to zero. The learning goal is to approximate the Q-function, (e.g. to find true Q-values for each action in every state using received sequences of rewards). The RL framework with *shared state* is shown on Figure 7.

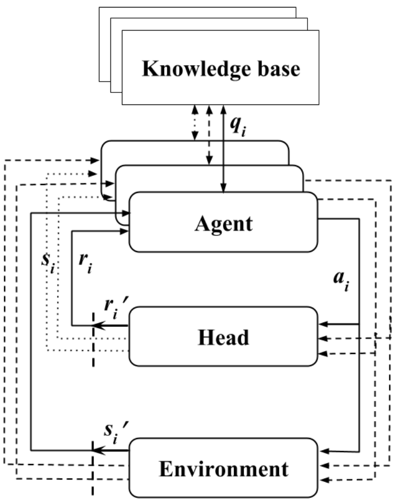

Figure 7. Reinforcement learning framework with shared state.

4. EXPERIMENTS RESULT

The experimental part covers only simulation experiments. We divided the experiments into two parts: the learning part and the driving part that will be described sequentially.

4.1. THE LEARNING PART

At this part agents learn how to work in two provided modes. Every driving agent has the driving scheme that is shown at Figure 8. Two agents at the front of the wheelchair have only one control parameter v – the velocity of the wheel. According to that we learn the agent with one output parameter. The third agent that is placed at the back of the wheelchair has two output parameters: *wheel velocity* v and *the angle relative to the wheelchair* φ. The error of the wheelchair trajectory is an important parameter that is used as reward for the agents. The agents are learnt in team to drive in two modes.

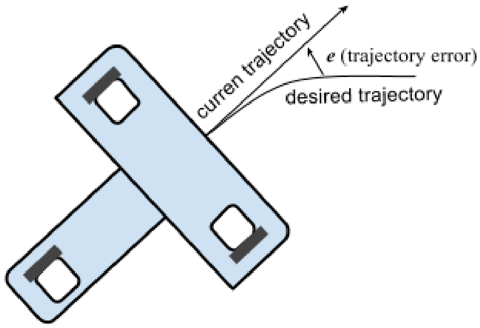

Figure 8. The wheelchair trajectory error calculation.

The driving model was taking from the real mobile robot – wheelchair (Figure 4). The robot was built in laboratory of robotics, university of Weingarten, Germany. For the simulation experiments part, the accurate wheelchair model was created (Figure 9).

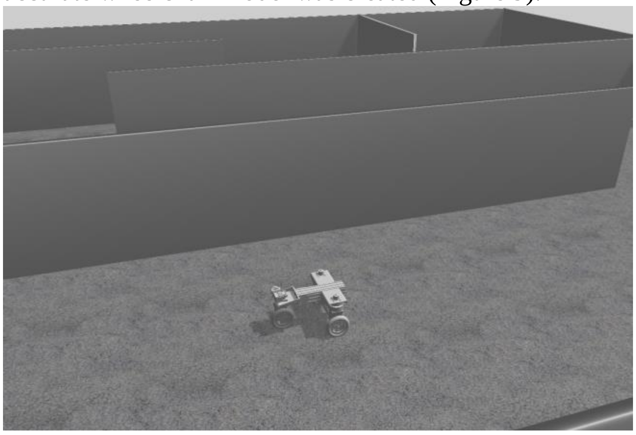

Figure 9. The simulation environment with the base of the wheelchair.

4.2. THE SIMULATION EXPERIMENTS

To test driving abilities of the wheelchair model we create an artificial maze environment by Robotics Operation System. The Gazebo 2.0 was used as a simulator [9].

For the virtual wheelchair we defined three tasks in which we tested it. In Figure 10 we showed the prepared trajectories for the model. The tasks descriptions are:

a) Driving along the straight line in mode one.
b) Driving along the curve line in mode one.
c) Driving along the straight line in mode two.

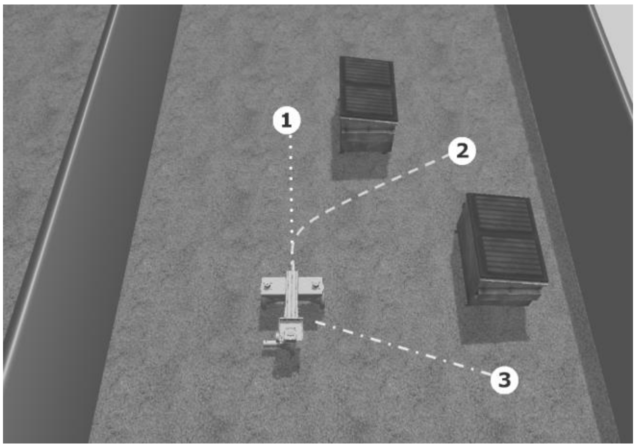

Figure 10. The trajectories of the tasks in driving part of experiments.

The driving experiments results are represented in Table 1. This table consists of the median results that were collected from 20 experiments for each of the tasks.

Table 1. The median of the experiments results in driving task

Task	Velocity, seconds	Trajectory error, %	Goal error, %
Straight driving (1)	1,33	0,7	0,3
Straing curve driving (2)	1,92	1,41	0,3
Back curve side driving (3)	1,87	1,01	0,3

These results are the proof of the quality of the control system, based on multi-agent reinforcement learning architecture. The advantages of this method are adaptability – using reinforcement learning the multi-agent system adapts to control the wheelchair platform through the commands of the head agent; sharing state of the agents decreases the learning time and increases the provided quality of the already learnt system.

5. CONCLUSIONS

The control approaches described in this paper were developed for and tested on a holonomic, omnidirectional wheelchair. Such wheelchairs are easier to use for elderly people but also beneficial for semi-autonomous and autonomous operations. As mentioned above, the framework can solve the control problem efficiently. The experiments were already done with different robot platforms with different configuration [2, 5, 6, 11] but in this paper only last successful wheelchair launch is covered.

6. REFERENCES

[1] Bin Safar, M. J. A.: Tip-over Stability Analysis and Enhancement for Omnidirectional Personal Mobility Robots with Active Dual-wheel Caster Assemblies. Ph. D. Thesis, Okayama University, 2014.

[2] Diomin, U., P. Witczak, and R. Stetter. "Event-based and Multi Agent Control of an Innovative Wheelchair." *Journal of Physics: Conference Series.* Vol. 659. No. 1. IOP Publishing, 2015.

[3] Wada M: Motion Control of a Four-wheel-drive Omnidirectional Wheelchair with High Step Climbing Capability. In: Behnam Miripour (Ed.): Climbing and Walking Robots, 2010.

[4] Borgolte, U.; Hoyer, H.; Bühler, C.; Heck, H.; Hoelper, R.: Architectural Concepts of a Semi-autonomous Wheelchair. Journal of Intelligent and Robotic Systems 22. Kluwer Academic Publishers, 1998; pp. 233–253.

[5] Dziomin, U., Kabysh, A., Golovko, V., & Stetter, R. (2013, September). A multi-agent reinforcement learning approach for the efficient control of mobile robot. In Intelligent Data Acquisition and Advanced Computing Systems (IDAACS), 2013 IEEE 7th International Conference on (Vol. 2, pp. 867-873). IEEE.

[6] Dziomin, U., Kabysh, A., Golovko, V., & Stetter, R. (2014). A Multi-agent Efficient Control System for a Production Mobile Robot. In Neural Networks and Artificial Intelligence (pp. 171-181). Springer International Publishing.

[7] Russell, Stuart, Peter Norvig. Artificial Intelligence. A modern approach. *Artificial Intelligence. Prentice-Hall*, Egnlewood Cliffs 25 (1995): 27.

[8] A.G. Barto. Reinforcement learning: An introduction. MIT press, 1998.

[9] Koenig, Nathan, and Andrew Howard.: Design and use paradigms for gazebo, an open-source multi-robot simulator. Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on. Vol. 3. IEEE, 2004.

[10] Lipnickas A., Kabysh A., Golovko V. Influence Learning for Multi-Agent Systems Based on Reinforcement Learning//International Journal of Computing, Vol.1, N.1, 2012, p. 39-44.

[11] Dziomin, U. A Multi-Agent Reinforcement Learning Approach for the Efficient Control of Mobile Robots / U. Dziomin, A. Kabysh, R. Stetter and V. Golovko // in book Advanced in Robotics and Collaborative Automation. – River Publishers, 2015. – P. 123-146.