

# Scalable Task Assignment for Heterogeneous Multi-Robot Teams

Regular Paper

Paula García<sup>1</sup>, Pilar Caamaño<sup>2</sup>, Richard J. Duro<sup>2</sup> and Francisco Bellas<sup>2,\*</sup><sup>1</sup> Grupo de Arquitectura de Computadores, Dpto. de Electrónica y Sistemas, Universidade da Coruña, Spain<sup>2</sup> Integrated Group for Engineering Research, Universidade da Coruña, Spain

\* Corresponding author E-mail: francisco.bellas@udc.es

Received 14 Jun 2012; Accepted 12 Dec 2012

DOI: 10.5772/55489

© 2013 García et al.; licensee InTech. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

**Abstract** This work deals with the development of a dynamic task assignment strategy for heterogeneous multi-robot teams in typical real world scenarios. The strategy must be efficiently scalable to support problems of increasing complexity with minimum designer intervention. To this end, we have selected a very simple auction-based strategy, which has been implemented and analysed in a multi-robot cleaning problem that requires strong coordination and dynamic complex subtask organization. We will show that the selection of a simple auction strategy provides a linear computational cost increase with the number of robots that make up the team and allows the solving of highly complex assignment problems in dynamic conditions by means of a hierarchical sub-auction policy. To coordinate and control the team, a layered behaviour-based architecture has been applied that allows the reusing of the auction-based strategy to achieve different coordination levels.

**Keywords** Auction-Based Task Assignment, Multi-Robot Coordination, Behaviour-Based Architectures

## 1. Introduction

Facing real-world problems like multi-target observation (surveillance), exploration or collective object

transportation with a team of autonomous robots implies taking into consideration two very relevant aspects. Firstly, and in general, these types of missions require a division into non-trivial subtasks to be solved efficiently, and the robots that make up the team must be dynamically assigned to them during real time operation. Secondly, as a consequence of the above, the robots that make up the team might end up having to be heterogeneous, as it may be impractical to have robots with the complete set of hardware capabilities needed to individually solve all the subtasks. This introduces a higher level of complexity in the task assignment structure.

Therefore, the multi-robot control architecture that manages the system must be capable of dealing with robots of different functionalities and coordinating them in real time. Several examples of coordination architectures may be found in the literature, ranging from strongly centralized and strongly coordinated systems to distributed ones with no coordination at all, all of them providing successful application results [1]. However after reviewing this topic in depth, it can be concluded that there exists a clear tendency towards designing decentralized and emergent coordination control architectures in homogeneous multi-robot systems

(MRSs) made up of several simple units, as in the case of swarm robotics [2], being centralized and strongly coordinated MRSs more typical of heterogeneous robot systems usually in the form of small teams. One of the main reasons for this, as commented in [3], is that when concentrating on real-world applications, system robustness and fault tolerance become mandatory, implying a stronger coordination strategy with direct and explicit communications (as opposed to implicit and indirect ones, known as 'stigmergy') that can be taken under control.

Here we are interested in this last type of MRS as well and, therefore, we can assume that the robots cooperate explicitly and with a purpose, using communication strategies to achieve this coordination. Thus, the main problem faced here is how to efficiently assign the subtasks that the global mission must be divided into. This is a widely discussed topic in the MRS field that has been formalized by Mataric and Gerkey in [3] and that has been confronted using different techniques, like potential fields [4], control theory [5], swarm intelligence [6], social structures [7] and metaheuristics [8]. However, when dealing with heterogeneous MRS and non-trivial task assignments (the same robot can accomplish different tasks or the number of robots is larger than the number of tasks), auction-based strategies have been shown to be one of the most efficient and robust approaches [9][10][11][12].

One of the main features of auction-based strategies is their simplicity, requiring uncomplicated algorithms to organize complex teams and subtasks in dynamic problems [13][14]. However, in most approaches presented in the literature a problem of scalability and dynamicity arises that tends to hinder these processes as the number of robots in the team or the complexity of the problem increases, especially when this occurs in real-time. The objective of this paper is to address these issues, proposing a very simple auction strategy that can be very easily scaled to distributed multi-sub-auctions as the problem becomes more complex and distributed. We will show how the computational cost increases linearly with the number of robots in the team in simple cases and how, through a hierarchical sub-auction policy, highly complex assignment problems can be solved while maintaining the linear computational cost increase. This last analysis is the main contribution of this work, as the automatic creation and management of sub-auctions in real cases has rarely been addressed within the field [14].

To carry out the study, we have set up a collective cleaning problem to be accomplished by a team of heterogeneous robots that are controlled by means of a behaviour-based coordination architecture. Although the architecture is not the objective of this paper, it must be formally defined to better understand the specific

auction-based strategy that has been implemented and the overall behaviour of the MRS. Moreover, the architecture allows the reuse of simple behaviours that will be necessary to easily create the hierarchical sub-auction strategy.

The remainder of the paper is structured as follows: section 2 contains the formal description of the behaviour-based architecture that provides the controllers of the robots that make up the MRS and the description of the specific auction-based task assignment strategy that has been implemented. Section 3 is devoted to the presentation of the collective cleaning experiment and the main results obtained in different setups of increasing complexity. Finally, section 4 contains the main conclusions that can be extracted from this work.

## 2. Coordinated control and task assignment

### 2.1 Behaviour-based architecture

When a real-world task is addressed using a MRS, coordination between robots seems to be mandatory (although completely uncoupled missions could be efficiently solved by independent robots). As a consequence, the robots that make up the team must be endowed with some type of coordinated control to accomplish the task properly and in an efficient manner. To this end, a coordination architecture that supports the definition and implementation of the control system is needed.

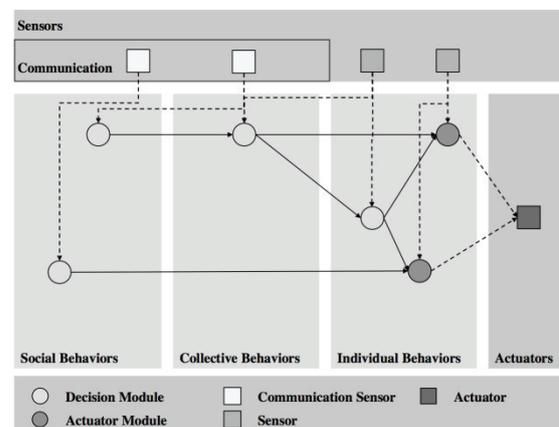


Figure 1. Schematic of the coordination architecture.

Here, a set of three basic requirements were imposed on the design of the MRS coordination architecture: an ability to act autonomously and independently, and a capacity for cooperation and social development. This has led to a hierarchical structure for the proposed coordination architecture with three blocks, as shown in the diagram of Fig. 1. This hierarchical division helps to structure the components of the system into levels, favouring abstraction and providing mechanisms for them to share features. As

we go from right to left in the architecture, a higher level of abstraction is achieved. Each level has a different specific purpose, and is made up of modules with independent behaviours. Specifically, we have:

1. **Individual behaviours:** corresponding to the lower level of the control architecture. They provide the robot with the ability to act autonomously and independently, without requiring help from other robots in the system. There is no communication between robots at this level of the architecture. The inputs to the modules' behaviours come from the robot's sensors, providing information on the state of the environment and also from the communication sensors. Only the modules on this level are connected to the robot's actuators.
2. **Collective behaviours:** corresponding to the second level of the control architecture, where the degree of abstraction increases. It provides the robot with the ability to cooperate or collaborate with other robots in the system in order to achieve a common objective. To do this, communications between robots - whether direct or indirect (stigmergy) - are necessary. Accordingly, the behaviour modules belonging to this level receive inputs from the sensors of the robot, as was the case for the previous one. The output of these modules can only modify the individual behaviour modules of the lower level (for instance, through modulation [17]).
3. **Social behaviours:** corresponding to the highest abstraction level of the control architecture. These have to do with behaviour changes in the conduct of a robot due to the fact that it belongs to a society with certain customs and rules. The application problems - which we will present later - do not include any module at this level but it is, nonetheless, a core feature of the global architecture and we include it in the diagram for the sake of completeness. As in the previous case, the modules in this level obtain their inputs from the sensors of the robot and modify the behaviours of the level below, i.e., collective behaviours. Thus, cooperation in the MRS is achieved by the collective behaviours level, and this third level can modify such cooperation based on social parameters (for instance, it may introduce preferences as to types of robot with which it is more advisable to collaborate).

Hierarchical architectures are very popular in autonomous robotics [15] and they can also be found in MRSs with a similar organization to the one displayed in Fig. 1 [16]. However, in this case a modular structure is included to simplify the hierarchical interactions. All the behaviour modules in the architecture shown in Fig. 1 have inputs, a control strategy and outputs. The individual control of each robot must be robust and energy efficient, but we do not impose any constraint on the particular strategy (neural control, finite state machine, rule-based system, etc.) used in each module.

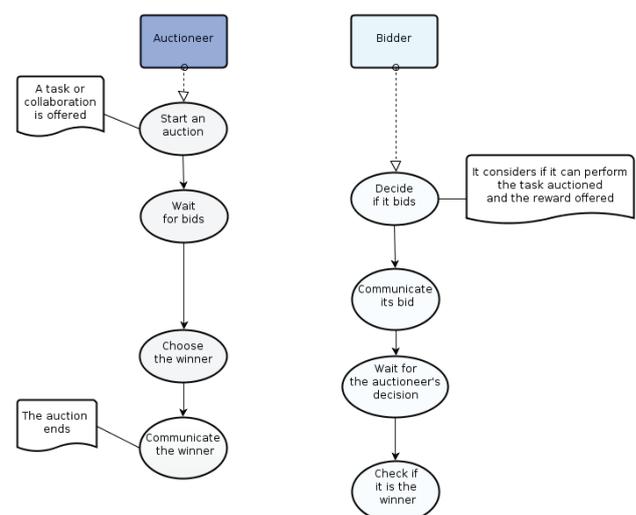
The inputs always come from the sensors and the outputs can be directly connected to the actuators of the robot or can control other modules. As shown in the figure, within each level, the behaviour modules are organized as a hierarchy and some of them (decision modules) control the behaviour of others (actuator modules). This philosophy has been extracted from previous work carried out in our group for single robot systems [17] and recently expanded to collective tasks [18]. According to it, once we have a problem to solve, it must be decomposed into subtasks with an associated primitive behavior module. Complex behaviours can be obtained by combining simple ones, either directly by hand or else by using an automatic procedure, for example, a genetic algorithm [18].

Thus, in using this architecture, a general approach to obtain cooperating behaviours in a MRS is achieved from an organizational point of view. However, the particular modules that implement a particular coordination strategy depend upon the application.

## 2.2 Auction-based task assignment strategy

Auction-based strategies are inspired by the economic techniques of the free market where individuals act to pursue their own interests yet this behaviour results in a productive society. Sometimes, the individuals compete with each other and sometimes they collaborate to try to produce goods or services at the minimum cost to maximize profit.

Within this economic framework, auction-based mechanisms include a public sale in which there is an auctioneer or seller who sells one or more goods and some bidders or buyers who bid against each other, offering to pay a particular price for the goods under auction. From a practical point of view, auctions are efficient and simple techniques for allocating goods or tasks to a group of sellers.



**Figure 2.** Operational diagram of the auction steps performed by the auctioneer robot and by the bidder robot.

In this work, and with the aim of using the simplest algorithm, the assignment of subtasks in the MRS mission is carried out through a single round auction. There is a member of the team that acts as the auctioneer while the other robots are the bidders. Fig. 2 summarizes the operations performed by each type of robot according to this type of strategy and provides an overview of its simplicity.

To deal with complex assignment problems, the single round auction described in Fig. 2 can be hierarchically repeated creating sub-auctions where a bidder robot in a higher-level auction becomes the auctioneer if its individual control system decides to do so. A specific implementation of this hierarchical auction process will be explained in section 3.2. With this setup, many auctions can be carried out in a distributed manner at different coordination levels without any change in the cooperative control system.

### 3. Multi-robot cleaning scenario

The main purpose of this work is to analyse the scalability of a simple auction-based task assignment strategy in realistic heterogeneous multi-robot missions. The problem selected as a testing platform is a collective cleaning task. This type of problem is a typical benchmark [1] in strongly coordinated MRSs because it contains all the elements of a real-world problem suitable for them: due to cost, the robots that make up the team are usually specialized in different aspects of cleaning, it requires strong synchronization to be efficiently performed, the sub-task assignment is highly dynamic because the robots must be reassigned to new areas continuously, the complexity of the task can be very high depending upon the physical features of the zone to be cleaned, and so on. Consequently, the conclusions extracted from this experiment can be easily transferred to other real-world multi-robot problems.

For the sake of clarity, we have divided the experimental analysis into two cases. First, we will describe the experiment, the control system and the auction strategy in detail by means of a simple task assignment problem where the scalability of this approach will be analysed in terms of computational cost. In the second part of the experiment, we will study a realistic case where a more complex assignment is required and where the hierarchical sub-auction policy proposed above will be analysed.

#### 3.1 Simple task assignment

To carry out the experiments, we have made use of the Stage 2D robotic simulator [19] to create the simulated cleaning scenario. Thus, a simplified version of a real cleaning task has been created and is displayed in the screenshots of Fig. 5. We can see an environment made

up of 4 rooms with two different types of surfaces that can be distinguished by their colour (red and green). The problem contains two subtasks represented by the two different types of surfaces, which must be cleaned using different tools. These tasks can be accomplished simultaneously. Just to make things simple for the reader, we have called these two tasks vacuuming (green surface) and mopping (red surface), and the specialized robots that carry each one are the vacuum cleaner and the mop. In fact, it does not really matter what the individual tasks are, just that they require a differently tooled robot to carry them out.

Regarding the robots, the MRS is made up of  $n$  heterogeneous robots of three different types:

- **Coordinator:** there is just one coordinator (red in Fig. 5). It is the only robot equipped with a camera, so that it can detect the type of room surface. In addition, it has sonar sensors to navigate.
- **Vacuum cleaner:** there are several robots of this type (represented in Fig. 5 using blue) and they are equipped with sonar and laser sensors to navigate and follow the coordinator and with a tool that is appropriate for vacuuming (green surface).
- **Mop:** there are several robots of this type too (represented in Fig. 5 using yellow) and they are equipped with sonar and laser sensors to navigate and follow the coordinator and with a tool that allows them to mop (red surface).

Obviously, the final objective of the MRS is to clean the four rooms efficiently - that is, to assign the vacuum cleaner robots to the green rooms and the mop robots to the red ones. To do this, the coordinator must detect the type of room and start a single auction, where the robots can bid according to their preferences, as we will explain later.

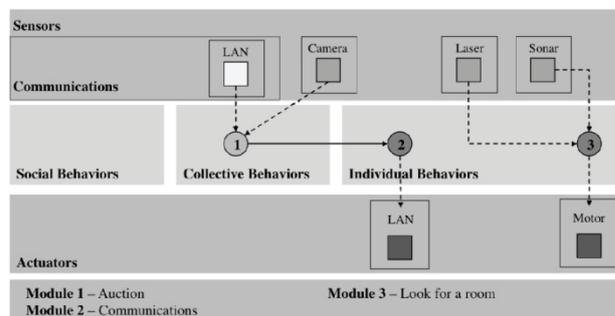


Figure 3. Behaviour modules for the coordinator robot

Regarding the controllers of the robots, they have been designed using the cooperation architecture presented in section 2.1 and executed using the Player/Stage framework. Specifically, the behaviour modules corresponding to the different types of robots considered are as follows:

- The coordinator robot has two individual behaviours (look for a room and communicate) and one collective behaviour (auction). Fig. 3 displays these behaviours in the architecture's representation, connected to the corresponding sensors and actuators. According to the architecture's principles, the auction behaviour is a decision module that affects a low-level behaviour - in this case, the communication behaviour.
- The cleaner robots - both vacuum cleaners and mops - only have individual behaviours in this case: look for the coordinator, follow the coordinator, communications and cleaning (the last one uses a different tool for each type of robot due to their heterogeneity). These behaviours are shown in Fig. 4.

All of the modules are internally represented using a simple finite state machine, but they solve their particular task successfully. In this work, we have used such simple controllers because we want to focus our attention on the coordination parameters rather than on individual task improvement.

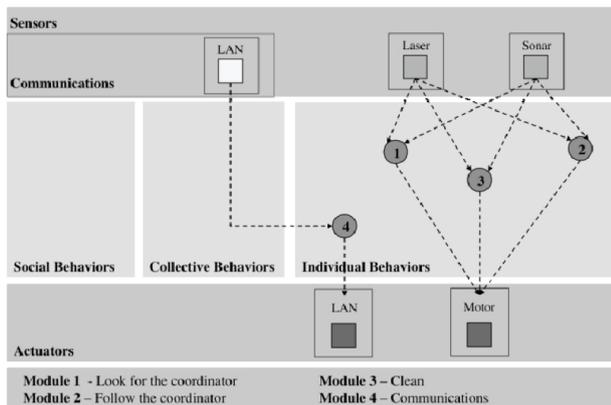


Figure 4. Behaviour modules for the cleaner robots

As we can see from this description, this first experiment is a very simple but common assignment problem. Here we want to analyse the performance of a simple single round auction-based strategy as the number of robots is increased, which implies that the number of bidders increases too, making the assignment problem more dynamic (the robots can be reused). To this end, we have carried out several executions of this setup, increasing the cleaner team size. In every case, we compute the auction time as the sum of the time spent in each room (from the moment the coordinator finds a room until one cleaner is assigned to clean it) and, by performing five executions for each team size, we calculate the average auction time that characterizes the computational cost of this MRS.

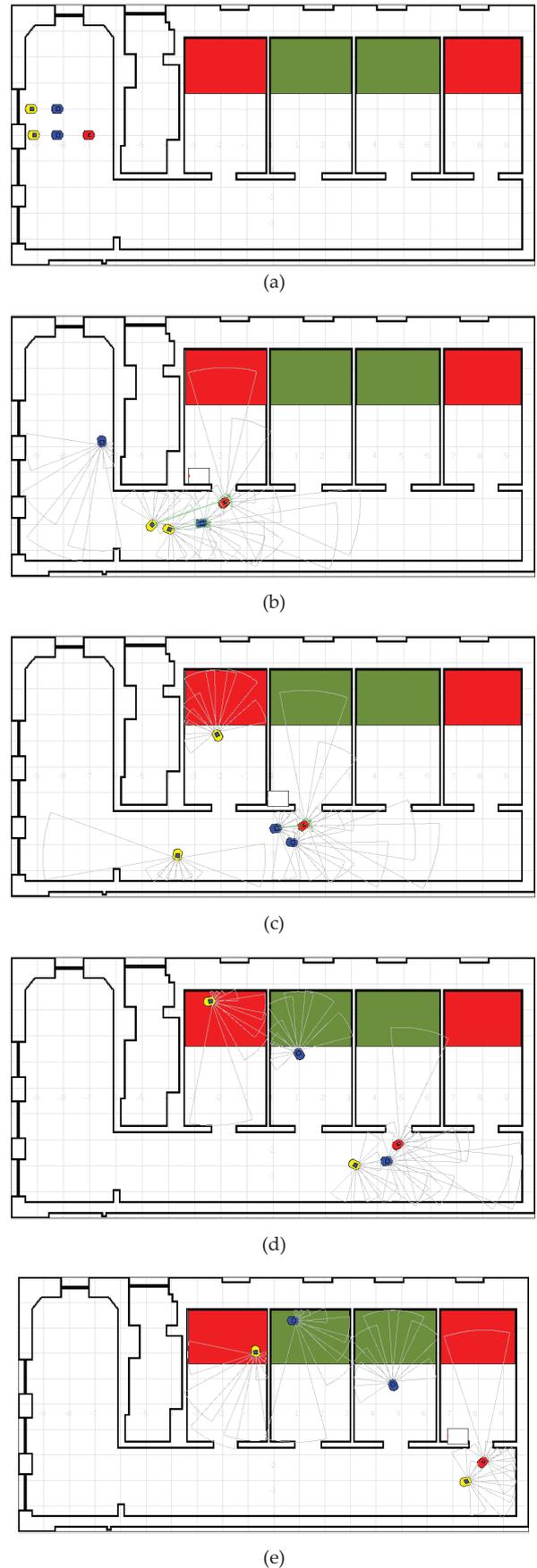
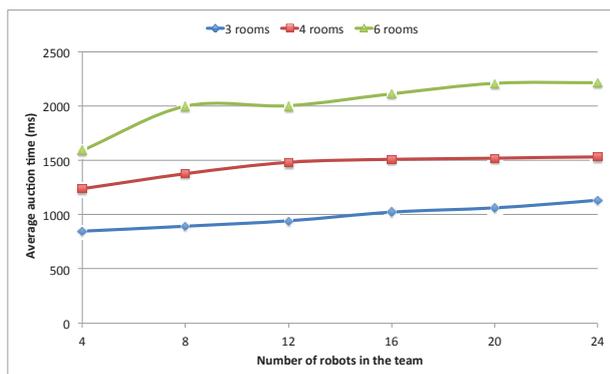


Figure 5. Five consecutive steps of execution of the MRS in the simple scenario with four rooms and four robots

In each execution, every robot starts in the same room, the coordinator robot starts searching for rooms in the environment and the cleaner team follows it. In the different screenshots of Fig. 5, we display a typical execution for a cleaner team of four robots. When the coordinator finds a room, it starts the auction (see Fig. 5b, Fig. 5c, Fig. 5d and Fig. 5e), providing information about the type of room (type of surface) and proposing a reward. In this case, the reward is a numerical value that implies a 50% of battery level increase for the robots, and it is the same for all the rooms. Each component of the team, according to its capabilities, bids to win this sub-task (it only bids if it is capable of solving the task). The auction follows the scheme described in section 2.2.

The bid, in this case, is simply proportional to the distance of the robot to the coordinator. This distance depends upon the *look for the coordinator* and *follow the coordinator* behaviours that operate using the sonar and laser sensors simulated in player/stage, so they are stochastic. Fig. 5e shows a typical final configuration of the MRS, with all the robots properly spread out over the rooms assigned according to their speciality.

We have repeated this execution with 8, 12, 16, 20 and 24 cleaner robots in the team. Moreover, we have run the experiment in two scenarios, one with three rooms and another with six rooms, to obtain more general conclusions. The results obtained are displayed in Fig. 6. As shown, when the number of robots increases, the average auction time increases slightly and in a linear manner independently of the number of rooms and robots. This computational scalability is a very important feature in real-world applications.



**Figure 6.** Average auction time (in milliseconds) for different team sizes and numbers of rooms in the scenario in the simple experiment

### 3.2 More complex task assignment

For the second example, we use the same simulated environment, but the complexity of the task assignment has been greatly increased. In this case, we have three types of rooms with varying degrees of dirt (represented using different shades of green in the screenshots of Fig. 8) and, in addition, we have included box-like objects that

the robots have to move in order to be able to clean the rooms (see Fig. 8). Now the MRS is made up of the following types of robots:

- **Coordinator:** this is exactly the same type of robot as in the previous example (represented in Fig. 8 in red).
- **Sweeper:** there are several robots of this type (represented in Fig. 8 in green), equipped with sonar and laser sensors to navigate and follow the coordinator and with a tool that is appropriate for sweeping the room.
- **Gripper:** several robots again (represented in Fig. 8 in yellow) equipped with sonar, laser sensors and with a gripper to grasp objects.
- **Camera:** several robots (represented in Fig. 8 in blue) equipped with sonar, laser and with a camera.

Again, the basic specs of the real problems we aim to confront in this work are present: specialized heterogeneous robots to solve the specialized tasks of the mission in dynamic conditions (mainly because the robots that finish a sub-task become available for other tasks).

Specifically, three types of rooms are now considered with a minimum number of robots required in each to accomplish the task:

1. Type 1 (light green): low dirt level, without boxes.
  - Only 1 sweeper is required
2. Type 2 (medium green): medium dirt level, 2 boxes
  - 1 sweeper, 1 gripper and 1 camera are required
3. Type 3 (dark green): high dirt level, 4 boxes
  - 2 sweeper, 2 gripper and 1 camera are required

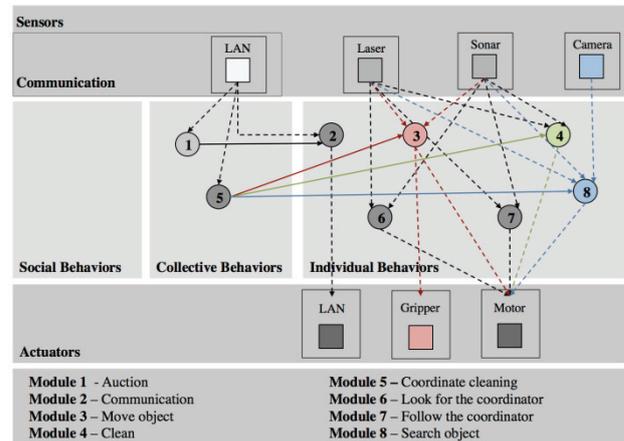
The coordinator detects the type of room through the camera and uses this information to ensure that the appropriate number of robots is assigned to each room. The dirt level implies a minimum number of sweeper robots and the presence of boxes defines whether gripper robots must also participate in the team. The gripper robots cannot see the boxes, so each time a room with boxes is found the gripper and camera robots must collaborate to clean it. In this way, we have set up a more complex assignment problem requiring a higher degree of coordination, where a single robot cannot clean most of the rooms, as in the previous case. Moreover, coordination is required between the robots that are in a room to clean it efficiently.

To solve this problem, the hierarchical auction strategy commented upon in section 2.2 has been implemented. Now, the single round auction strategy used by the coordinator in the previous experiment can be performed by any other member of the cleaning team. To do this, once the room is found by the coordinator it informs the others of its characteristics and the proposed reward (as in the previous example, the robots that successfully finish their

work receive a reward that increases their battery level by 50%). Now, each robot is able to start a sub-auction in order to form sub-teams to clean up the room. In this way, the individual control of the coordinator has the same complexity as in the previous experiment and only has to deal with the bids of the sub-auctioneers. These are now in charge of “presenting” a reliable sub-team proposal for cleaning the room. The following pseudo-code summarizes the auction algorithm implemented by the robots:

1. Receive the room type (dirt level, number of boxes) and the reward provided by the coordinator
2. IF (room is type 1)
  - 2.1. IF (I am sweeper)
    - 2.1.1. Bid with my current distance to the coordinator (laser sensor output) and battery level
  - 2.2. ELSE
    - 2.2.1. Bid with a battery level of 0
  - 2.3. END
3. ELSE (room is type 2 or 3)
  - 3.1. Compute the new reward (depending upon my battery level, with the aim of always being above 50%)
  - 3.2. Initiate a sub-auction process to set up a team (become the auctioneer)
  - 3.3. Inform the robots about the room type and the reward
  - 3.4. Receive information of other sub-auctions
  - 3.5. FOR EACH sub-auction
    - 3.5.1. IF (I am the auctioneer)
      - Wait for bids (max 15 ms)
      - IF (number of responses > minimum number of robots)
        - Select the robots with a higher bid (sum of distance to the coordinator and battery level)
        - Inform the selected robots
        - Send a bid to the coordinator with the team average distance and battery level
      - ELSE
        - My sub-auction is finished
      - END
    - 3.5.2. ELSE IF (I am a bidder)
      - IF (there is a sub-auction with a reward greater than that of my own team)
        - FOR EACH selected sub-auction
          - Send a bid to the sub-coordinator
          - Wait the for sub-coordinator response
        - IF (I am selected)
          - My sub-auction is finished
        - END
      - END FOR-EACH
    - 3.5.3. END
  - 3.6. END FOR-EACH
  - 3.7. Wait until the coordinator decides the winner of the auction
  - 3.8. IF (I am the winner)
    - 3.8.1. Clean the room with my sub-team
  - 3.9. ELSE
    - 3.9.1. Follow the coordinator until the next room
  - 3.10. END
4. END

The robots can only be in one sub-team when bidding to the coordinator. Thus, all the robots try to create their own auction by proposing a reward that depends upon their battery level, with the aim of always being above 50%. At the same time, they try to participate in other teams if the proposed reward is higher than that of their own sub-team. If they are accepted in another sub-team, their own team is terminated.



**Figure 7.** Behaviour modules for the sweeper, gripper and camera robots of the second experiment. Grey modules are common to the three types of robots; the red module corresponds only to the gripper robot; the blue one to the camera robot; and the green one to the sweeper robot.

Once the sub-teams are created, they all bid to the coordinator adding their average distances and battery levels, as explained in the previous pseudo-code. Other magnitudes could be considered here to make the bid, but we have decided to use an external measurement (distance) and an internal one (battery) so that other robots cannot infer their colleagues’ bids simply by their position. In this way, depending upon the specific auction behaviour of each robot, it can adopt a more aggressive or soft policy in the bids while using only partial information, which leads to a more interesting dynamics in the task assignment.

Once the sub-teams make their bid, the coordinator robot processes them and decides which one is assigned to the task. The following pseudo-code summarizes the auction algorithm implemented by the coordinator:

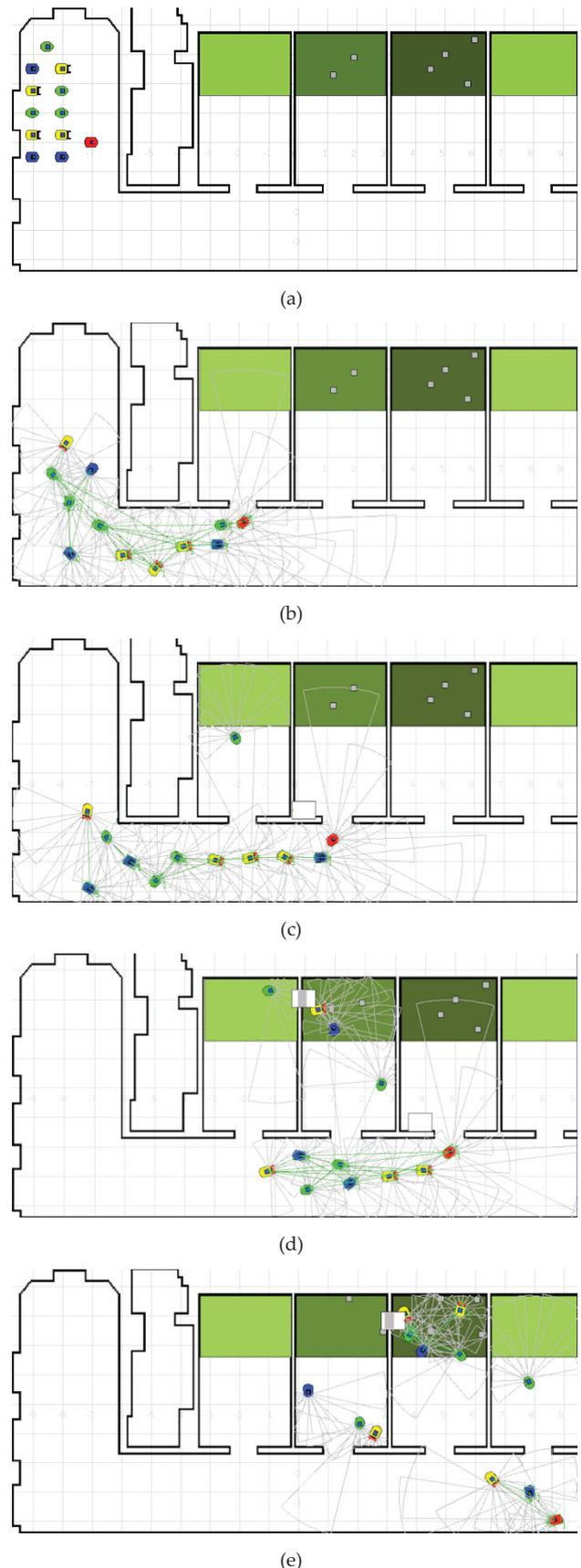
1. Inform the other robots that a room has been found and sends the room type (dirt level, number of boxes) and the reward
2. Wait for the sub-auctioneers bids
3. IF (there is a highest bid)
  - 3.1. Inform the winner
4. ELSE IF (there is a tie)
  - 4.1. IF (there is a sub-team with a lower number of robots)
    - 4.1.1. Inform the winner
  - 4.2. ELSE IF (there is a tie)
    - 4.2.1. Select the sub-team that started the sub-auction first
    - 4.2.2. Inform the winner
  - 4.3. END
5. END

The specific behaviour modules within each robot for this second experiment are as follows:

- The coordinator robot has the same behaviour modules as in the previous experiment: those displayed in Fig. 3.
- The sweeper, camera and gripper robots now share three individual behaviour modules: look for the coordinator, follow the coordinator and communications. These behaviours are displayed in Fig. 7 in grey. In addition, the sweeper robot has a 'clean' individual behaviour (the green module in Fig. 7), the gripper robot has a 'grip' individual behaviour (the red module in Fig. 7) and the camera robot has a 'search for object' individual behaviour (the blue module in Fig. 7). Furthermore, they all have two collective behaviours. The first one is, in this case, an auction behaviour – the same as that of the coordinator robot - in order to carry out the sub-auction if required (see the module with number 1 in Fig. 7) This behaviour module is a decision module that controls the communications' low-level behaviour. The second collective behaviour is necessary to coordinate cleaning in a room once the sub-auction has finished (for example, when the room is not too dirty and a single sweeper can clean it or when it has boxes). This decision module affects a different individual behaviour module for each type of robot, as displayed in Fig. 7 with arrows of different colours. In summary, in this second example, all the robots except for the coordinator have 4 individual behaviours and two collective ones.

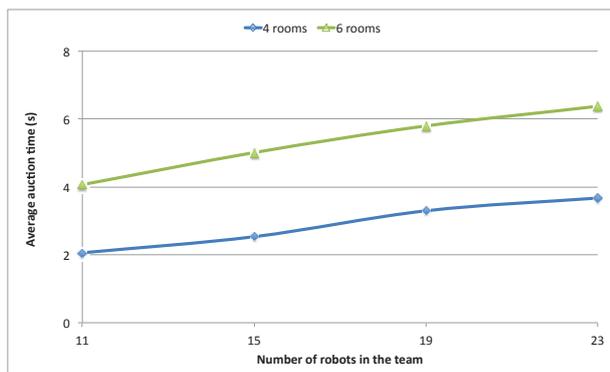
A key point in this experiment is that the *auction* behaviour implemented for the coordinator robot in the first experiment is exactly the same that the sweeper, camera and gripper robots have now. This decision goes in the line of implementing a complex assignment strategy that is as simple as possible - in this case as a hierarchical replication of auction behaviours. With this approach, one might think that the computational cost would be compromised as the team size increases, and this is what we study from this point forward.

Fig. 8 contains five screenshots of a typical simulation in the case of having 3 camera robots, 4 gripper robots and 4 sweepers (Fig. 8a) in a scenario with four rooms. The first room has a low dirt level and one sweeper won the auction (Fig. 8c). The second room has a medium dirt level with two boxes. The team that won the auction had 3 robots - one of each type (the minimum to clean the room in this case), as can be seen in Fig. 8d. The third room is the dirtiest and has 4 boxes; therefore, five robots were necessary to clean it properly (see Fig. 8e). Finally, the last room is like the first one and needs a single sweeper. In this case, the coordinator had to wait until a sweeper robot assigned to the other room finished its work to end the auction. As displayed in Fig. 8e, this was the sweeper of the first room.



**Figure 8.** Five consecutive steps of the execution of the MRS in the complex scenario

As in the previous experiment, we have computed the auction time as the average of the time spent on each room (from the moment the coordinator finds a room until one sub-team is assigned to clean it) and we calculate the average auction time in five independent runs. This experiment has been carried out using two scenarios, one with four rooms and another with six, created with two rooms of each type. Fig. 9 displays the results obtained in these two cases.



**Figure 9.** Average auction time (in seconds) for different team sizes and numbers of rooms in the scenario in the complex experiment

As we can see from these curves, the average auction time is much higher than in the previous experiment, due to the time spent for all the robots that try to create their own sub-auctions in parallel. The time magnitude is not the point we want to highlight from Fig. 9 but rather the fact that time increases linearly with the number of robots in both cases. This is very interesting, if we take into account the high complexity of the task/robot organization, which can lead to situations where the auction has to wait until a specific type of robot is available. The main reason for this is the simplicity of the single auction strategy that has been employed, which has allowed the implementing of a complex task assignment strategy by hierarchically replicating such a simple algorithm in the architecture.

## 5. Discussion and conclusions

After presenting the results obtained in the experiments, several conclusions can be extracted. First, the extremely simple auction-based strategy described in section 2.2 has provided successful results in terms of assigning tasks appropriately in simple and complex cases. Second, the computational implementation of the strategy, summarized in Fig. 2, has been exactly the same in both experiments. That is, we did not have to design a new version to deal with more complex assignments and we simply had to replicate the basic algorithm in a hierarchical way. Third, the computational time increases linearly with the team size in the worst case, both in simple and complex cases. Consequently, and as the main conclusion of this work, we must highlight the scalability

of simple auction-based strategies in task assignment problems of increasing complexity. This idea is the opposite of that of developing a more sophisticated and complex task assignment strategy where the scalability in computational cost and designer effort is clearly compromised. This is very relevant for real-world multi-robot applications where the types of robots and tasks that must be accomplished are very different.

## 6. Acknowledgments

This work was partially funded by the Xunta de Galicia and the European Regional Development Funds through projects 09DPI012166PR and 10DPI005CT.

## 5. References

- [1] Farinelli, A., Locchi, L., Nardi, D. (2004). Multirobot Systems: A Classification Focused on Coordination, *IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics*, Vol. 34, No. 5, pp. 2015-2028, IEEE Press
- [2] Sahin, E., Spears, W. (2005). *Swarm Robotics*, SAB 2004 International Workshop, Santa Monica, CA, USA, July 17, 2004, Revised Selected Papers, LNCS 3342, Springer
- [3] Mataric, M., Gerkey, B. (2004). A Formal Analysis and Taxonomy of Task Allocation in Multi-Robot Systems, *The International Journal of Robotics Research*, Vol. 23, No. 9, pp. 939-954, SAGE
- [4] Vail, D., Veloso, M. (2003). Multi-Robot Dynamic Role Assignment and Coordination Through Shared Potential Fields, *Multi-Robot Systems*, Kluwer
- [5] Tsalatsanis, A., Yalcin, A. and Valavanis, K. P. (2010). Dynamic Task Allocation in Cooperative Robot Teams, *International Journal of Advanced Robotic Systems*, V. Kordic, A. Lazinica and M. Merdan (Eds.), InTech
- [6] Liu, S., Sun, T., Hung, C. C. (2011). Multi-Robot Task Allocation Based on Swarm Intelligence, *Multi-Robot Systems, Trends and Development*, T. Yasuda (Ed.), InTech
- [7] Dahl, T. S., Mataric, M. and Sukhatme, G. S. (2009). Multi-robot task allocation through vacancy chain scheduling, *Robotics and Autonomous Systems Vol 57, Issues 6-7*, pp. 674-687, Elsevier
- [8] Jain, L., Aidman, E., Abeynayake, C. and Dasgupta, P. (2011). Multi-Robot Task Allocation for Performing Cooperative Foraging Tasks in an Initially Unknown Environment, *Innovations in Defense Support Systems 2*, Vol. 338, pp. 5-20, Springer Berlin Heidelberg
- [9] Michael, N., Zavlanos, M., Kumar, V. and Pappas, G. (2008). Distributed Multi-Robot Task Assignment and Formation Control, *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 128-133, IEEE Press

- [10] Thomas, G. and Williams, B. (2009). Sequential auctions for heterogeneous task allocation in multiagent routing domains, In Proceedings of the 2009 IEEE international conference on Systems, Man and Cybernetics (SMC'09), pp. 1995-2000, IEEE Press
- [11] Oliver, G. and Guerrero, J. (2011). Auction and Swarm Multi-Robot Task Allocation Algorithms in Real Time Scenarios, Multi-Robot Systems, Trends and Development, T. Yasuda (Ed.), InTech
- [12] Gerkey, B. and Mataric, M. (2002) Sold!: Auction Methods for Multirobot Coordination, IEEE Transactions on Robotics and Automation, Vol. 18, No. 5, IEEE Press
- [13] Dias, M. B., Zlot, R., Kalra, N. and Stentz, A. (2006) Market-Based Multirobot Coordination: A Survey and Analysis, Proceedings of the IEEE, Vol. 94, Issue 7, pp. 1257-1270, IEEE Press
- [14] Zlot, R. and Stentz, A. (2006), Market-based multirobot coordination for complex tasks, International Journal of Robotics Research, Vol. 25, Issue 1, pp. 73-101, SAGE
- [15] Goerick, C. (2011) Towards an Understanding of Hierarchical Architectures. IEEE Transactions on Autonomous Mental Development, Vol. 3, No. 1, pp. 54-63, IEEE Press
- [16] Nicolescu, M. and Mataric, M. (2002). A Hierarchical Architecture for Behavior-Based Robots, In Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems, pp. 227-233, ACM
- [17] Becerra, J. A., Bellas, F., Santos, J. and Duro, R. J. (2005). Complex Behaviours through modulation in Autonomous Robot Control, Lecture Notes in Computer Science, Vol. 3512, pp. 717-723 Springer-Verlag
- [18] Caamaño, P., Becerra, J. A., Bellas, F., Prieto, A. and Duro, R. J. (2011). Evolutionary Procedure for the Progressive Design of Controllers for Collective Behaviors. HAIS 2011, Lecture Notes in Computer Science, 2011, Vol. 6679/2011, pp. 471-478, Springer
- [19] Collett, T., MacDonald, B. and Gerkey, B. (2005). Player 2.0: Toward a Practical Robot Programming Framework, In Proceedings of the Australasian Conference on Robotics and Automation (ACRA 2005), Sydney, Australia