Physical interference impact in multi-robot task allocation auction methods

José Guerrero and Gabriel Oliver
Universitat de les Illes Balears, Mathematics and Computer Science Department
Cra. de Valldemossa, Km. 7,5, 07122
Palma de Mallorca, Spain
e-mail:{jose.guerrero, goliver}@uib.es

Abstract

Task allocation is one of the main problems in multi-robot systems. Among other factors, to get a good task allocation, we have to take into account the physical interference effects between robots, that is, when two or more robots want to access to the same point at the same time. This paper analyzes interference impact using auction methods, one of the most popular task allocation systems. We will show how the performance of the auction utility function can be improved if interference impact is included in it. We also provide a framework to simplify one of the main problems of all auction systems which is finding a good utility function. Our method has been tested using transport like tasks, where each object must be transported to a delivery point before a deadline. This is a simple task that is very useful to isolate the interference effects.

1. Introduction

Multi-robot systems can provide several advantages over single-robot systems: robustness, flexibility and efficiency among others. To benefit from these potential aspects several problems have to be solved. Among all these problems, we focus on task allocation issues, that is, selecting the best robot or robots to execute a task. Some tasks require that two or more robots cooperate to execute them creating coalitions. In this case we have to find the best set of robots to execute the task and also the optimum number of these robots. As it has been demonstrated in different studies [4], the number of robots has an important impact on the system performance, among other factors, due to the physical interference effect. Interference appears when two or more robots need to reach the same point at the same time. This factor has only been modeled and analyzed using very simple environments [4]. A lot of research has been done to solve the task allocation and coalition formation problems but they are still open problems. One of the most used and well studied task allocation solutions are auction methods [1]. Only a few auction strategies, like [5], allow to allocate several robots to the same task, but these methods don't take into account the interference effect. Moreover, one of the main problems of all auction methods is to find or to learn a good utility function. The utility function depends on a lot of factors, and very specially on the interference effect.

In this paper we analyze how to introduce the interference effect in the auction's utility functions. We study the impact of this factor showing that the system performance can be improved when interference is taken into account. To our knowledge, this is the first time that this kind of analysis is made using auction strategies. This paper also proposes a framework to reduce the complexity on finding utility functions when robots must create coalitions. This framework divides the learning process in three stages. During the first phase, only the knowledge of individual robots is included in the utility function. The second phase includes the knowledge about robots that form the coalition. Finally, the last stage include the information of other coalitions. One of the most importat information that should be included in the second phase is a measure of the physical interference produced between robots of the same group. To test our system we use a foraging like task, where the robots must find a set of objects and carry them to a delivery point. During this task, multiple robots can cooperate to transport the same object. In this case we have to decide how many robots and which ones do we need to transport each object according to its weight and to the robots characteristics. Each task, or object, must be executed before a deadline, and the goal of our auction method is to maximize the number of tasks executed before their deadline. The results show that including the interference information in the utility function, the system performance can be significantly improved.

The rest of this paper is organized as follows: section 2 presents the task allocation algorithm used; sections 3 exposes the designed framework to simplify the search of good utility functions; section 4 analyzes the effect of the physical interference on the utility functions; section 5 shows the results of the task allocation system using the interference effect; finally, section 6 exposes some conclusions and future work.

2. Task allocation

Our task allocation mechanisms, including the groups' formation, membership policy and task assignment is briefly described in the following paragraphs. This new mechanism extends our previous work [3] to allow deadlines. Here we only show very briefly the main aspects of our method to understand the interference model that will be explained during section 5.

Classical auction methods has been modified to select which robots, and very specially, how many of them are needed to execute a task. In an initial stage, each robot is looking for a task. When a robot finds a new task, it will try to lead it. There is only one leader for each task. If a robot is promoted to leader, it will create, if necessary, a work group; that is, a set of robots that will cooperate to execute this specific task. In that case, the leader must decide which the optimum group size is and what robots will be part of the group. To take this decision, the leader uses an auction like mechanism. During this process robots bid using their work capacity. The work capacity is the amount of work that a robot can execute per time unit, thus, this value is the utility function of our auction method. The leader selects the robots with the highest work capacity, until it detects that the group is able to reach its deadline, that is, until this condition is verified:

$$DL_g = \frac{taskWorkLoad}{groupCapacity} \leq DL \tag{1}$$

Where $taskWorkLoad$ is the amount of work required to finish the assigned task that is calculated by the leader; $groupCapacity$ is the work capacity of the group, that is, the amount of work that the group can process each time unit and, finally, DL is the deadline of the task. As it can be seen, DL_g is the expected time required to finish the task. In general, the work capacity of the group is not the sum of the work capacity of each single robot, that is $groupCapacity \neq \sum_{1 \leq i \leq N} workCapacity_i$, where N is the number of robots of the group. Among other factors, this inequality is due to the interference effect. During the following sections we will try to find a good method to calculate individual utility of each robot and specially the group utility. From this point, we will consider robot utility and group utility synonymous of robot's work capacity and work capacity of the grup. As it has been said earlier, the details of this task allocation method, like how a robot can be promoted to a leader, can be found in [3].

3. A framework to get utility functions

This section describes a framework that will help us to get a good utility function for auction methods when robots must create coalitions. Getting a good utility function is a difficult process, very specially when the robots must form coalitions or when the utility of a robot depends on the utility of other robots. We can use learning algorithms to get this function, but these algorithms require a lot of time, and it is not clear what the robot has to learn. Also, in general, utility functions are not linear, so the learning process can be very hard. To simplify the process, some parameters can be analyzed previously, using an ideal environment, and then modified during the execution of the task. We will do this in 3 steps:

- Individual utility: during the first stage, we evaluate the characteristics of each single robot without taking into account the others. Here it will be include some characteristics like velocity, acceleration, etc.
- Group utility: in this step, the robot will take into account the other ones to create a coalition or working group. Here some parameters, like interference effect, will be included. That is, the robots will calculate the utility function of the group.
- Inter-Group utility: finally, the robots have to take into account that the decision of one group can affect to other groups. This inter-group dependency must be included in the utility function during the final step.

In this paper we analyze the two firsts steps paying spacial attention to the interference effects. As an example of how to use the framework proposed, a transport like task will be used. 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. To maintain the initial conditions, when an object is fully transported to the delivery point, it immediately appears another one, with identical characteristics in a random place. Figure 1 shows a typical initial situation, where the squares represent the objects to collect, the delivery point is the big circle at the right 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 at once. Thus, 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. Of course, this is a very simple environment but it allows us to isolate the interference effects of other factors that can appear in more complex tasks. It is under study whether a similar reasoning can be made for different kind of tasks, like exploration, surface cleaning or mapping for example.

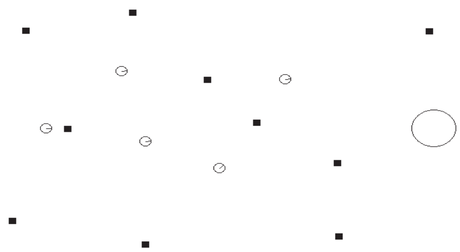

Figure 1. Example of initial situation of transport task

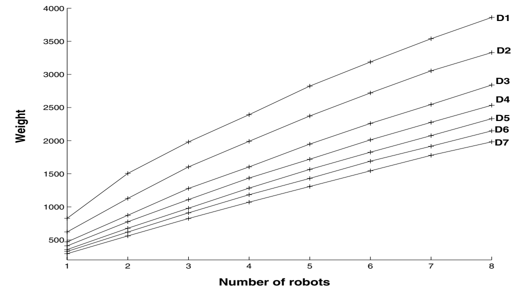

Figure 2. Total transported weight during the experiments to model interference effect

4. Individual utility function

Now we will describe the first step of our framework, called individual utility, to find the utility function of each single robot. As an example, the transport task explained during the last section will be used. The work capacity of a robot is the amount of object's weight that this robot can transport to the delivery point per time unit. Under ideal conditions, that is, we supose an open environment without any other obstacle or robot between the object and the delivery point, the robot's work capacity is easy to calculate. Let r_i be a robot and C_i the load capacity of the robot. V_i is the maximum velocity, d the distance between object and delivery point and C the weight of the object. The number of trips between the delivery point and the object that the robot must do to transport the full object is $2 * \frac{C}{C_i}$. If we neglect the acceleration and deceleration time, for each one of these trips the robots will need $\frac{d}{V_i}$ time units. We also consider that a robot needs one time unit to load or to unload each weight unit. For example, if the robots has to load 2 weight units it will require 2 time units to load all this weight and 2 time units more to unload when it arrives to the delivery point. Thus, to load and to unload the full object, the robot will spend $2C$ time units. Therefore, the total time required to transport the full object is $T = 2\frac{C}{C_i}(C_i + \frac{d}{V_i})$ and the work capacity is $\frac{C}{T}$, that is:

$$workCapacity_i = \frac{C_i V_i}{2(C_i V_i + d)} \tag{2}$$

5. Group utility function: interference effect

During this section we will analyze the second step of our framework to get the utility function of the group. We will also use the transport task to show how to get this function value and, therefore the utility of the group will be the expected amount of work that the set of robots can execute per time unit. This value is not the sum of each individual robots' work capacity because of the interference effect, among other factors. Thus, first of all, we will model this interference effect and then we will show how to use this information to get the utility functions of the robots. All the experiments carried out to model the interference have been executed using a multi-robot simulator called RoboCoT (Robot Colonies Tool). RobotCoT is a software tool developed by the authors at the University of Balearic Islands [2].

5.1. Interference effect analysis

To analyze the interference effect we have executed a task where several robots must transport a single object and the total weight transported by the robots after 40000 time units is calculated. All the robots have the same load capacity (2 weight units) and the same velocity (3 distance units/time units), and therefore, all they have the same work capacity. Moreover, the environment doesn't have any obstacle, only exists the robots, the object and the delivery point. Seven diferent distances between object and delivery point have been tested: $D_1 = 138$ units, $D_2 = 184$ units, $D_3 = 247$ units, $D4 = 282$ units, $D5 = 329$ units, $D_6 = 359$ units and $D_7 = 399$ units. Figure 2 shows the total transported weight during these experiments when the number of robots varies from 1 to 8. As it can be seen, and as has been pointed by other authors, the relation between the number of robots and the transported weight is not linear. The difference between the expected transported weight, calculated as the sum of the individual robot's work capacities, and the real transported weight can only be due to the interference. This figure also shows that the interference effect increases as the distance between the object and the delivery point decreases. To analyze the interference effect, two models have been used: a polynomial fit and an exponential fit.

The polynomial fit model suposes that the work capacity of the group follows this equation:

$$groupCapacity = \sum_{1 \le i \le N} workCapacity_i - I(N) \quad (3)$$

where $workCapacity_i$ is the individual work capacity of the ith robot of the group, calculated using equation 2; N is the number of robots of the group and $I(N)$ is a polynomial of degree 2 that fits the interference effect as a function of the number of robots. We have used this function because of its simplicity and it fits with a very low error the real results. Moreover, due to its simplicity, only 3 parameters must be adjusted. We have also tested polynomials of higher degrees, but the results do not improve significantly, and due to their complexity they have not been used during the experiments. Thus, this polynomial models the difference between the expected work capacity without interference and the results of our simulations. Function $I(N)$ has the following form:

$$I(N) = \alpha N^2 + \beta N + \gamma \quad (4)$$

Table 1 shows the values of the parameters of function $I(N)$, and figure 3 shows the form of the $I(N)$ function that fits the real results. To improve the quality of the figure, only some distances have been represented (D_1, $D2$, $D3$ and $D7$). The crosses correspond to real data. The y axis represents the interference effect for each 1000 time units, that is, $1000 * I(N)$. The resulting parameters seem to be very low, but it should be pointed out that the utility of each robot is also very low because of the high values of the distance value (d) of the individual utility equation, as expressed in equation 2. However, the interference effect can modify very significantly the group utility, for example, when the distance is D_7 and there are 8 robots, the interference decrease the work capacity of the group about a 76%. Moreover, as it can be seen in table 1, as the distance between the object and the delivery point increases, the values of α and β decrease. For the time being, a new function which relates the interference to the distance and the number of robots is under study. Finally, we have to note that the errors between the real results and the interference function fitted are, in general, very low but they increase as the number of robots is reduced. In fact, our equation is not suitable when the distance between the object and the delivery point is very low and $N = 1$.

Using all this information, the utility function of a robot, calculated in equation 2 must be modified. In fact, for a group with N robots, this new utility measure can be defined as the difference between the utility of the working group with and without the Nth robot. Therefore, this equation must be used:

$$utility_j = groupCapacity_t(N) - groupCapacity_{t-1}(N-1) \quad (5)$$

-	α	β	γ
D_1	0.3589	7.5029	-12.3571
D_2	0.3476	2.964	-4.2857
D_3	0.2432	1.3347	-2.0107
D_4	0.2006	1.062	-1.6321
D_5	0.1619	0.4737	-0.6071
D_6	0.1494	0.37337	-0.5071
D_7	0.1071	0.405	-0.5

Table 1. Parameters of the interference function for each 1000 time units($I(N)$)

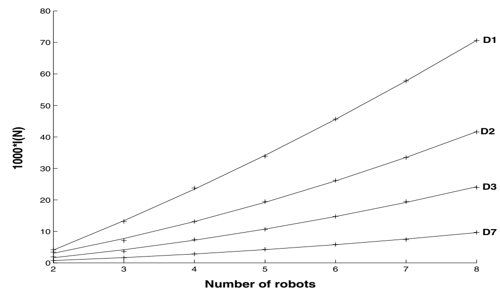

Figure 3. Polynomial fit of $I(N)$ for different values of distance between the object and the delivery point

Where $groupCapacity_{t-1}(N-1)$ is the work capacity of the group before the new robot was added; $groupCapacity_t(N)$ is the group capacity with the new robot and N is the number of robots of the group included the new one. Therefore, if this equation is calculated, we can find that the utility of a new robot, $utility_j$, is:

$$utility_j = workCapacity_j - (\alpha(2N-1) - \beta) \quad (6)$$

Using the market based system vocabulary [1], we can say that the left side of the equation 6 is the benefit that the robot will get if it executes the task and the right side is the cost of this execution. We can also note that the utility of each single robot only depends on two values, α and β. Thus, future learning algorithms only will need to tune this two parameters, instead of making large searches in unknown state spaces.

The second model that has been tested to fit the interference is the exponential fit, where we supose that the group work capacity can be modeled using this equation:

$$groupCapacity = \sum_{1 \le i \le N} workCapacity_i (1 - e^{p(N)}) \quad (7)$$

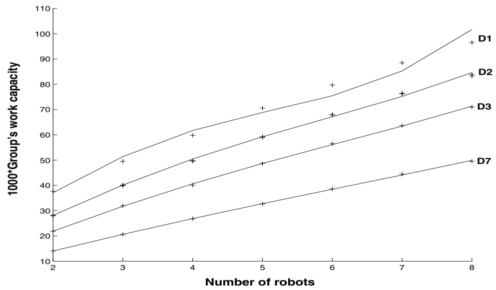

Figure 4. Exponential fit for work capacity of the group using different values of distance between the object and the delivery point

Where N is the number of robots of the group; $workCapacity_i$ is the work capacity of the ith robot and $p(N)$ is the polynomial that we have to find. After several experiments we found that the best polynomial was a 2 degree one, that is, p(N) has the same form of $I(N)$ but with other parameters α, β and γ. Again, this function only is valid for $N > 1$. This is not a serious problem because it doesn't exist interference when there is only a robot. Figure 4 shows the form of the $groupCapacity$ using equation 7. In this figure only some distance has been represented (D_1, $D2$, $D3$ and $D7$) and the crosses are the real data. In all the cases the error of the estimated work capacity of the group is lower than a 5.3%.

5.2. Interference and task allocation

This section will show how to modify the auction process explained in section 2 to take into account the interference effect. We only will use the polynomial fit explained during the last section due to its simplicity and good results. The eventual benefits of the exponential fit are under study.

As has been explained in section 2, the leader selects the best robots (robots with the highest bids) until the equation 1 is verified. Now, using the interference information, the leader of the group will only include a new robot if this operation increases the work capacity of the group. Therefore, the auction process will finish when the equation 1 or when this condition is verified. Thus, now the robots will not only bid using their work capacity, but also including the interference factor. In the bidding process, the robot only bids to the group leader its work capacity, and then this leader calculates the correct bid with equation 5. Using this extra information about interference, and as will be seen during the next section, the robot can find a better set of robots to verify the deadline.

6. Task allocation experiments

During this section we will show the results of several experiments performed to study the impact of the physical interference on our auction method. During all the experiments RoboCoT has been used, which is the same simulator used during section 5. The robots must execute the transport task explained in section 2. The transport tasks have a deadline. The main objective is to transport each object before its deadline. If the fulfilment of this objective is not possible, the robots continue their execution until the object is fully transported. The time to deadline starts when the object appears in the environment. Thus, we prioritize the accomplishment of the tasks' deadline over the increment of the total transported weight. To simplify the analysis, the robots know the situation of each object in the environment.

During all the experiments we use 10 robots and 3 objects to gather. All the robots have the same characteristics than during the experiments of section 5, that is, their load capacity is 2 and the velocity is 3. All the tasks have a weight equal to 40 weight units. Different values of deadline have been tested: 1500 time units, 1200 time units, and 900 time units. Deadline value is the same for all the tasks, so if there is a long distance between the object and the delivery point, this task will require more robots than a nearer one. Robots carry out the mission during 35000 time units. After this period, we get the time required to transport each object and the number of object gathered. Each experiment has been repeated 4 times.

Figures 5 shows the results when the deadline is equal to 1200 time units without taking into account the interference effect in the robot's work capacity. Thus, to calculate the work capacity of the group it only has been used equation 2 and not the interference effect of equation 3. In this figure it can be seen the percentage of robots that fulfil the deadline. A bar with the label 0.6 represents the percentage of tasks that its execution time exceeds a 60% of the deadline. A bar with the label 0.5 represents the percentage of tasks that require less than a 60% and more than 50% of the deadline time, etc. The negative numbers represent the tasks that have been fulfilled the deadline. For example, the bar with -0.2 represents the task that requires to finish less than a 20% of the deadline time. As it can be seen, 63,4% of the objects have been executed before the deadline. During all the experiments the robots transported 557 objects. On the other hand, figure 6 shows the results of the same experiment but including the interference effect into the utility function. Now, the number of tasks that fulfil the deadline is a 88,7%, a 25,3% more tasks than during the experiments without interference. Therefore, interference has a clear impact on the system performance, and it can be used to fit better the number of robots needed to fulfil the tasks's deadline. Moreover using interference the robots transported

604 objects, a 8,4%more than without using interference.

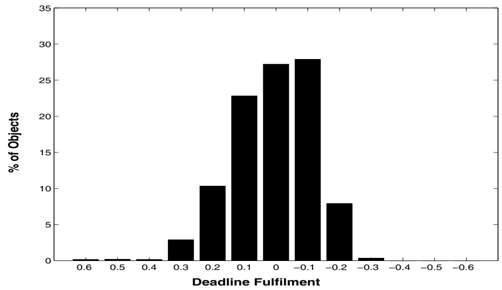

Figure 5. Deadline fulfilment without using interference effect and with a deadline equal to 1200 time units

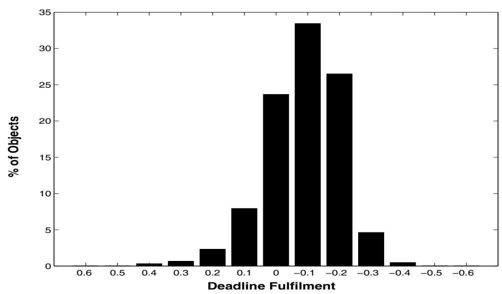

Figure 6. Deadline fulfilment using the interference effect into the utility functions and with a deadline equal to 1200 time units

The experiments using a deadline of 1500 time units do not present significant differences between including and not including the interference effect. This is because the robots have enough time and work capacity to reach the deadline in both cases. On the other hand, when the deadline is set to 900 time units, the results (not shown graphically here) demonstrate that, if the interference factor is not used, only a 24% of the tasks fulfil the deadline. Once again, this percentage is increased up to a 42,6% when the new proposed method is used.

7. Conclusion and future work

This paper analyzes the impact of the interference effect on the utility function used in an auction like system. First of all, an auction method has been presented that, unlike other auction methods, allows to assign multiple robots to the same task creating coalitions. Moreover, our method includes the concept of deadline, that is, the idea that a task should be executed, if possible, before certain period of time. One of the main problems of all the auction systems is to find a good utility function. To simplify this problem when the robots must create coalitions, we propose a framework that divide the search in 3 steps. The first and the second step of this framework have been studied for the execution of transport like tasks. One of the main aspects that we have to take into account to calculate the utility function is the physical interference between robots. This influence has been analyzed and fitted using a polynomial function. The experiments carried out show that including interference in the utility functions the robots can better fulfil the tasks's deadline. Thus, the importance of interference has ben showed. To our knowledge, this is the first time that this study has been made for this kind of tasks.

The work presented is in progress and has some challenging aspects to add and to improve. We are working to use a preemption auction method, that is a method that allow the exchange of robots between working groups. We also will study the interference effect between robots that belong to different groups, and thus, complete the last step of the framework presented in section 3. Moreover, learning algorithms will be introduced to find other parameters of our system, and a deeper analysis of the exponential fit function will be done. Finally, we will extend this experiments using real robots and other kind of tasks, like exploration and other environments with more obstacles. During these new experiments other factors, like the energy of the robot, will be taken into account to select the best robots for each task.

Acknowledgments

This work has been partially supported by project CICYT-DPI2005-09001-C03-02 and FEDER fundings.

References

[1] M. B. Dias and A. Stentz. Traderbots: A market-based approach for resource, role, and task allocation in multirobot coordination. Technical Report CMU-RI-TR-03-19, Carnegie Mellon University, The Robotics Institute, Pittsburgh, August 2003.

[2] J. Guerrero and G. Oliver. On simulating behaviour-based robot colonies in the classroom. In *First EURON Workshop on Robotics Education and Training*, Weingarden (Germany), 2001.

[3] J. Guerrero and G. Oliver. Multi-robot task allocation method for heterogeneous tasks with priorities. In *7th. International Symposium on Distributed Autonomous Robotic Systems*, pages 311–317, Tolouse (France), June 2004.

[4] K. Lerman and A. Galstyan. Mathematical model of foraging in a group of robots: Effect of interference. *Autonomous Robots*, 13(2), 2002.

[5] L. Vig and J. A. Adams. A framework for multi-robot coalition formation. In *Proceedings of the 2nd Indian International Conference on Artificial Intelligence*, 2005.