A Multi-Robot Task Allocation Method to Regulate Working Groups Sizes

José Guerrero, Gabriel Oliver *

Dept. de Matemátiques i Informática, Universitat de les Illes Balears (UIB),
Cra. de Valldemossa, km 7.5, 07071
Palma de Mallorca
SPAIN

Abstract. Task allocation is a complex and open problem for multiple robots systems. In this paper, we present a solution for dynamically allocating tasks in a colony of mobile robots. The system is partially inspired on auction methods and tries to determine the optimum number of robots that are needed to solve specific tasks, or sub-goals, in order to minimize the execution time of a mission. The method has been extensively tested for a modification of the well-known foraging task. Initial situations with different combinations of workload and work capacity have been used to test the proposed solution on robot teams with different homogeneities. Experimental results are presented to show the benefits of the proposed method.

1 Introduction

Multi-robot systems can provide several advantages over a single-robot system: robustness, flexibility, efficiency, etc. To take advantage of these potential aspects the robots must cooperate to carry out their common mission. It is well known that several problems have to be solved to get this objective. Some tasks can be carried out by a single robot but if two or more robots can cooperate, the task can be archive faster. In this situation, the following question has to be answered: How many robots and which of them do we need to optimize the execution time of this task? As it has been demonstrated in different studies, if the number of robots is increased with wrong criteria, the execution time can increase, among other causes, due to the interference effect. The interference is the result of competition for the shared resources, especially the physical space and the robots themselves. In [9] it is shown how the interference can affect the execution time of the foraging tasks and how increasing the number of robots, the time to finish the mission also increases. Besides, if a certain task captures the attention of an excessive number of robots, other tasks can be forsaken.Therefore, our challenge is to design a distributed system to coordinate a heterogeneous robot team working in an unknown and dynamic environment. Among these problems we mainly focus on the task allocation, that is, selecting the best robot or robots to perform a specific task.

In this paper, we propose a decentralized method of task allocation for groups of heterogeneous robots executing foraging-like tasks. Our method is partially inspired in both the auction-based works [6], [5] and the threshold models of the ant-inspired approach [3]. Special attention will be focused on deciding the optimal number of robots to execute a specific task. This problem is closely connected with the diversity level of the team of robots, as will be shown later. To measure the heterogeneity of the robot collectivity we use, among others, the social entropy proposed by T. Balch [2]. We will also study the relation between this metric and some of our architecture parameters.

In the system here presented, when a robot detects a target it tries to organize a *working group*, this group consists of all the robots that will cooperate to execute the specific task. One of the robots of this group will be the leader of the group. This leader will decide how many robots and which of them will be needed to execute the task. To select these robots the leader starts an auction process, similar to [6]. The auction mechanism does not allow deciding how many robots will form the group. Thus, to take this decision, the leader together with the auction mechanism has to calculate the ratio between the work to do and the work capacity of the robots members of the group. One of the objectives of the leader is to create a group

* Email:{jose.guerrero,goliver}@uib.es

as reduced as possible with the maximum work capacity. It is important to mention that the number of robots of the group can vary during the task execution. As it will be shown, during the mission, the leader can accept more robots to be members of the group as well as other robots can abandon it. The experiences shown here are very similar to the so-called foraging tasks. In the present case, each robot has a limited load capacity (work capacity). If a robot can't load the full object, it loads a portion of the weight, goes to the delivery point and comes back for more.

The rest of this paper is organized as follows: section 2 presents some relevant work in the field of multi-robot task allocation; section 3 describes our method and it implementation; section 4 shows the experiments carried out to validate our approach; section 5 explains some future works; conclusions and future work appear in section 6.

2 Related Work

The computer engineering community has done a lot of work to solve the task allocation problem. In recent years, some studies on multi-robot systems have used some similar ideas to solve the problem of how team robots' can distribute their individual work capacity to efficiently achieve a common task. This section shortly relates some of those works that have inspired us. Dias and Stenz [5] have proposed cooperation mechanisms based on explicit coordination between robots in the so-called market-based mechanisms. In the same line, Gerkey and Matarić are working on auction-based mechanisms [6]. In this kind of systems the robots act as self-interest agents and they bid for the tasks. The robot with the highest bit wins the auction process and gets the task. The bids are adjusted to the robots' interest (capacity) to carry out the goal. Thus, when a robot finds a task, it can start an auction process and subcontract its execution to another robot. Other works have proposed biology-inspired solutions, especially from insect colonies like ants and wasps. To implement these systems some authors make use of the *response thresholds* [3], [8]. In these systems, each robot has a stimuli associated with each task to execute which value depends of the kind of robot, the task performance, etc. When the level of the task-associated stimuli exceeds a threshold, the robot starts its execution. In [4] a comparison between biology-based and marked-based task allocation algorithms is presented. The experiments shown in that paper are performed with industrial manipulators and they conclude that the performance of both, biology-based and marked-based systems is very similar.

Our approach is partially inspired in the auction mechanisms and, consequently, the best robot for a specific task can be chosen. However, while previous works can not determine the optimal number of robots to execute a task, our method allows deciding this number as a function of both, the amount of work required to complete the task and the work capacity of the robots involved in the auction process.

3 Working group leading, formation and updating

The coordination mechanism description, including the groups' formation, the leading, the membership policy and the task assignment is described in the following paragraphs.

In an initial stage, each robot is looking for a task. When a robot finds by itself a new task, it will try to leader it. As for each task it can only be one leader, the candidate will first check if this task is assigned to any other robot or not. If there are two or more robots requesting the leadership of the same task, it will be assigned to the better robot. To determine their suitability, we evaluate each candidate using a measure that depends on the specific task to do. During our experiments this measure is the robots' load capacity.

When a robot is promoted to leader of a task, it evaluates the work needed to carry it out. Then, it will create, if necessary, the work group; that is, the set of robots that will cooperate to execute this specific task. In that case, the leader must decide which is the optimum group size. In this work, we propose the leader to decide, as it will be exposed in the auction mechanism, using the following equation:

$$\frac{taskWorkLoad}{\sum_{i=1}^{N} workCapacity_i} < TH \tag{1}$$

Where *taskWorkLoad* is the amount of work required to finish the assigned task that is calculated by the leader. N is the number of robots of the group and $workCapacity_i$ is the individual work capacity of the ith robot. Finally, TH is the *group threshold*; this value is a parameter that will be used to compare the efficiency of the group formation policy. Using equation (1) the leader fixes the maximum ratio between work to do and available work capacity and, therefore, it fixes the maximum number of robots that will be part of the group. This concept is similar to the response threshold of the ant-like algorithms, where the stimulus for each robot is a combination of its load capacity and the taskWorkload. In the present case, the decision process is also centralized on the leader.

To select the robots that will be part of its group the leader uses an auction process. Unlike other auction-based methods, we use the inequality (1) to select the robots. This process is as follows:

– *Auction beginning*: the leader sends a massage to inform that it has found a new task and needs to form a working group.
– *Bids*: each robot without any assigned task (robots that are not part of any group) sends to the leader its work capacity.
– *Auction finishing*: after a fixed time has passed, the leader closes the auction and doesn't accept more bids.
– *Selecting the robots*: the leader selects the robots with the highest work capacity provided the inequality (1) is verified.
– *Agreement and refuse negotiation*: The leader sends an agreement massage to the selected robots and, if the offer is accepted, they start to work for the group. The robots that do not accept to be part of the group reply with a refuse message and the leader subtracts their work capacity from the group capacity. A robot can refuse an offer if it joined another group or it became a leader task during the auction process.

If after the process the equation (1) is not fulfilled, the leader starts a new auction round. When the task is finished, the working group is dissolved immediately.

The size of the group can vary during the execution of the task by means of two mechanisms. The group size can be reduced by a *robot segregation process*. If a non-leader robot, which is part of a group, finds a new task, it will try to leave its present group and become a leader of the new task. When this situation occurs, the segregated robot communicates this circumstance to its old leader which will recalculate equation (1) for the new situation of its group and, eventually, can initiate a new auction process. On the other hand, the size of the group can increase thanks to a *robot aggregation process*. This process acts when after a certain time has passed and a robot does not have any task to execute. Then, it sends a message asking for a task. If a leader receives one of these messages, it accepts the robot in its group if the group size is less than a fixed number. Using this mechanism, the inactive robots are avoided.

4 Experiments and Validation

This section explains the experiments carried out to validate our approach and how the value of the group threshold and the social entropy of the group affect the execution time. All the experiments have been done using the RoboCoT environment. The mission to be carried out by the robots is a modification of the classical foraging task.

4.1 Platform and task description

We use as test bed a multi-robot simulator called RoboCoT (Robot Colonies Tool). RoboCoT is a software tool developed and used by the authors at the University of the Balearic Islands that allows testing the performance of individual or colonies of mobile robots in a very flexible, fast and non-expensive way. This simulator implements robots that work according to a control architecture based on behaviors, as they were introduced by Ronald C. Arkin [1]. It has been widely proved that with these architectures efficient robot navigation capabilities can be obtained at a reasonable low computational cost. A detailed discussion about the RoboCoT architecture can be found in [7].

The task to be carried out by the robots is described as follows: some randomly placed robots must locate objects, randomly placed too, and carry them to a common delivery point. Figure 1 shows a typical initial situation, where the squares represent the objects to collect, the delivery point is the big circle in the middle of the image and the robots are the little circles. Each object to gather has a weight and each robot has a load capacity. The robot load capacity is the amount of weight that it can carry. If a robot cannot carry the entire object at once, it takes a part of it, goes to the delivery point and comes back to the object for more bits. Therefore, unlike a classical foraging task, the objects can be transported by the robots in little transportable bits. It is obvious that in this kind of experiments, a robot can always complete any specific task, carrying bit to bit all the objects to the delivery point. However, if several robots cooperate, the task can be done faster.

The critical question to answer is: how many robots are needed to completely load al the bits of an object? To answer this question we use the equation (1) together with the auction process, explained in the last section. During the experiments, the *taskWorkLoad* value is the object weight and the *workCapacity* of a robot is its load capacity.

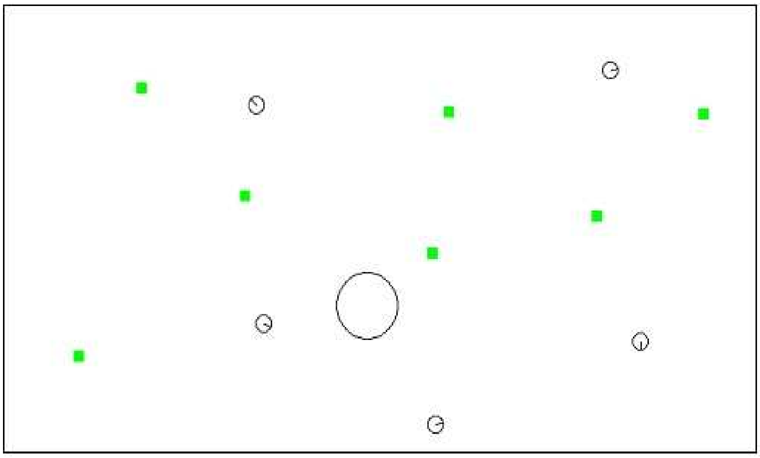

Fig. 1. Example of an initial situation for our experiments.

4.2 Measure of the homogeneity of the collectivity

It has to be emphasized that each robot can have a different load capacity. Thus, we have focused part of our study on the measure of the performance of the collectivity while its homogeneity is changed. To characterize the homogeneity of the collectivity of robots we use two entropy measures. On the one hand, the simple entropy measure based on the Shannon's information entropy:

$$H = \sum_{i=1}^{M} p_i \log_2(\frac{1}{p_i}) \tag{2}$$

where M is the number of classes of robots and pi is the proportion of robots in the ith class. On the other hand, the social entropy as it has been formulated by Balch [2], extends the simple entropy to take into account the quantitative differences between groups. In our experiments, these differences are the load capacity of each robot; therefore, two robots belong to the same group if they have the same load capacity.

4.3 Results

In all the experiments here exposed, we have used seven objects to load and five robots. The objects' weights are: 1, 3, 5, 5, 8, 10 and 15 units. All the robots have the same sensorial, behavioral and communication capabilities, and they only differ in their load capacity. To study the impact of the robots' homogeneity over the execution time we have used four different configurations or combinations for the load capacities of the robots. Table 1 shows these configurations, the load capacities of the robots, and the value of simple (H) and social (SH) entropy of the robot community. For each configuration, we have used as values for the group threshold (TH): 0, 1, 2, 3. In addition, we have executed the experiments without communication between robots. In the case TH=0, the equation (1) has not been used, and therefore the number of robots per group is not limited. We have repeated each experiment 100 times using 20 different environments. The environments are different because of the initial placement of the robots and the objects.

The figures 2 and 3 show the mean of the execution time for our experiments. As it can be seen in these figures, in most cases the equation (1) involves the reduction of the execution time compared to the system

Configuration	R1	R2	R3	R4	R5	H	HS
1 (Homogeneous)	3	3	3	3	3	0	0
2	1	1	1	1	11	0,722	7,22
3	1	1	3	5	5	1,522	4,49
4	1	2	3	4	5	2,322	2,322

Table 1. Robots' load capacities used during the experiments. R1..R5 represent the robots' load capacities. The H represents the simple entropy of the configuration and HS the social entropy.

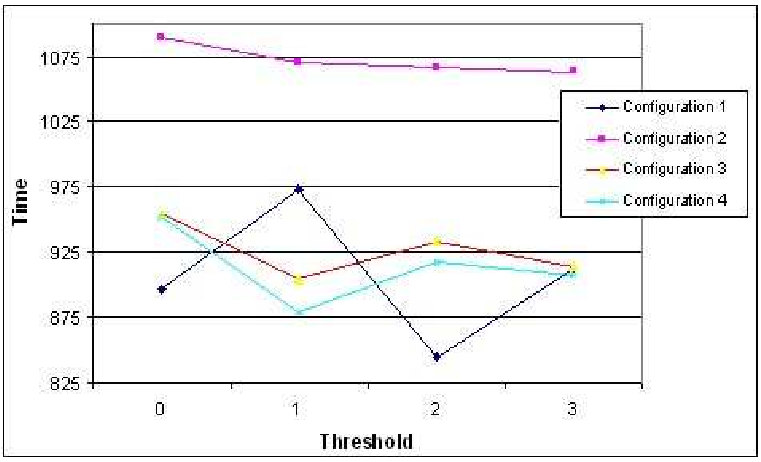

Fig. 2. Execution time as a function of threshold.

with null threshold. As the simple entropy (H parameter) decreases, this benefit is increased. For example, the configuration with greatest simple entropy (configuration 4) presents the greatest benefit (7,69%) of all the cases due to the use of the threshold ($TH \neq 0$). On the other hand, in the homogeneous configuration (configuration 1), the threshold causes the increase of the time. In the worst case, ($TH = 1$) this increase reaches the 8,61%. Figure 2 shows this reduction of the mission's time due to the use of the threshold. As it can be seen in figure 3, the use of communication produces a considerable reduction of the execution time in all the cases, and this time increases as the social entropy (HS) grows. This figure also shows that the worst case is always the configuration with the highest social entropy value (configuration 2).

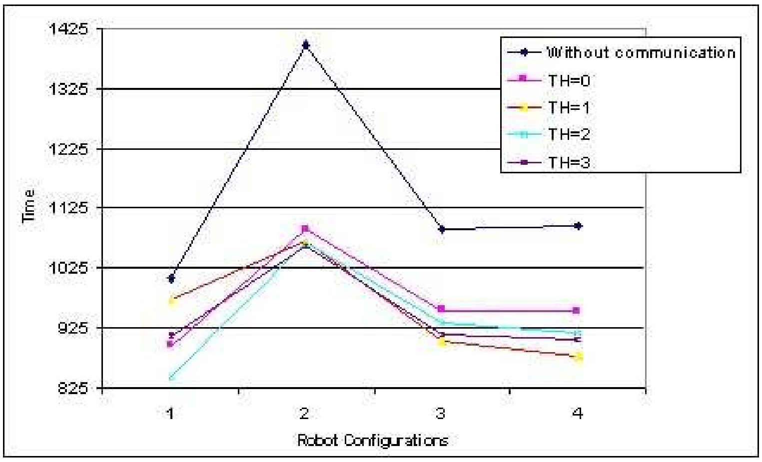

Fig. 3. Execution time as a function of robot configuration.

5 Future Work and conclusion

This paper presents a work in progress that has a lot of challenging aspects to add and to improve. For the time being we are focused on a deep analysis of the data available obtained from a huge set of experiments that should take us to a precise understanding of the relation between heterogeneity and group thresholding as well as robot segregation and aggregation processes or auction algorithms. We also plan to allow the communication among the different group leaders. Allowing this capability, the groups could be seen as entities and negotiation among them could be carried out. The different groups could interchange their robots (work capacities) and they would fit better to their work loads. This interchanging mechanism could allow taking into account tasks priorities. Thus, if a leader finds a high priority task, it could ask for help to other groups with lower priority tasks and some of their robots could be temporarily be transferred to the demanding leader. We are also working on the definition of new protocols to allow the inter-group, or inter-leader, negotiation. Another aspect we are working on is the replacement of the leader mechanisms. These strategies should be activated when the leader robot faults and should improve the robustness of the system. Another aspect of the systems that should be improved is the use of a fixed threshold. Dynamic thresholding techniques in the line of those exposed in [3] will be tested.

This paper presents a simple and efficient way to solve the task allocation problem and, more specially, to decide how many robots are needed to execute a specific task. We have adapted the classical auction system to fit the optimal number of robots. In addition, our method allows changing the robots assigned to a task as new objectives are found. Thus, we have provided a faster and flexible way to regulate the optimal number robots as a function of the kind of the task and the available robots. The system has been validated using a modification of the classical foraging task to allow a higher cooperation degree between robots. The execution results of this task prove that our mechanism reduces the mission's execution time.

References

1. R.C. Arkin, *'Behaviour-Based Robotics'*, The MIT Press 1998.
2. T. Balch, "Social Entropy: A New Metric for Learning Multi-Robot Teams", *10th international Florida Artificial Intelligent Research Society Conference*, Daytona Beach, FL, MenloPark, CA:AAAI Press, 1997 .
3. E. Bonabeau, A. Sobkowski, G. Theraulaz and J. L. Deneubourg, "Adaptive Task Allocation Inspired by a Model of Division of Labor in Social Insects", *Bio Computation and Emergent Computing*, edited by D. Lundh, B. Olsson, Narayanan, pp. 3645, Singapore: World Scientific, 1997.
4. M. Campos, E. Bonabeau and G. Théraulaz, J. L. Deneubourg, "Dynamic Scheduling and Division of Labor in Social Insects", *Adaptive Behavior*, Vol. 8-2, pp. 83-92, 2001.
5. M.B. Dias and A. Stentz, "A Free Market Architecture for Distributed Control of a Multirobot System", *6th International Conference on Intelligent Autonomous Systems*, pp. 115-122, Venice, Italy, 2000.
6. B. P. Gerkey and M. J. Matarić, "Sold!: Auction methods for multi-robot coordination", *IEEE Transactions on Robotics and Automation, Special Issue on Multi-robot Systems*, Vol. 18, No. 5 pp. 758-768, 2002
7. J. Guerrero, G. Oliver and A. Ortiz, "On Simulating Behaviour-based Robot Colonies in the Clasroom", *First EURON Workshop on Robotics Education and Training*, pp. 91-98 Weingarden, Germany, 2001.
8. M. J. Krieger, J. G. Billeter and L. Keller, "Ant-like task allocation and recruitment in cooperative robots", *Nature 406*, pp. 992-995, 2000.
9. Kristina Lerman, Aram Galstyan, "Mathematical Model of Foraging in a Group of Robots: Effect of Interference", *Autonomous Robots 13 (2)*, pp. 127-141, 2002.

This article was processed using
the TEX macro package with ECMR2003 style