Archives of Control Sciences
Volume 24(LX), 2014
No. 3, *pages 257–279*

Handling uncertainty of resource division in multi agent system using game against nature

KRZYSZTOF SKRZYPCZYK and MARTIN MELLADO

This paper addresses the problem of resource division for robotic agents in the framework of Multi-Agent System. Knowledge of the environment represented in the system is uncertain, incomplete and distributed among the individual agents that have both limited sensing and communication abilities. The pick-up-and-collection problem is considered in order to illustrate the idea presented. In this paper a framework for cooperative task assignment to individual agents is discussed. The process of negotiating access to common resources by intercommunicating agents is modeled and solved as a game against Nature. The working of the proposed system was verified by multiple simulations. Selected, exemplary simulations are presented in the paper to illustrate the approach discussed.

Key words: multi-agent system, distributed system, negotiations, uncertainty, Game Theory

1. Introduction

In Multi-Agent Systems (MAS), a primary task the system is intended to perform is distributed between a number of entities (agents). It was recognized that there are a number of tasks MAS can perform more efficiently than centralized, single-unit-based systems. Thanks to the feature of MAS - *modularity*, even the domain of complex problems which are sometimes unpredictable can be solved by a number of simple entities, specialized in solving a particular part of the primary problem [28, 32]. On the other hand, there are a lot of challenges that must be met in order to design effective and robust systems that are able to solve problems or execute tasks. These challenges were discussed in [28] and it is enough to point out the problems like coordination [6, 15, 33], task division [11, 14, 30] etc. The potential advantages of MAS were quickly noticed by researchers who deal with problems related to Robotics. They have discovered that

K. Skrzypczyk, the corresponding author, is with Institute of Automatic Control, Silesian University of Technology, Gliwice, Poland. E-mail: krzysztof.skrzypczyk@polsl.pl. M. Mellado is with Instituto de Automática e Informática Industrial Universidad Politecnica de Valencia, Spain. E-mail: martin@ai2.upv.es.

This study is partially supported by IMPA (Inspection & Manipulation of Food Products) project (ref. DPI2010-20286), funded by the Spanish Government MICINN and by Programa de Apoyo a la Investigación y Desarrollo (PAID-02-11) of the Universitat Politècnica de València.

Received 17.03.2014.

some specific tasks whose execution requires building and operating complex, powerful robotic units, can be performed more effectively by simpler, highly specialized robots. There are a number of problems that have become benchmarks and test-beds for the quality assessment and analysis of Multi-Robot Systems (MRS) functioning. One of them is the exploration problem. It is clear that exploring large areas by multiple mobile robots can be done more effectively than with the use of one single unit. There is an extensive literature on studies that have been done on this subject [1, 4, 22]. Another problem often being implemented in the MAS framework is *pick-up-and-collection* which is similar to the *foraging problem* and consists of the collective searching for and picking up of objects distributed within a constrained or unconstrained workspace. [18]. Another problem that has often been reported in the literature is multi robot formation control. The key issue in this problem is to provide a coordination mechanism for individual robots that enables movement of all the robots along a given path to be obtained while keeping the formation pattern [10, 19, 24, 25]. The successful solution for this problem is the basis for another problem that has been the subject of much research - the transportation of large and heavy objects by a number of tightly coupled and coordinated smaller transporters [16, 27]. Different approaches to modeling MAS have been developed over the years. Starting from the reactive agent based system [3] through to a purely deliberative agent architecture [31] and hybrid systems [2]. Other approaches that are very promising are the Economics-based ones, particularly market mechanisms which are becoming more and more attractive for MAS developers [8, 12].

The one which application is considered in this paper is Game Theory [21, 23]. This seems to be a proper tool for dealing with problems such as coordination in MAS [17, 23]. Therefore, it has been a subject of numerous studies in this area [9, 10, 13, 20].

In contrast to many studies [17, 20, 24] that present the use of the game theoretical framework in a centralized system that synchronously controls the agents' actions, the approach presented in this paper is different. The asynchronous negotiation process between agents is modeled as a game against a virtual, unreasonable player who is referred to as Nature. This player is a personification of the uncertainty that appears during the negotiation process in a distributed system. Since each agent has only limited knowledge of the process, it must take the decision on the basis of its own data and the data it exchanges with other agents. While negotiating the resources division there is uncertainty about the result of this process. In this case this uncertainty must be included in the decision-making model. The process of modeling this approach is carried out on the example of a pick-up-and-collection task. Moreover, the classical formulation of this task is extended by introducing the uncertainty about the environment, which is related to the limited sensing abilities of agents. In the approach presented, thanks to the possibility of exchanging information, a certainty about the environment can be increased. The system was modeled in MATLAB simulation environment. A few, relevant results were selected in order to illustrate the properties of the approach presented.

2. System description

The approach to negotiating a division of resources is presented on the example of *pick-up-and-collection* task. This task consists of collecting a number of objects scattered inside a workspace of limited area. The task is supposed to be performed in the framework of the distributed system that is described in this section. The system consists of a number of agents. Each robotic-agent has apriori-defined the mission target as well as collaborative strategy embedded. The final behavior of the system emerges from mutual interaction between the agents. Fig. 1 illustrates the principal idea of the system. There is an unknown number of objects to collect scattered over an area being the workspace of the system. The system consists of a specified number of robotic-units also called agents. Each agent can be defined by the following features [32]:

- limited sensing abilities,
- possibility of communication with limited number of other agents,
- a set of actions that an agent is able to take,
- reasoning about those actions based on its own knowledge,
- limited payload.

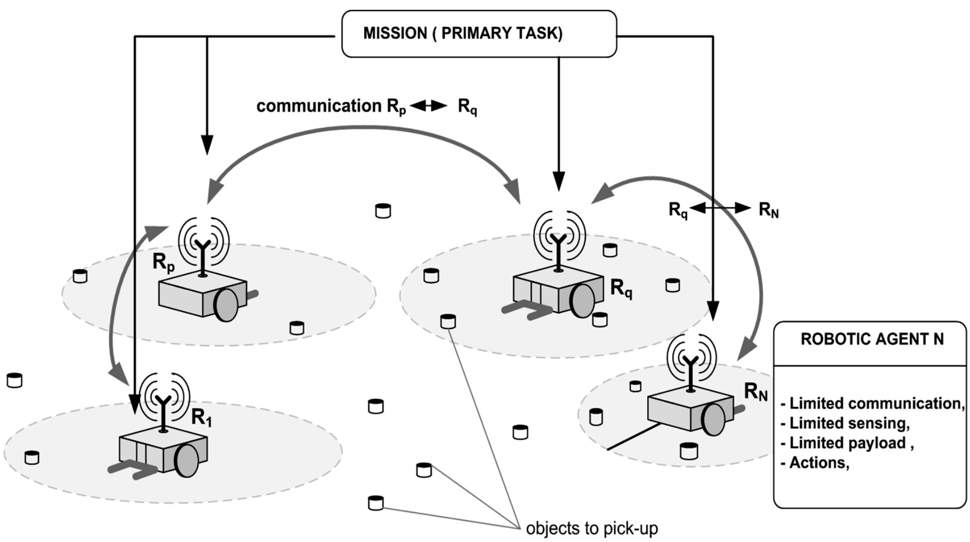

Figure 1. Diagram that illustrates organization of the MAS designed for performing the pick-up-and-collection task.

Since agents can communicate with each other they can exchange information about objects that are detected by their sensing devices and this way increase their knowledge.

The agents that establish communication form the coalition. Inside the coalition it is possible negotiating division of resources which are represented by the objects to collect. Some objects that are in range of one agent can be left for the others in order to improve the effectiveness of the mission performance.

2.1. Agent description

The system consists of N robotic agents that are equipped with devices which allow them to detect and determine location of objects. Let us define the robots' team as the set of indices:

$$R = \{i\}\,,\ i = 1,2,...,N. \tag{1}$$

A state of each robot is determined by $(x_{r,i}, y_{r,i}, \theta_{r,i}, c_{r,i}) \quad i \in R$, where $x_{r,i}$, $y_{r,i}$ denote coordinates of the center of the robot and $\theta_{r,i}$ is a heading defined in the workspace coordinate frame. The fourth element $c_{r,i} \in [0, c_{i,\max}]$ denotes the current load of the ith agent.

Sensing

Each agent is provided with perception devices that allow it to detect objects in its vicinity. Therefore, let us define a set of objects detected by the ith agent as a set of indices:

$$O_i = \{j\},\ i \in R,\ j \in O \wedge O_i \subseteq O. \tag{2}$$

The perception of each agent is limited by the range of its sensors. The range of the sensors is constrained by $s_{r,i}$, which implies that only those objects can be can be detected that satisfies the following:

$$k \in O_i \ \Leftrightarrow\ d^o_{i,k} \leqslant s_{r,i} \tag{3}$$

where $d^o_{i,k}$ denotes the Euclidean distance between the ith agent and the kth object.

Sensing uncertainty

In order to increase the realism of the process modeled it is assumed that the fact of detection of the particular object bears some uncertainty. This uncertainty is partially related to the distance defined by (3). The closer the agent is to the object the greater certainty of detection there is. This uncertainty may be influenced by the accuracy and characteristic of the sensor, the method of detection etc. In this study the model of the uncertainty is not considered. It is assumed that the detection certainty of the jth object by the ith agent is denoted by:

$$p_{i,k} \in [0,1]. \tag{4}$$

Since the same object can be detected by multiple agents, the global certainty about detection event derives from the fusion of information exchanged by interconnecting agents. Let us define a set of certainty factors that determines the events of detection of the kth object by multiple agents:

$$P_k = \{p_{j,k}\},\ j \in R. \tag{5}$$

The global certainty of the kth object detection determined using information (5) collected by individual agents is equal to $\hat{p}_k$. The value depends on a method applied to merging information (5). The method itself in not a subject of this study. For implementing the negotiation schema the fusion method based on reliability criterion, described in [34] was adapted.

Communication

Agents are provided with communication devices that allow them exchanging information. Communication abilities of agents are limited. In the case of this study the limitations are narrowed down to the range of the communication devices. Let us define the set of gents that ith agent is able to communicate with:

$$\Phi_i = \{j\}, \;\; j \in R \wedge j \neq i. \tag{6}$$

Two agents are in communication only when the transmission of data can be made in both directions:

$$j \in \Phi_i \;\; \Leftrightarrow \; d^r_{i,j} < b_i \, \wedge \, d^r_{i,j} < \; b_j \tag{7}$$

where $b_{i/j}$ denotes the range of communication devices of the ith and jth agent, while $d^r_{i,j}$ in (7) is the distance between the ith and jth agent.

Agent's load

Each agent is designed to transport a certain number of objects. The current load of the agent is defined by:

$$c_i \in [0, c_{i,\max}] \tag{8}$$

where $c_{i,\max}$ is the maximum payload of the ith agent which is the number of objects it can accommodate on board.

2.2. The system operation states

The control process of individual agents in a distributed system is decentralized and there is no central coordination. Therefore each agent must be provided with operational management mechanism. In this study it is organized as a finite state automaton (Fig. 2).

After detecting objects the agent tries to establish communication with other agents in order to share information. If it succeeds it sends appropriate data and then it waits for a response. If it is not able to communicate, it builds its own process representation and skip to the state of execution of the task. If the response is received the agent determines these object detected by others agents that it wants to collect and sends this information to those agents. Simultaneously the same can be requested of the agent - to leave some objects for other agents. After completing the negotiations it sends its response that contains information about the negotiations results. Completing the negotiations determine the *sub-task* which is the set of objects the agent decides to collect. Next it skips to the state of execution of the task.

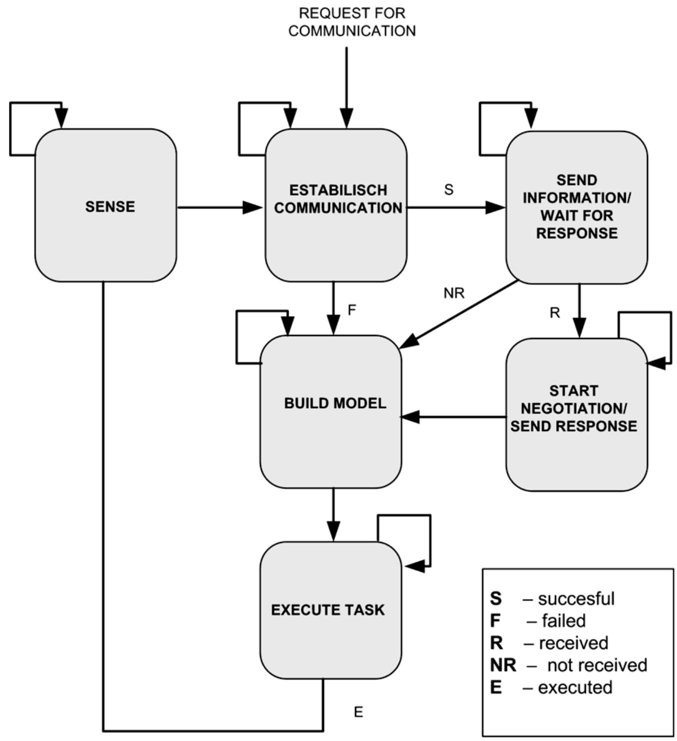

Figure 2. Finite state automaton defining the possible activities of the agent control system.

3. Modeling the collection process

Let us assume that the *i*th agent has detected objects defined by (2) with appropriate certainty factors defined by (4). After establishing communication with nearby agents it receive information (location, certainty factors, current load and the agent' state) about object detected:

$$\hat{O}_i = \{O_k\}, \ \ k \in \Phi_i \ \ , k \neq i. \tag{9}$$

Simultaneously it sends the same data related to its own detection results and its current state to all agents specified in Φ_i. Since the goal of each agent is to collect as many objects as possible but simultaneously it is required to distribute the load among other agents uniformly. Assuming the objects are very heavy, the cost of exploration, picking up and transportation of the objects by the agent that is already carrying a large number of objects is not effective. For this reason the agent estimates if it is able to collect some objects from (10) more effectively. This estimation is made on the basis of agents' loads.

Let us define the value of the predicted load of the ith agent:

$$\hat{c}_i = \begin{cases} c_i + \overline{\overline{O_i}} & \text{for } \hat{c}_i < c_{i,\max} \\ c_{i,\max} & \text{otherwise} \end{cases} \tag{10}$$

where $\overline{\overline{O_i}}$ determines the number of objects detected by the ith agent. The value (10) is just an estimation of the possible loads of the agents intercommunicated with the ith agent. So agents that carry to many objects on board and simultaneously detect relatively large object' number are considered as those that need help from the ith agent. Let us define a set of these agents as:

$$\Phi_{P,i} = \{k\},\ \ k \neq i,\ \ k \in \Phi_i \ \wedge\ \hat{c}_i - \hat{c}_k \geqslant 2. \tag{11}$$

The assumption $\hat{c}_i - \hat{c}_k \geqslant 2$ ensures that after taking one object from the other agent's resources, the loads of these two agents will be equal. It is assumed that the ith agent is allowed to take only one, the nearest object from the agent defined in (11). Hence the set of objects that the ith agent will probably be permitted to take is equal to:

$$\hat{O}_{P,i} = \{j\},\ j \in O_k,\ k \in \Phi_{P,i} \ \wedge\ d^o_{i,j} = \min_{m \in O_k} \left(d^o_{i,m}\right). \tag{12}$$

The message sent to other agents for leaving requested objects contains information defined in (12). On the other hand, at the same time the ith agent can also be requested by other agents to leave some objects from its own resources. Let us similarly denote the set containing those agents that ask the agent ith to leave some objects:

$$\Phi_{R,i} = \{k\},\ k \neq i,\ k \in \Phi_i \ \wedge\ \hat{c}_k - \hat{c}_i \geqslant 2 \tag{13}$$

and the objects that the ith agent is supposed to share with other agents:

$$\hat{O}_{R,i} = \{j\},\ j \in O_i \ \wedge\ d^o_{k,j} = \min_{m \in O_i} \left(d^o_{k,m}\right),\ k \in \Phi_{P,i} \tag{14}$$

The sets defined by (12) and (14) are the subject of negotiations between intercommunicated agents. The information flow in this system is assumed to be defined by peer-to-peer architecture. The agent that sends a request for information or objects will be called *consumer* hereafter. Therefore the ith agent must decide if it will leave some of its objects and sends a response to the *consumers*. The agent asked for sharing its resources will be called *producer* hereafter. The decision in this case must be taken without the knowledge of a decision taken by *producers* that were sent requests by the *consumers*. Such organization of information flow in the system reduces its complexity and makes it easy to implement. But on the other hand it implies the uncertainty that influences the decision making process by the *consumer* and must be resolved by an appropriate negotiation scheme. In the next section the negotiation model will be presented, as well as the arbitration schema that held the aforementioned uncertainty.

4. Resource division

Let us assume that the ith agent determined the negotiation' subject defined by (11)-(14). The arbitration goal is determining the response and send it to agents that requested it. The response contains indices of objects that the ith agent decides to share with *consumers*. Simultaneously, it must take into account uncertainty related to the responses of the *producers* that were requested by itself. After it takes a decision about leaving objects and receives a response from *producers*, the ith agent determines the set of objects it must collect. The process of collecting objects defined by the aforementioned set is the final result of the arbitration process and is a partial task the given agent carries out.

4.1. Case without uncertainty

First, let us consider a relatively simple case without uncertainty, when the set defined by (11) is empty, which means the ith agent does not request any objects that "belong" to other agents. The role of arbitration is to give a response to the *consumer-agents* that sent their requests. This response contains information about objects the ith agent is ready to share with others. In this case arbitration consists of selecting a group of agents that satisfy the following:

- the predicted load of the ith agent after leaving objects is not smaller than the maximum, predicted load among the group of agents,
- the group has to minimize the costs of collection of objects.

Let us first formulate the condition related to the demand for the uniform distribution of agents' loads and denote the subset of the set (13), which determines the group of agents for which the ith agent decide to leave objects:

$$\Phi_{R,i}^{*} \subseteq \Phi_{R,i} \Leftrightarrow \ \hat{c}_i - \overline{\overline{\Phi}}_{R,i}^{*} \geqslant \max_{k \in \Phi_{R,i}^{*}} (\hat{c}_k + 1). \tag{15}$$

If the inequality (15) is satisfied by , this agent whose costs of collecting the object from the set (14) is the highest is removed. The cost of collecting the jth object by the kth agent is defined as follows:

$$e_{k,j} = \delta \frac{1}{\hat{p}_j} d_{k,j}^{o} c_k. \tag{16}$$

The cost is related to the work that the moving agent must do in order to reach the jth object. The cost is then proportional to the way $d_{k,j}^{o}$ must travel, and to its mass. The mass is proportional to the current load $\hat{c}_k$ of the kth agent. The last variable in (16) is the certainty factor $\hat{p}_j$ that reflects conviction of the fact the jth object is on its place. It is assumed that this certainty inversely affects the estimated cost of collection of the object. Therefore the selection of agents defined by (15) has the property that minimizes the cost the group of agents incur for collecting objects:

$$\min \sum_{k \in \Phi_{R,i}^{*}} e_{k,j}, \ j \in \hat{O}_{R,i}. \tag{17}$$

4.2. Case with uncertainty

Let us focus on the case when due to the possibility of collection of some objects by the ith agent, an uncertainty in the negotiation process arises. In a further part of this section the process of modeling the decision-making in a game theory framework will be presented. One of the branches of a general Game Theory are problems in which one player (decision maker) does not collaborate or even more its potential behavior cannot be predicted or it cannot even be classified as rational. This class of problems are referred to as *games against nature*. In this case the *nature* reflects the uncertainty related to the producers' reply to the consumer ask. Let us define the game between the ith agent and *nature*:

$$G = \{A, J\} \tag{18}$$

where A is the game's action space defined as:

$$A = A_i \times A_N \tag{19}$$

where A_i is the set of possible actions that the ith agent can take, and A_N is a set of actions that virtual opponent called *Nature* can take. The element J in (18) is a function that reflects the costs of applying a combination of decisions:

$$J : A \to \Re. \tag{20}$$

This function will be called the *cost function* hereafter.

Action sets

The ith agent that must take the best possible decision and simultaneously take into account the uncertainty of behavior of the *producers*. Let us assume that the decision that the ith agent can take is an answer sent to the *consumers*. The reply sent to the individual agent can be twofold. It can permit the agent that made the request to take one object or refuse to do it. Hence, let us define the action set of the ith agent:

$$A_i = \{a_k\},\ k \in \Phi_{R,i},\ a_k \in \{0,1\}. \tag{21}$$

The number of possible actions of the ith agent is equal to $A_i = 2^{R_{no}}$ where $R_{no} = \overline{\overline{\Phi}}_{R,i}$ is the number of agents that request for resources sharing. On the other hand a space of possible actions of *nature* must be defined. These actions have to reflect the uncertainty of the fact that the ith agent is not sure about the decisions of the *producers*. Thus the action set of a second player is defined as:

$$A_N = \{a_j\},\ \ j \in O_{P,i},\ \ a_j \in \{0,1\}. \tag{22}$$

The number of possible actions of *Nature* is equal to $A_N = 2^{O_{no}}$, where $O_{no} = \overline{\overline{O}}_{P,i}$ is the number of objects the ith agent is requesting. The decision $a_j = 1$ taken by *nature* means that the ith agent is allowed to collect the jth object from the set $O_{P,i}$. Otherwise $(a_j = 0)$ it is not allowed to do it.

Cost function

The cost function is the core of the process model. It maps a particular combination of a player's actions into the cost space. The cost reflects an accuracy of decisions taken with respect to assumed criteria. Let us distinguish factors that affect the costs. The first one is called *reward* and is related to the aim of the ith agent, which is to collect as many objects as possible. This factor stimulates the agent to collect more and more objects. For each object picked up from the workspace the agent obtains an assumed premium which in this case is an abstract value that must be scaled to the general cost amount:

$$q_i = q = \text{const.} \;\; i = 1,2,...N, \; q \in \Re \tag{23}$$

The amount of premium can be varied with respect to a particular agent; nevertheless in this study, it is equal for each agent (agents are homogenous entities). Since the agents are not assumed to be egocentric entities, they tend to improve the efficiency of the mission they carry out in common. It was also said that one of the quality factors that evaluate mission performance is a uniform distribution of loads for agents. In this case the utility of the reward for each agent depends on its current and predicted load. Let us define the linear function that maps the reward into the space of the agent's utility, which depends on its load:

$$f_u(q_i) = \begin{cases} q_i \left(1 - \frac{c_i}{c_{i,\max}}\right) & \text{for} \quad c_i < c_{i,\max} \\ 0 & \text{otherwise} \end{cases} \tag{24}$$

Moreover, the agent should take into account the possible loads of other agents. It is clear that if one agent reaches its maximum payload and simultaneously detects a large number of objects, its aim should not be to collect all those objects. It perceives different utility (real value) of the reward than the agent whose load is low. The ith agent must also evaluate if leaving objects for other agents is profitable with respect to the energy spent for the collection of objects that were left. Let us formulate the cost related to a given combination of decisions as a function that consists of two components

$$J(a_i, a_N) = Q_i + E. \tag{25}$$

The first one is related to the real value of the reward the ith agent obtains after decision a_i and in the case when the decisions of the *producers* are defined by a_N. As was discussed before, the real utility of the reward is a function of the load of the ith agent as well as the predicted loads of other agents after taking decisions $(a_i,\, a_N)$. So the first component of (25) is defined as:

$$Q_i = f_u\left(q_i(a_i, a_N, q)\right) \tag{26}$$

where $q_i(a_i, a_N, q)$ is the reward value that reflects mutual dependencies between agents' loads. Hence it is defined as a function that consists of three components:

$$q_i(a_i, a_N, q) = q(-\hat{c}_i^A + s + g) \tag{27}$$

where $\hat{c}_i^A$ denotes the predicted load of the ith agent after taking its decision a_i in the case that the response of the *producers* is defined by the set a_N:

$$\hat{c}_i^A(a_i, a_N) = \hat{c}_i - \sum_{k \in \Phi_{R,i}} a_i(k) - \sum_{j \in \Phi_{P,i}} a_i(j). \tag{28}$$

The second component in (27) is related to the distribution of payloads between the ith agent and the *consumers* that request objects. It is an implementation of a simple strategy. If in the worst case, the predicted load of the ith agent is greater than or equal to the summary of the predicted loads of the *consumers* that are considered for permission to take objects, then the reward is decreased. The decrease is proportional to the aforementioned difference. Also, if this difference is smaller than zero the reward has to be decreased, in which this case the decrease in the reward is even greater. The strategy is formulated using the following:

$$s = \begin{cases} \underline{\hat{c}}_i - \sum_{k \in \Phi_{R,i}} a_i \left(\overline{\overline{O}}_k + 1 \right) & \text{for} \quad \text{s} > 0 \\ s^2 & \text{otherwise} \end{cases} \tag{29}$$

where $\underline{\hat{c}}_i$ is the worst-case load of the ith agent given as:

$$\underline{\hat{c}}_i = \overline{\overline{O}}_i - \overline{\overline{O}}_{R,i}. \tag{30}$$

The third component in (27) takes into account the possible responses of *producers*. It also maps another intuitive social rule into the space of "rewards". Therefore behavior that consists of the ith agent is taking additional objects from the *producers'* resources. In the case when the ith agent decides not to share its object with the *consumers*, it should be "punished". Hence the value of third component in (27) is defined by the following formula:

$$g = \begin{cases} -\sum_{j \in \Phi_{P,i}} a_N(j) & \text{for s} \leqslant 0 \\ -g & \text{otherwise.} \end{cases} \tag{31}$$

By defining these three components, the first factor of the cost function (25) was designed. It is related to virtual rewards that are awarded for a uniform distribution of loads and the "care" of the given agent for effective and collaborative behavior. The second component of (25) is related to the real, energetic cost of the execution of a partial task, related with hypothetical decisions of ith agent. This factor takes into account total, estimated costs that incurs the group of agents performing the scenario that results from the decisions of the ith agent. Let us then define the energetic factor of the cost function (25) as:

$$E(a_i, a_N) = \sum_{k \in O_{R,i}, j \in \Phi_{R,i}} a_i(k) e_{k,j} + \sum_{l \in O_{P,i}} a_N(l) e_{k,i}. \tag{32}$$

The first component of (32) is a summary cost of the *consumers* collecting those objects the ith agent permitted them to collect. On the other hand, the second component is the cost of collecting those objects the ith agent is permitted to take from the set of the *producers*.

4.3. Solution

The result of the game is defined by the set a_i^0 of the decision that the ith agent took with respect to each agent defined by (13).The problem solving method should take into account the response uncertainty received from the *producers*. This uncertainty is just modeled as different scenarios represented by "actions" taken by *Nature*. Considering the part of Game Theory related to modeling conflicts with irrational opponents – like *Nature*, a number of solutions of such conflicts have been studied and proposed [15]. In order to present a functioning of the proposed methodology it was applied the most common, Wald's criterion, which corresponds to pessimistic behavior of the player. According to this criterion, the player is prepared for the worst situation. The behavior of the player applying Wald's criterion consists of minimizing the possible costs of its decision. Therefore the ith agent takes the decision:

$$a_i^0 = \min_{a_i \in A_i} \max_{a_N \in A_N} \left(J\left(a_i, a_N\right) \right). \tag{33}$$

Using strategy (33) the ith agent secures its possible costs at a level called the *security level*. This solution does not give the best possible results but in the case of high uncertainty allows critical results of the decision to be avoided. After carrying out the arbitration, the given agent plans the execution of the partial task. The partial task consists of collecting all the objects that belong to the set defined after the negotiation process. This set can be defined as:

$$O_i^* = O_i \cup O_{i,R}(a_i^0) / O_{i,P}^* \tag{34}$$

where $O_{i,R}(a_i^0) \subseteq O_{i,R}$ denotes the set of objects that the ith agent decided to leave for other *consumers*. On the other hand, the set $O_{i,P}^*$ is the result of negotiations carried out by the producers and seen from the perspective of the ith agent. This set contains information about objects the *producers* decided to leave for the ith agent. Thus the set (32) contains the objects the ith agent is to collect. The task of the agent's local planner is to find optimal solution of the partial task. The partial task is a sub-problem of the primary one which is to find the path that allows the collection of all objects by spending minimal amount of energy. In this study a graph representation was used to model spatial relations between objects and estimate collecting costs using Hamiltonian Path.

5. Resource division summary

Let us consider the following scenario (Fig. 3): one agent (which will be referred to as the *i*th hereafter) detects a number of objects defined by the set O_i. After that, it looks for other agents in order to check if it is possible to improve the objects collecting process by sharing or exchanging objects with other agents.

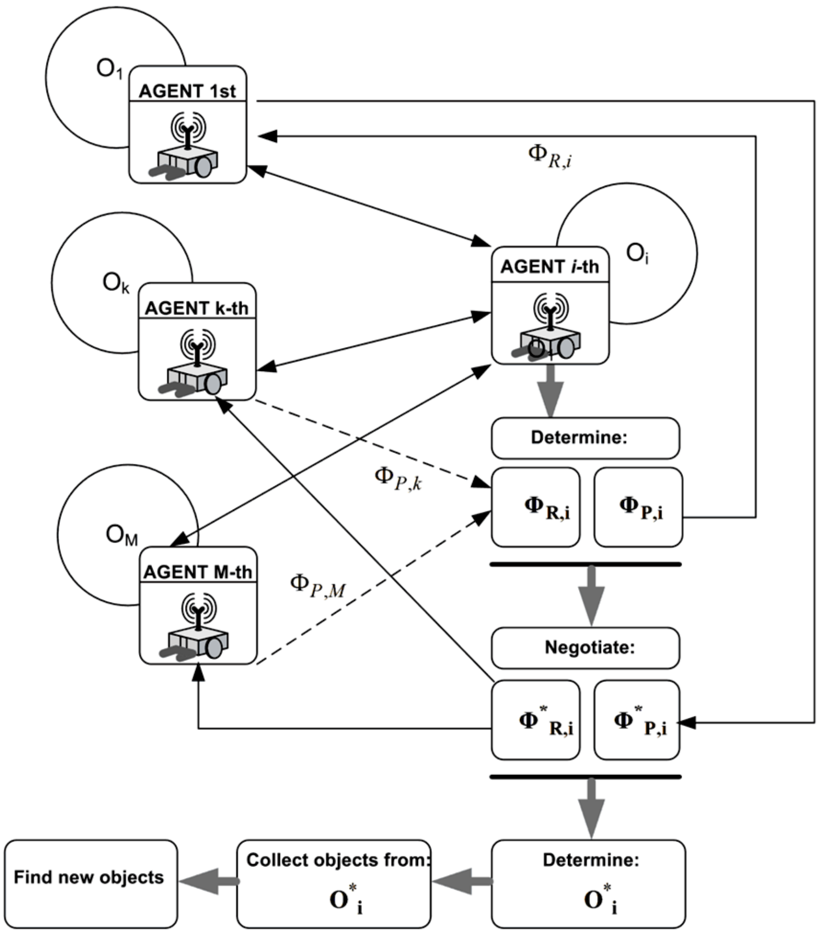

Figure 3. Diagram of information flow while executing the single stage of the mission.

Let us suppose that it established a communication link with the following agents: agent 1, agent k and agent M. It sends them information about detected objects, its load and location. On the other hand it receives the status data from other interconnected agents. Using received data, its knowledge of the state of the process has been enhanced. It tries to define these agents that detected too many objects or are overloaded. These agents are requested to leave some objects for the *i*th agent. They are defined by the set $\Phi_{P,i}$. This kind of reasoning is followed by other agents which send requests to the ith agent. These requests are defined by the set $\Phi_{R,i}$. Now the agent must decide which objects it is ready to share in order to improve the distribution of the load among an interconnected group of agents. This decision must be taken under uncertainty about responses from agents requested by the ith agent. The decision is taken as a result of an arbitration process that was carried out and is defined by the set $\Phi^*_{R,i}$. Information about its decision is sent to agents specified by the set $\Phi_{R,i}$. Simultaneously this kind of reasoning is followed by other interconnected agents. The result of this process is sent to the ith agent and defines its set $\Phi^*_{P,i}$. After closing negotiations the agent determines the set of objects O_i^* that it finally decided to collect. The next step of the agent's operation is to plan a path a (sequence of "move and pick-up" actions) . This plan is an optimal solution for the local, partial task. After completing the task, it starts to look for other objects.

6. Simulation results

6.1. Single stage case study

In order to explain and clarify the proposed idea of distributing tasks among agents (which is somewhat complicated), in the beginning let us consider just single stage of the process. It should explain the information flow between agents and the idea of negotiations. The exemplary scenario is presented in Fig. 4.

There are 20 objects scattered within the workspace and 4 robotic-agents indexed from 1 to 4, with their initial configurations as shown in Fig. 4. The payloads of the agents are equal to 5. The initial loads c_i of the agents are correspondingly equal to $c_i \in \{0,2,1,1\}$, $i = 1,2,\ldots,4$. The range of sensors of particular agents is defined by the set $s_{r,i} \in \{30,50,40,30\}$, $i = 1,2,\ldots,4$. For each agent its communication range is limited to $b_i = 80$ [m], $i = 1,2,\ldots,4$. Thus particular agents can detect a number of objects depending on the range of their sensors as well as on their current locations. The sets of detected objects are defined by:

$$
\begin{aligned}
O_1 &= \{17\} \\
O_2 &= \{1,3,6,18,19\} \\
O_3 &= \{2,10,16\} \\
O_4 &= \{5,7,9\}
\end{aligned}
$$

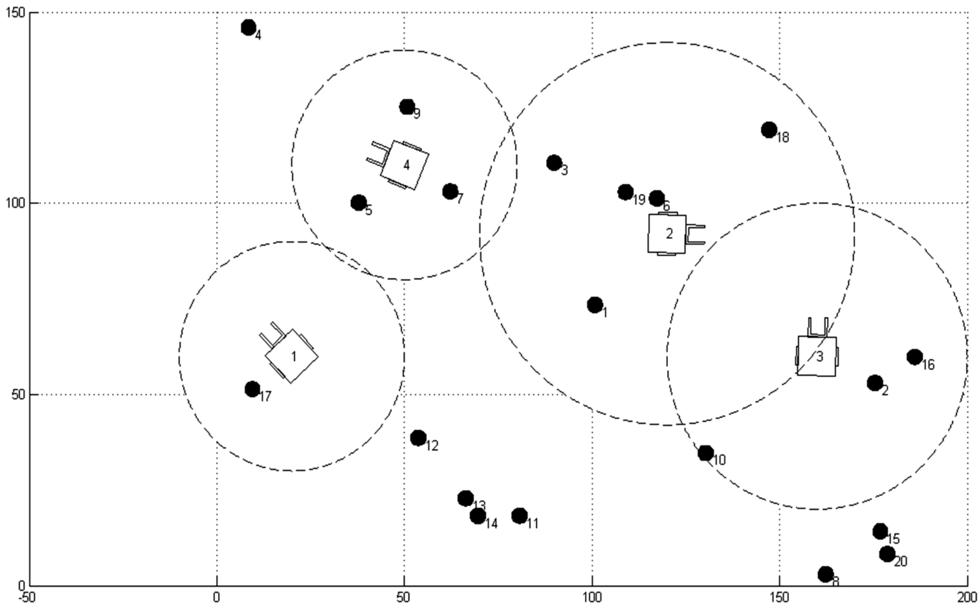

Figure 4. Exemplary scenario for a single stage case study.

The next stage is to establish communication with other agents. The sets that define the agents' indices the given agent is able to communicate with are as follows:

$$\begin{aligned}\Phi_1 &= \{4\}\\ \Phi_2 &= \{3,4\}\\ \Phi_3 &= \{2\}\\ \Phi_4 &= \{1,2\}\end{aligned}$$

After this, each agent starts negotiations with others. The input of the negotiation process are sets defining objects that each agent can leave or take from others. In the case studied here they are equal to:

$$\begin{array}{llll}\Phi_{R,1} = \varnothing, & \Phi_{P,1} = \{4\}, & \hat{O}_{R,1} = \varnothing, & \hat{O}_{P,1} = \{5\}\\ \Phi_{R,2} = \{3,4\}, & \Phi_{P,2} = \varnothing, & \hat{O}_{R,2} = \{6,3\}, & \hat{O}_{P,2} = \varnothing\\ \Phi_{R,3} = \varnothing, & \Phi_{P,3} = \{2\}, & \hat{O}_{R,3} = \varnothing, & \hat{O}_{P,3} = \{6\}\\ \Phi_{R,4} = \{1\}, & \Phi_{P,4} = \{2\}, & \hat{O}_{R,4} = \{5\}, & \hat{O}_{P,4} = \{3\}\end{array}$$

The results of negotiations are defined by sets that contain the indices of objects the given agent is ready to share:

$$O^*{}_{R,1} = \emptyset$$
$$O^*{}_{R,2} = \{6, 3\}$$
$$O^*{}_{R,3} = \emptyset$$
$$O^*{}_{R,4} = \{5\}$$

Thus the result of negotiations is that agent 1 gains additional objects from agent 4. Simultaneously, agent 4 gains object 3 from agent 2 and agent 3 gains object 6 from agent 2. The final sets particular agents negotiated to collect are defined as follows:

$$O^*{}_1 = \{5, 17\}$$
$$O^*{}_2 = \{1, 18, 19\}$$
$$O^*{}_3 = \{2, 6, 10, 16\}$$
$$O^*{}_4 = \{3, 7, 9\}$$

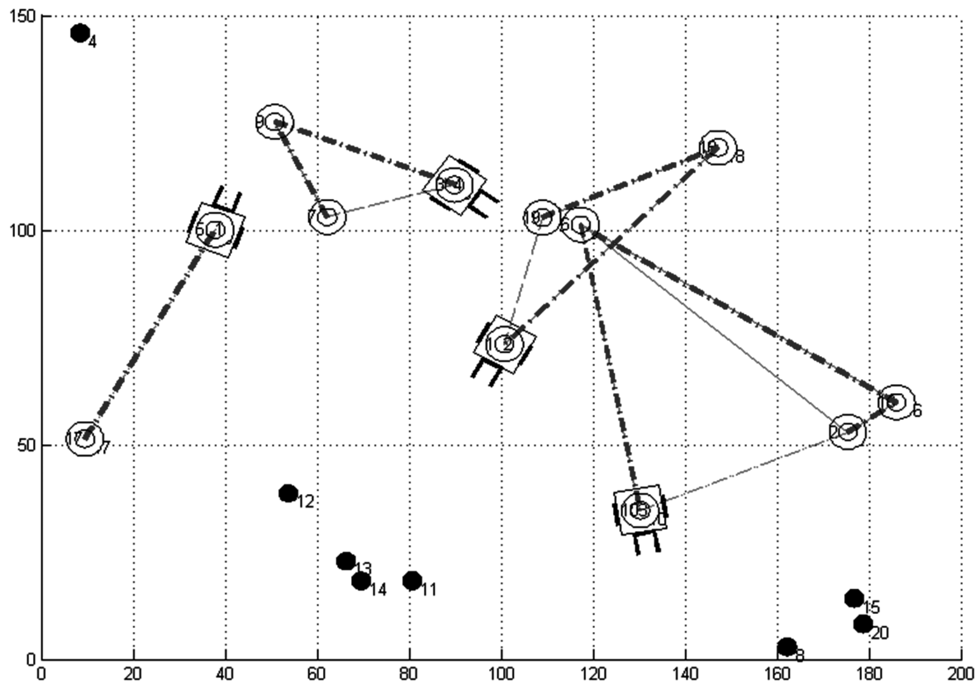

Figure 5. Graphical interpretation of the solution for the single stage of the collection process.

Finally, optimal solutions that are time-ordered sequences of objects the given agent planned to collect are:

$$S^*{}_1 = \{17, 5\}$$
$$S^*{}_2 = \{19, 18, 1\}$$
$$S^*{}_3 = \{2, 16, 6, 10\}$$
$$S^*{}_4 = \{7, 9, 3\}$$

A graphical interpretation of the transitions of agents resulting from this solution is presented in Fig. 5. When a particular agent completes the task it tries to detect new objects from the location it reached after completing its partial task. If it is not able to do so, it moves in a random direction and tries to detect objects again. In both cases, whether it did or did not detect new objects, it tries to establish a connection with other agents and the overall process described is repeated again. The results of continuation of this experiment are presented in Fig. 6. The paths of agents while collecting objects are marked with different line styles. It can be seen that thanks to collaboration between agents the task is completed more efficiently than in the case of using self-oriented agents. It is worth stressing that the task was completed with a uniform distribution of loads between agents.

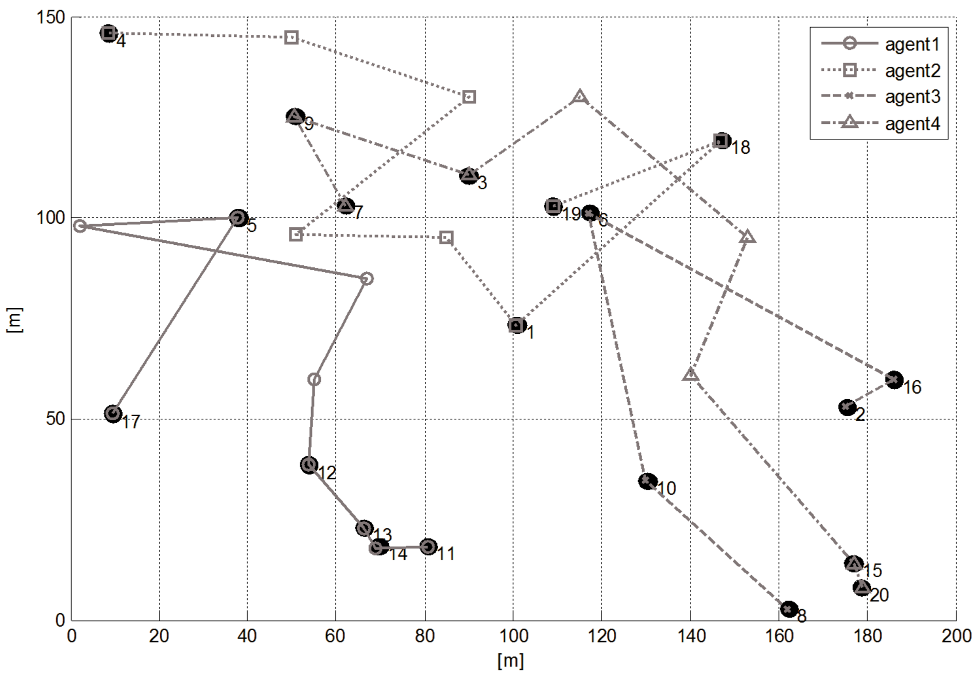

Figure 6. Collection of 20 objects by 4 agents – illustration of paths of individual agents.

6.2. Further experiments

In order to present the functioning of the method proposed two experiments were selected. The experiments consists of collecting 30 and 40 objects distributed throughout the workspace, by four agents. The agents start work in different initial configurations which are shown in Figures 7 and 8. In the first experiment (Fig. 7), the agents are placed close to each other in the given region of the workspace. It can be noticed that the quality of the task execution depends strongly on the distribution of the objects and the initial positions of agents. In such cases the key issue in effective task execution is communication and collaboration between agents. For example, in the experiment mentioned agent 1 after collecting object 3 went into the dead-zone where there were no objects. Thanks to communication with agent 2 and its "will" to collaborate it was informed and permitted to pick up the object 17. Finally the task was executed effectively as can be seen in Fig. 7. Agents terminated the work with equal loads corresponding to $\{7\ 8\ 8\ 7\}$, which shows a uniform distribution of loads and a correct task division while carrying out the mission.

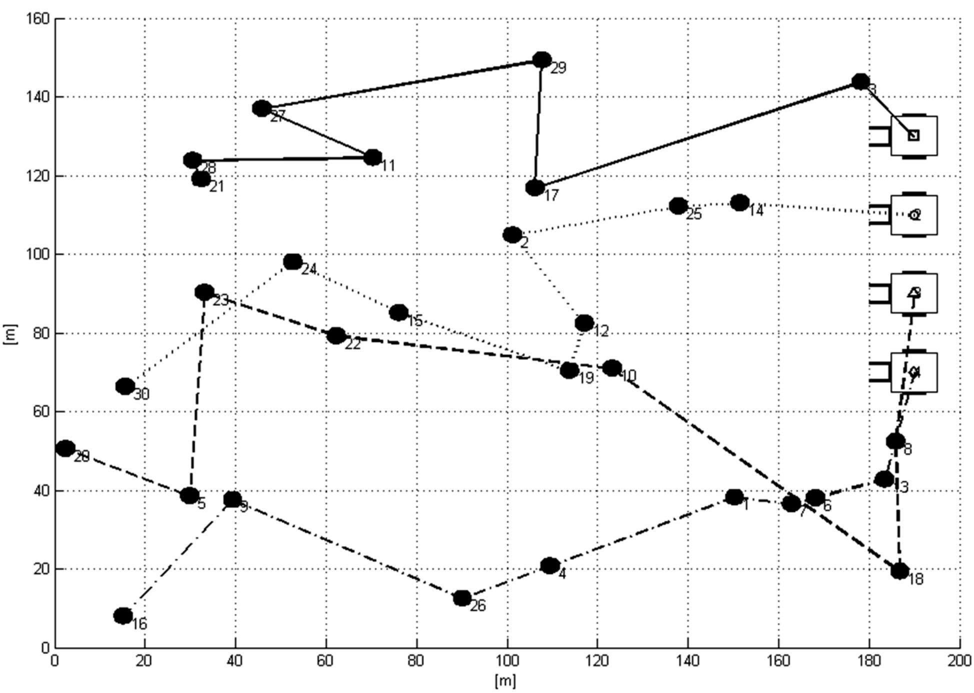

Figure 7. Experiment in collection of 30 objects by 4 agents.

The second experiment shows the work of the method with different initial configurations of agents and a larger number of objects to collect. Once again it can be observed (Fig. 8) that the task was executed in an effective way. Communication and the collaborative behavior by each agent brought improvements to their work when they entered in dead-zones without any objects in their range. Also the distribution of loads $\{10\ 10\ 10\ 10\}$ proves efficiency of the method, of course when the overriding criterion of an efficiency is a uniform distribution of loads of individual agents.

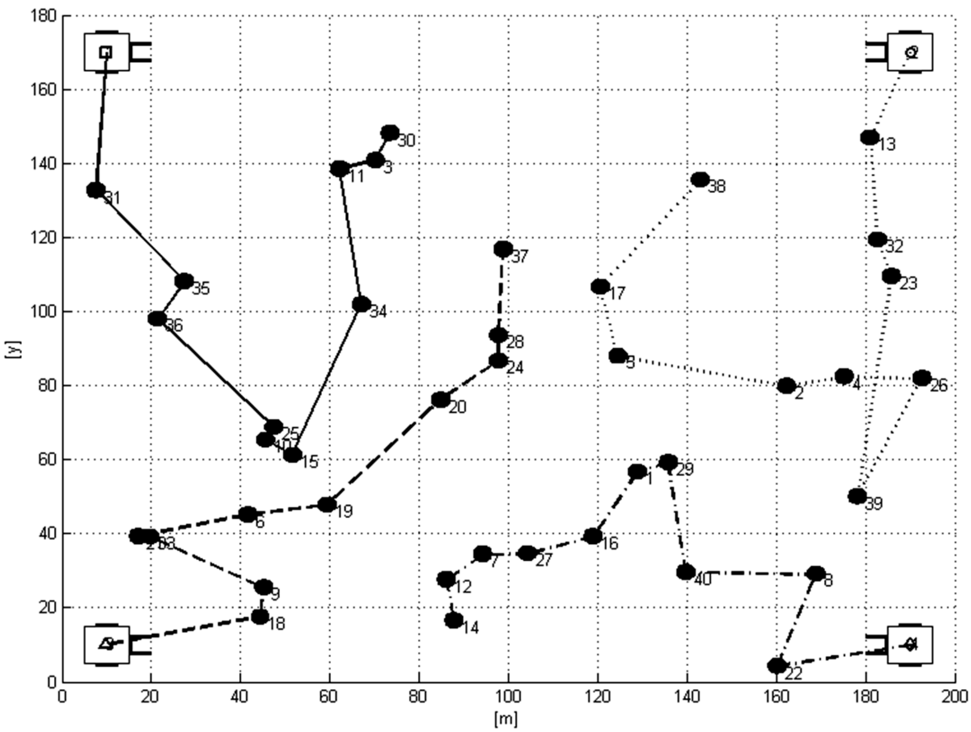

Figure 8. Experiment in the collection of 40 objects by 4 agents.

7. Conclusion

This paper addressed the task division and action coordination problem of robotic agents in a framework of distributed, Multi-Robot System. The agents' team was intended to perform the pick-up and collection task. A limitation of the sensing abilities as well as the communication range of each agent was assumed. That implied that knowledge of the environment was uncertain, incomplete and distributed among the individual agents. In this paper a framework for cooperative task assignment to individual agents is proposed. The process of resource division was based on the arbitration that utilize the basics of Game Theory. The process of negotiating access to common resources by intercommunicating agents was modeled and solved as a game against Nature that in this case symbolizes the uncertainty. The system was assumed to be an asynchronous one. Therefore each agent was provided with an embedded state machine based operational manager, which defined the communicational peer-to-peer protocol. The work of the system was verified by numerous simulations that were carried out. The results of simulation allowed evaluation of quality of the method. The results showed that the method presented enables proper task execution. Due to the exchange of information between agents it is possible to obtain a team behavior that tends to equalization of loads between particular agents. Moreover the cost involved to the team for the collection of objects is minimized by the collective behavior of the agents. The results are not guaranteed to be optimal but seem to be correct and feasible. The effectiveness of the task division depends on the communication range and the distribution of the objects as well as the algorithm of object search. In this paper a probabilistic approach to objects search was applied and therefore the performance of the method is not deterministic. Future research related to this subject will be focused on examining different search strategies as well as modifying the negotiation scheme in the proposed fixed framework.

References

[1] C. BALDASSANO and N.E. LEONARD: Explore vs. exploit: Task allocation for multi-robot foraging. Preprint. 2009. Available online http://www.princeton.edu/naomi/publications.html.

[2] R.P. BONASSO, D. KORTENKAMO, D.P. MILLER and M. SLACK: Experiences with an architecture for intelligent, reactive agents. In M. WOOLDRIDGE, J.P. M ULLER and M. TAMBE eds. Intelligent Agents II. Lecture Notes in Articial Intellignce **1037** New York, Springer-Verlag, 1996, 187-202.

[3] R.A. BROOKS: Intelligence without representation. *Artificial Intelligence*, **47**(1-3), (1991), 139-159.

[4] W. BURGARD, M. MOORS, D. FOX, R. SIMMONS and S. THRUN: Collaborative multi-robot exploration. *Proc. IEEE Int. Conf. on Robotics and Automation*, San Francisco, CA, (2000), 476-481.

[5] C. CANDEA, H. HU, L. IOCCHI, D. NARDI and M. PIAGGIO: Coordination in Multi-Agent RoboCup Teams. *Robotics and Autonomous Systems*, **36** (2001), 67-86.

[6] X. CHENG, J. SHEN, H. LIU and G. GU: Multi-robot cooperation based on hierarchical reinforcement learning. *Lecture Notes in Computer Science*, **4489** 2007, 90-97.

[7] R. FIERRO, A. DAS, J. SPLETZER, J. ESPOSITO, V. KUMAR, J.P. OSTROWSKI, G. PAPPAS, C.J. TAYLOR, Y. HUR, R. ALUR, I. LEE, G. GRUDIC and J.

SOUTHALL: A framework and architecture for multi-robot coordination. *International J. of Robotics Research*, **21**(10-11), (2002), 977-995.

[8] B.P. GERKEY and M.J. MATARIC: Sold!: Auction methods for multi-robot coordination. *IEEE Trans. on Robotics and Automation*, **18**(5), (2002), 758-768.

[9] R. GHRIST and S.M. LAVALLE: Nonpositive curvature and pareto-optimal coordination of robots. *SIAM J. of Control and Optimization*, **45**(5), (2006), 1697-1713.

[10] I. HARMATI and K. SKRZYPCZYK: Robot team coordination for target tracking using fuzzy logic controller in game theoretic framework. *J. of Robotics and Autonomous Systems*, **57**(1), (2009), 75-86.

[11] J. JONES and M. MATARIC: Adaptive division of labor in large-scale minimalist multi-robot systems. *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Las Vegas, (2003), 1969-1974.

[12] N. KALRA and A. MARTINOLI: Comparative study of market-based and threshold-based task allocation. *Distributed Autonomous Robotic Systems*, **7** (2006), 91-101.

[13] G.A. KAMINKA, D. ERUSALIMCHIK and S. KRAUS: Adaptive multi-robot coordination: A Game-Theoretic perspective. *Proc. of IEEE Int. Conf. on Robotics and Automation*, Anchorage, Alaska, USA, (2010), 328-334.

[14] M. KLUSCH and A. GERBER: Dynamic coalition formation among rational agents. *IEEE Intelligent Systems*, **17**(3), (2002), 42-47.

[15] J.K. KOK and M.T.J. SPAAN and N. VLASSIS: Non-communicative multi-robot coordination in dynamic environments. *Robotics and Autonomous Systems*, **50**(2-3), (2005), 99-114.

[16] K. KOSUGE, Y. HIRATA, H. ASAMA, H. KAETSU, and K. KAWABATA: Motion control of multiple autonomous mobile robots handling a large object in coordination. *Proc. IEEE Int. Conf. on Robotics and Automation*, Detroit, MI, (2000), 2666-2673.

[17] S.M. LAVALLE and S.A. HUTCHINSON: Game theory as a unifying structure for a variety of robot tasks. *Proc. IEEE Int. Symp. on Intelligent Control*, (1993), 429-434.

[18] Y. LI and X. CHEN: Modeling and simulation of swarms for collecting objects. *Robotica*, **00** (2005), 1-10.

[19] J. LAWTON, B. YOUNG and R. BEARD: A decentralized approach to elementary formation maneuvers. *Proc. IEEE Int. Conf. on Robotics and Automation*, San Francisco, CA, (2000), 2728-2733.

[20] Y. MENG: Multi-robot searching using Game-Theory based approach. *Int. J. of Advanced Robotic Systems*, **5**(4), (2008), 341-350.

[21] D.P. STRAFFIN: Game Theory and Strategy. The Mathematical Association of America. 1993.

[22] I. REKLEITISY, V. LEE-SHUEY, A.P. NEW and H. CHOSET: Limited communication, Multi-Robot Team based coverage. *Proc. of Int. Conf. on Robotics and Automation*, New Orleans, LA, (2004), 3463-3468.

[23] Y. SHOHAM and K.L. BROWN: Multiagent systems: Algorithmic, Game-Theoretic, and logical foundations. Cambridge University Press, NY, USA, 2009. ISBN: 978-0-521-89943-7.

[24] K. SKRZYPCZYK: On multi agent coordination in the presence of incomplete information. *Proc. Int. Conf. on Artificial Intelligence and Soft Computing*, Zakopane, Poland, (2008), 1254-1256.

[25] M. SONG, G. GU, R. ZHANG and X. WANG: A method of multi-robotformation with the least total cost. *Int. J. of Informationand System Science*, **1**(3-4), (2005), 364-371.

[26] P. STONE and M. VELOSO: Multiagent systems: A survey from the machine learning perspective. Technical report, CMU-CS-97-193, School of Computer Science, Carnegie Mellon University, 1997.

[27] T. SUGAR and V. KUMAR: Control and coordination ofmultiple mobile robots in manipulation and material handling tasks. In P. CORKE and J. TREVELYAN eds. Experimental Robotics VI: Lecture Notes in Control and Information Sciences, **250** 15-24, Berlin: Springer-Verlag, 2000.

[28] K. SYCARA: 1998: Multiagent systems. *Artificial Intelligence Magazine*, **19**(2), (1998), 79-92.

[29] R.J. WILSON: Introduction to Graph Theory. 3rd ed., New York: Longman Scientific & Technical, 1987. ISBN 0-582-44685-6.

[30] K.J. WINKFELD and G. ZLOTKIN: Multiagent negotiation under time constraints. *Artificial Intelligence*, **75** (1995), 297-345.

[31] M. WOOLDRIDGE and N. JENNINGS: Intelligent agents: Theory and practice. *Knowledge Engineering Review*, **10**(2), (1995), 115-152.

[32] M. WOOLDRIDGE: An Introduction to Multiagent Systems. Johnn Wiley & Sons Ltd, UK, 2009, ISBN:978-0-470-51946-2.

[33] D. VAIL and M. VELOSO: Dynamic multi-robot coordination. In A. SCHULTZ, L. PARKER and F. SCHNEIDER, eds. Multi-Robot Systems: From Swarms to Intelligent Automata, **2** Kluwer Academic Publishers, the Netherlands, 2003, 87-98.

[34] M. MELLADO and K. SKRZYPCZYK: Information fusion in Multi-Agent System based on reliability criterion. In A. NAWRAT and Z. KUS eds. Vision Based Systems for UAV Applications. Series: Studies in Computational Intelligence, Springer, **481** 2013, 207-217, ISBN: 978-3-319-00368-9.