Workshop Notes of AAAI-06 Workshop on
Auction Mechanisms for Robot Coordination, 2006.

Layering Coalition Formation With Task Allocation

Fang Tang and Lynne E. Parker
Distributed Intelligence Laboratory, Department of Computer Science
The University of Tennessee, Knoxville, TN, 37996-3450
Email: {*ftang*|*parker*}@*cs.utk.edu*

Abstract

This paper presents an approach for layering lower-level coalition formation with higher-level, traditional task allocation. At the lower level, coalitions to solve multi-robot tasks are formed using our ASyMTRe approach that maps environmental sensors and perceptual and motor schemas to the required flow of information in the robot team, automatically reconfiguring the connections of schemas within and across robots to form efficient solutions. At the higher level, a traditional task allocation approach is used to enable individual robots and/or coalitions to compete for task assignments through time-extended task allocation. We present a motivating example of site clearing and formalize the problem. We then present the proposed approach of layering ASyMTRe with task-allocation. As this is still a work in progress, we outline planned experiments we intend to develop to validate our approach.

Introduction

Traditionally, task allocation approaches in multi-robot teams have dealt with the assignment of *single-robot* tasks, which are tasks (or collections of tasks or subtasks) that can be accomplished independently by a single robot. Another important type of task in multi-robot teams is the *multi-robot* task. Typically, a multi-robot task requires a *strongly cooperative* solution (Brown & Jennings 1995), meaning that the task is not trivially serializable, so that it cannot be decomposed into subtasks that can completed by individual robots operating independently; instead, it requires robots to act in concert to achieve the task. Sometimes, this type of task is also called *tightly-coupled* or *tightly-coordinated*. Robots that join together to solve this type of multi-robot task are referred to as *coalitions* by some researchers (Gerkey & Mataric 2004). In this paper, we also form coalitions for accomplishing these strongly cooperative multi-robot tasks. Even though we are not using the traditional definition of coalition by calculating payoffs as in game theory (Luce & Raiffa 1957), we share the same motivation behind coalition formation as mentioned in (Shehory & Kraus 1995); that is, robots in a coalition should work together to share resources and cooperate on task execution due to their decision that they would benefit more from working together as a coalition than they would working individually.

Many researchers have addressed the allocation of single-robot tasks (Parker 1998; Werger & Mataric 2000; Botelho & Alami 1999; Gerkey & Mataric 2002; Dias 2004; Zlot & Stentz 2006). Some recent work also addresses the allocation of multi-robot tasks (Jones *et al.* 2006; Kalra, Ferguson, & Stentz 2005; Lin & Zheng 2005). Our approach is different in that we are addressing the multi-robot tasks through the dynamic configuration of low-level behavioral building blocks instead of predefined plans or roles. A few researchers have addressed the formation of coalitions for *multi-robot tasks* (Parker & Tang 2006; Vig & Adams 2005). However, approaches are lacking that combine these techniques into a single system. The objective of this paper is therefore to define an approach that enables the allocation of both types of tasks into a single framework. Our approach layers the ASyMTRe coalition-formation system that we have previously developed (Parker & Tang 2006) with an auction-based mechanism for achieving the allocation of single-robot and/or independent subtasks. Our proposed approach enables robots to form coalitions at the lower level to solve a single multi-robot task (with a strongly cooperative solution). Coalitions, and possibly individual robots, then compete for tasks (or collections of tasks) at the higher level, using the more traditional task allocator.

In the remainder of this paper, we first provide additional background on our approach and its relationship to related work. We then describe an application example to motivate this work. We formalize the problem and outline our proposed approach. Since this work is preliminary, we outline planned experiments to validate our approach. We conclude with a summary and a description of our future work.

Background and Related Work

The *task allocation* problem is the problem of determining a suitable mapping between robots and tasks. The majority of work in task allocation for multi-robot systems (Parker 1998; Werger & Mataric 2000; Botelho & Alami 1999; Gerkey & Mataric 2002; Dias 2004; Zlot & Stentz 2006) focuses on allocating *single-robot tasks* to *single-task robots* with either *instantaneous assignment* or *time-extended assignment* (using the taxonometric terms of

(Gerkey & Mataric 2004). Typically, a task is decomposed into independent subtasks (Parker 1998), hierarchical task trees (Zlot & Stentz 2006), or roles (Simmons *et al.* 2000) either by a general autonomous planner or by the human designer. Independent subtasks or roles can be achieved concurrently, while subtasks in task trees are achieved in order according to their precedence constraints. The work of (Zlot & Stentz 2006) also addresses "tightly-coupled" multi-robot tasks, however, their task can be decomposed into multiple single-robot tasks and thus is different from our approach. A formal analysis comparing the computation, communication requirements and solution qualities of several well-known approaches is presented in (Gerkey & Mataric 2004).

Some recent work in task allocation (Jones *et al.* 2006; Kalra, Ferguson, & Stentz 2005; Lin & Zheng 2005) begin to address multi-robot task allocation, where team members need to tightly cooperate with each other to accomplish the task. The Hoplites approach (Kalra, Ferguson, & Stentz 2005) focuses on the selection of an appropriate joint plan for the team to execute by incorporating joint revenue and cost in the bid. The work in (Jones *et al.* 2006) achieves multi-robot task allocation through matching roles with robot capabilities. The work in (Lin & Zheng 2005) also matches task required capabilities with robot capabilities and accomplishes multi-robot tasks through combinatorial bids. Our approach of task allocation on the higher level is similar to the above approaches, but is different in the way that coalitions (subgroups) are formed to accomplish a single multi-robot task. Our approach forms a coalition through configuring the sensors and preprogrammed schemas on every team member so that they share sensory or computational information with each other in order to accomplish the task. We are generating the coalitions "on the fly" instead of using predefined plans, roles, etc.

Role-based approaches, such as the work of (Simmons *et al.* 2000), also assume a pre-defined coordination among robots, according to their roles. In contrast, our ASyMTRe approach can automatically configure new coalition strategies (such as which combination of sensors and low-level schemas to activate, based on the available robots), without pre-defining how the robots will interact.

Multi-robot coalition formation for multi-robot tasks deals with the issue of how to organize robots into subgroups to accomplish multi-robot tasks, using a strongly cooperative solution approach. The motivation behind coalition formation for multi-robot tasks is to enable team members to work together as a group to accomplish tasks that cannot be handled by individual robots working independently (i.e., tasks that are not trivially serializable, as defined by Brown and Jennings (Brown & Jennings 1995)). Since robots have different sensor, effector and computational capabilities, a team of resource-bounded robots may not individually possess all of the required capabilities to accomplish a task. However, they could work with other robots as a coalition to effectively accomplish the task objectives.

In the area of coalition formation, we are particularly interested in flexible techniques for automating the formation of coalitions to solve a multi-robot task, which may involve the sharing of sensory, perceptual, and computational capabilities across heterogeneous team members. In our recent work (Parker & Tang 2006), we have developed the ASyMTRe approach to address this issue of coalition formation. However, although ASyMTRe provides a way of generating robot coalitions, it can only handle a single multi-robot task at a time. For missions of multiple tasks, we would like to achieve task allocation amongst coalitions and/or individual robots, thus combining the benefits of low-level coalition formation with those of higher-level, more traditional, task allocation. Our idea in this paper is to layer ASyMTRe for low-level coalition formation (for solving a single multi-robot task), with a higher level, traditional task allocator (for solving a set of tasks). We believe the resulting approach would be a flexible mechanism for a broad range of realistic multi-robot applications, with the ability to generate both strongly cooperative and weakly cooperative solution strategies, as appropriate.

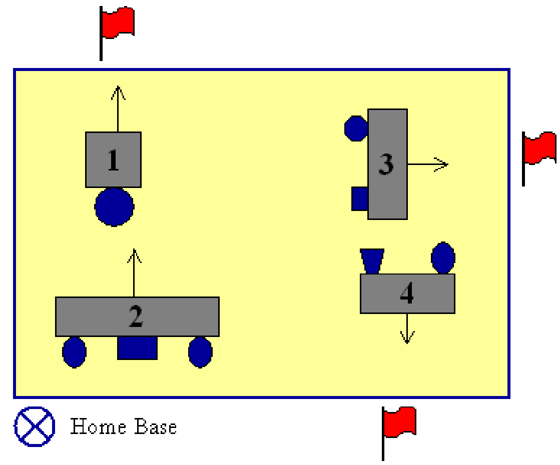

Figure 1: The site clearing application. Red flags represent collection points. Different shapes represent the heterogeneity of the robots.

Motivating Example: The Site Clearing Application

To motivate the need for the combination of coalitions and more traditional task allocators, we define a representative application, called the *site clearing* application. The site clearing application is a simplified version of the site preparation task (Parker *et al.* 2000), which has been identified by NASA as an important prerequisite for human missions to Mars. The site clearing application, illustrated in Figure 1, requires a specific area to be cleared of obstacles, which we simplify to be boxes with different weights or sizes. The objective of the application is to clear the site in as little time as possible while minimizing the cost to the robots (e.g., energy consumption or computational requirements). For the purposes of this discussion, we assume that a map is available to enable the robot team to determine the positions of the obstacles in the area. We assume that the obstacles to be removed from the site can either be pushed outside the area, or can be pushed to a common collection point, as indicated

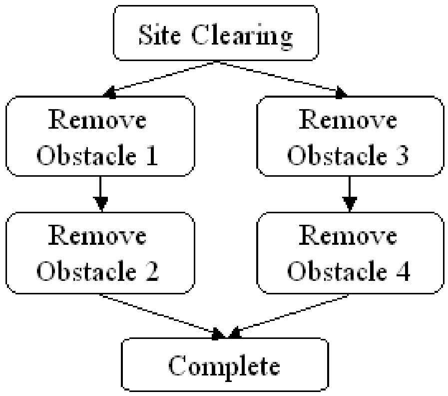

Figure 2: A partial-order plan for the site clearing application.

by a beacon. We further assume that a partial-order planner exists to determine the ordering constraints of removing the obstacles, in case certain obstacles need to be removed before other other obstacles can be cleared.

We define a number of constraints to make the site clearing problem challenging, such as:

- *Limited team size*: The number of obstacles is greater than the number of robots, thus requiring robots to iteratively move obstacles for several rounds to clear the site.
- *Varying weights/sizes of obstacles*: Robots have different weight/size requirements for the kind of obstacle they can manipulate. Thus, the weight or size of an obstacle determines the number of robots required to transport it. Robots working on the same obstacle need to cooperate with each other.
- *Heterogeneous robots*: Robots may differ in their capabilities, thus requiring the allocation approach to appropriately map robots to tasks.
- *Resource-bounded robots*: Robot team members may be resource-bounded, and thus unable to transport an obstacle independently, or navigate in the site independently. Robot coalitions may therefore be needed to share sensory, perceptual, computational, or effector resources to enable the team as a whole to accomplish the required task(s). Although sometimes these interactions can be trivially serializable (e.g., as in the box pushing example of (Parker 1994)), in the general case of resource-bounded robots, they cannot. Thus, this constraint illustrates the need for expanding current task allocation approaches to include coalition formation for multi-robot tasks.
- *Uncertainty*: the uncertainty of the environment and robot team capabilities (due to sensor or robot failures) requires that team solutions should be based on current team capabilities instead of predefined solutions.

The site clearing application can be decomposed into a series of tasks with ordering constraints. Each task is aimed at removing one obstacle from the site, which we call "Remove Obstacle". For example, the task shown in Figure 1 can be accomplished through the partial-order plan in Figure 2. Since only some tasks have ordering constraints, the system can allocate a subset of the tasks to the robots for concurrent execution. Thus, when making a task allocation decision, robots are considering more than one task at a time. In addition, because of the application challenges mentioned earlier, a "Remove Obstacle" task may require multiple robots to form a coalition to accomplish the task in a manner that efficiently uses the available robot capabilities. Additionally, when multiple coalitions are available, the system must determine which coalition is the best fit to the current task.

Note that from our perspective, an individual task (such as those defined in Figure 2) cannot be categorized in advance as a multi-robot task or a single-robot task. Instead, whether or not the task requires single or multiple robots depends upon the capabilities of the robot team members. Some robots may be able to perform a given task on their own (thus making the task a single-robot task), while other robots may require help from teammates to accomplish that same task (thus making that same task a multi-robot task). Our ASyMTRe approach is able to find combinations of robot capabilities that can accomplish the task in either the single-robot case or the multi-robot case, depending upon the team capabilities.

Formalism of the Problem

The multi-robot task we address can be formally defined as follows:

- $R = \{R_1, R_2, \cdots, R_n\}$ is a collection of *n* robots, where each robot R_i is represented by its available environmental sensors (ES), and its corresponding perceptual (PS), motor (MS), and communication schemas (CS). For a complete definition of R, please refer to (Tang & Parker 2005a).
- T is the team-level task to be accomplished, which is denoted as $T = \{t_1, t_2, t_3, \cdots\}$.
  - A set of *ordering constraints* defines a proper partial order of tasks. $t_i \prec t_j$ means that task t_i must be executed sometime before task t_j.
  - A set of *open preconditions*. A precondition is open if it is not achieved by some task in the plan.
  - A subset T^i of T can be allocated to robots concurrently if the tasks in T^i do not have ordering constraints and their preconditions are not open.
  - Each task t_i is further defined as a set of motor schemas that need to be activated in certain ways in order to accomplish this task.
- To accomplish a subset of tasks T^i, a collection of *m* coalitions, denoted $C^i = \{C_1^i, C_2^i, \cdots, C_m^i\}$, needs to be generated based on the task requirements of T^i and the robot capabilities (Tang & Parker 2005b).
- With multiple solutions available, we define a *cost* function for each robot, specifying the cost of the robot performing a given task, and then estimate the cost of a coalition performing the given task. We consider two types of cost:

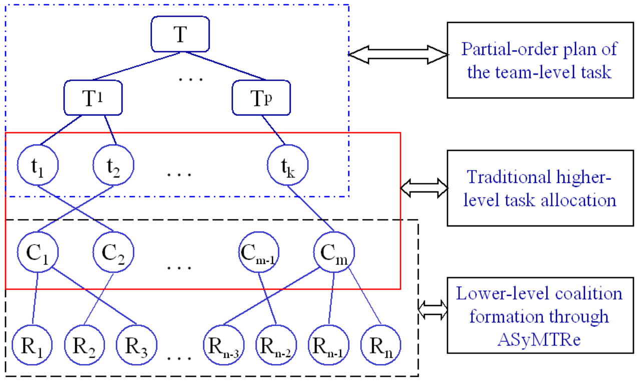

Figure 3: The relationships between tasks, coalitions and robots.

- A robot-inherent cost measures the inherent cost (e.g., in terms of energy consumption or computational requirements) of using particular capabilities on the robot (such as a laser or a mapping algorithm). We denote robot R_i's inherent cost by $robot\_cost(R_i)$.
- A task-specific cost measures cost according to task-related metrics, such as time, distance, success probability, etc. We denote the cost of R_i performing task t_j by $task\_cost(R_i, t_j)$.
- The *cost* function of R_i performing t_j is represented by $cost(R_i,\ t_j)$, which is a weighted combination of both the robot-inherent cost and task-specific cost, normally in the form of a linear function. Other type of costs can also be easily incorporated when necessary.
- The cost of a coalition C_i performing a task t_j is the sum of individual cost of robots that are in the coalition, which is denoted as:

$$cost(C_i, t_j) = \sum_{R_k \in C_i} cost(R_k, t_j) \qquad (1)$$

The problem we address here is: Given (T, R), assign a set of tasks T^i to coalitions of R such that the sum of the coalition costs $\sum_{t_k \in T^i} cost(C^i_j, t_k)$ are minimized.

The Approach: Layering Coalition Formation with Task Allocation

To allocate multi-robot tasks to a team of robots, we propose an approach encompassing four main steps as shown in Table 1. Figure 3 describes a general procedure that first decomposes a team-level task to a set of tasks with ordering constraints. At the lower level, coalitions from the team of robots are formed to address the given tasks. These coalitions are not distinct, but may share same team members. The coalitions then compete for the assignment of tasks using a traditional task allocation approach.

Lower-Level Coalition Formation

We now describe the lowest level of this layered approach – the coalition formation strategy, based on ASyMTRe (which stands for Automated Synthesis of Multi-robot Task solutions through software Reconfiguration, pronounced "Asymmetry"). The ASyMTRe approach (Tang & Parker 2005a; 2005b; Parker & Tang 2006) has been developed for addressing the formation of heterogeneous robot coalitions that solve a single multi-robot task. More generally, this approach deals with the issue of how to organize robots into subgroups into a strongly cooperative solution that accomplishes a task collaboratively based upon their individual capabilities.

The fundamental idea of ASyMTRe is to change the abstraction that is used to represent robot competences from the typical "task" abstraction to a biologically-inspired "schema" abstraction and providing a mechanism for the automatic reconfiguration of these schemas to address the multi-robot task at hand. To achieve this, we view robot capabilities as a set of environmental sensors that are available for the robot to use, as well as a set of perceptual schemas, motor schemas, and communication schemas that are preprogrammed into the robot at design time.

The ASyMTRe approach extends prior work on schema theory (Arkin 1987; Lyons & Arbib 1989) by autonomously connecting schemas at run time instead of using pre-defined connections. According to information invariants theory (Donald 1995), the information needed to activate a certain schema or to accomplish a task remains the same regardless of the way that the robot may obtain or generate it. We can therefore label inputs and outputs of all schemas with a set of information types, such as *laser range data*, *self global position*, etc. Two schemas can be connected if their input and output information labels match. Thus, schemas can be autonomously connected within or across robots based upon the flow of information required to accomplish a task. With the run time connection capabilities, task solutions can

Table 1: Allocating Multi-Robot Tasks to a Team of Robots

Input: (T, R)

1. Find the set of tasks T^i up to a constant number[a], such that both the ordering constraints and the preconditions of tasks are satisfied.
2. Configure solutions for each task t_j in T^i by forming a set of coalitions C^i, based on t_j's objective and the current team capabilities.
3. Allocate tasks in T^i to coalitions in C^i, such that:
   - The task-specific cost and the robot-inherent cost are minimized for the set of tasks.
   - A coalition can win at most one task at a time. Assuming $C' \subseteq C^i$ is the set of coalitions selected to perform the tasks in T^i, then the following condition must be satisfied: $\forall_{C'_i, C'_j \in C', i \neq j}, C'_i \cap C'_j = \emptyset$.
4. Monitor the execution of tasks. If the entire task is not complete, start the allocation process (go to step 1) when robots are within Δt time to complete their current tasks. Otherwise, exit.

[a] Note that the maximum number of tasks allowed for allocation is limited to a constant number a to decrease the computational complexity of the allocation of multiple tasks at once.

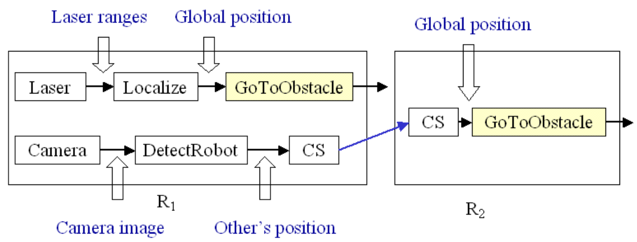

Figure 4: An example of "GoToObstacle" task. The connections between schemas are dynamically generated through ASyMTRe.

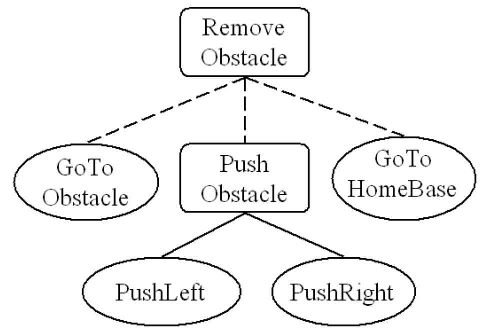

Figure 5: An example task tree for the Remove Obstacle task.

be configured in many ways to solve the same task or can be reconfigured to solve a new task. Additionally, robots can share information to assist each other in accomplishing a task. As an example, two robots required to remove an obstacle in the site clearing task must first navigate to where the obstacle is located. If we assume a robot R_2 is unable to determine its own position, then another robot could help it by providing localization information (see (Parker & Tang 2006) for many more details on these capabilities). Figure 4 gives an example of the schema connections on two robots for the "GoToObstacle" task, where robot R_1 provides *position information* to R_2 to guide its navigation.

We have implemented the ASyMTRe approach using a distributed negotiation protocol (Tang & Parker 2005b) inspired by the Contract Net Protocol (Smith 1980). We validated this approach through simulation and physical experiments and analyzed its performance in terms of robustness, scalability, and solution quality. These experimental results allowed us to conclude that the ASyMTRe approach provides beneficial mechanisms for multiple robots to: (1) synthesize task solutions using different combinations of robot sensors and effectors, (2) share information across distributed robots and form coalitions as needed to assist each other in accomplishing the task, and (3) reconfigure new task solutions to accommodate changes in team composition and task specification, or to compensate for faults during task execution. Thus, the ASyMTRe approach can serve as the lower-level solution generator in our approach.

Task Trees

Previously, we defined a task in ASyMTRe as a set of motor schemas that need to be activated to accomplish this task (Tang & Parker 2005a). Multiple motor schemas are related through AND and OR logical operator. However, these relationships are not rich enough for multiple tasks, since, for some applications, two motor schemas may need to be executed one after another. Therefore, to better characterize the relationships between motor schemas, we plan to use *task trees* to represent tasks, similar to the tree generated by TDL (Simmons & Apfelbaum 1998). The root of the task tree is the most abstract task description. Each successive level of the tree represents a refinement of the tasks in the immediate upper level. The tree will be refined until all the leaf nodes can be represented by motor schemas that are preprogrammed on the robots. The task tree embeds parent/child relationships and synchronization constraints between nodes, including: *sequential*, meaning that the tasks associated with the nodes need to be executed in a sequential order (such as from the leftmost child node to the rightmost child node); and *concurrent*, meaning that the tasks associated with the nodes can be executed at the same time, or roughly the same time. An example task tree for the "Remove Obstacle" task is shown in Figure 5, which involves the following sequential tasks: (1) navigating to the obstacle, (2) pushing the obstacle to the goal, and (3) navigating to the closest home base and waiting for new tasks. Given the task tree, each robot can then use the distributed ASyMTRe negotiation protocol to decide how to form coalitions to accomplish a task, while maintaining the synchronization constraints during task execution.

Higher-Level Task Allocation through Auction

Although ASyMTRe provides the mechanism for a heterogeneous robot team to accomplish a task by forming coalitions, it can only handle one multi-robot task at a time. We therefore propose the use of an auction mechanism to provide a higher-level task allocation approach on top of ASyMTRe for handling multiple tasks. Note that the intent of this approach is not to develop a new auction mechanism, but instead to layer existing auction mechanisms with the ASyMTRe approach for allocating multi-robot tasks to robot coalitions. The following higher-level auction process is similar to (Jones *et al.* 2006), although the techniques for coalition formation is different. Additionally, we allow the allocation of multiple tasks at a time instead of one.

The auction process is described as follows:

1. *Task announcement*: Initially, the human operator introduces the site clearing task T to the system. Each task t_i in T is embedded with task-specific information, such as the size and the position of the obstacle to be removed. The human operator has an interface "Auctioneer" that interacts with the other robots in the system (similar to OpTrader in (Dias 2004)). This auctioneer holds the partial-order plan for T, selects a subset of tasks T^i that satisfies the ordering constraints and the preconditions, and makes an auction call of T^i to all robots.
2. *Coalition formation*: Robots that receive T^i start negoti-

ating with others to generate solutions for accomplishing tasks in T^i. For each task t_j in T^i:

(a) Each robot tries to find a list of coalitions (up to a constant number b) that it can join to accomplish t_j. The revised ASyMTRe negotiation protocol returns the top b coalitions per task. The size of a coalition is limited to a max coalition size c assuming robots work in a non-super-additive environment (Shehory 1998)[1].

(b) Coalitions are not arbitrarily formed, but are selected based on the combination of the robot-inherent cost and the task-specific cost (please refer to *Formalism of the Problem* Section for details of cost estimation.).

3. *Bid submission*: Once coalitions are formed for each task t_j, a randomly selected coalition leader submits a bid to the auctioneer, including information such as the list of coalition members, the cost of this coalition performing t_j, the leader of the coalition, etc.

4. *Winner determination*: Once bids for all tasks in T^i are collected or a timeout has expired, the auctioneer then determines the winner coalition for each task. The goal for the auctioneer is to find a coalition C_j for each task t_j, such that the total cost of performing the tasks in T^i is minimized and there is no overlapping of coalition members assigned to the tasks. If no such coalition C_j exists for task t_j and C_k for t_k such that $C_j \cap C_k \neq \emptyset$, then one of the tasks (either t_j or t_k) is auctioned again in the next round. The problem of determining the winner is equivalent to the combinatorial auction where multiple tasks are offered and each coalition can bid a subset of tasks. Existing combinatorial auction clearing algorithms (such as (Sandholm *et al.* 2005)) can be applied here with a constraint that the assigned coalitions do not overlap for different tasks.

5. *Award acceptance*: Once winner coalitions are determined, the auctioneer awards each task to the leader of the selected coalition. The leader robot then contacts the other coalition members to get ready for the task. Once responses from other coalition members are received, the leader robot accepts the award by sending a task acceptance message to the auctioneer and the coalition members commit themselves to the task until the task is complete. Otherwise, the award is rejected and the task needs to be auctioned again.

Experiments

As this work is still preliminary, we have not yet conducted an implementation or experiments to validate our proposed approach. This section outlines our plans for this validation. We will demonstrate the site clearing task both in simulation and on a physical robot team. The physical robot team will be composed of three Pioneer robots. The possible sensors on the robots are: laser scanner, sonar, and camera. In this task scenario, the team needs to remove several boxes with different sizes from a specific area and return to the home base.

The underlying box pushing protocols we will use are based upon the protocols developed in (Donald 1995; Parker 1998). To date, we have implemented the following main perceptual and motor schemas that are essential to the application:

- A robot with a laser range scanner can calculate the orientation of the box relative to itself.
- A robot with a laser range scanner or a ring of sonars can determine whether the robot needs to align with the box so that it will not lose control of the box.
- A robot with a camera can perceive the red marker representing the collection points.
- Motor schemas are programmed enabling robots to push the box towards a goal position.

We will vary the robot team capabilities, and the number and sizes of boxes to collect data on the task completion time and the solution quality, and first compare them with the *single-task* auction.

Conclusion and Future Work

We have described preliminary plans for building multi-robot coalitions to perform multi-robot tasks. The lower-level ASyMTRe approach automatically forms coalitions according to the task objective. The higher-level auction-based task allocation provides the mechanism for the team to allocate sets of tasks, holding auctions to assign tasks to the best-fitting individual robots or coalitions.

Our ongoing work includes developing the higher-level auction-based approach that enables a set of multi-robot tasks to be allocated simultaneously, incorporating the higher-level task allocation with the lower-level coalition formation, and performing experiments to validate our approach.

We also believe that the ASyMTRe approach for coalition formation can be merged with other, non-auction-based approaches to task allocation, such as the motivation-based approach of ALLIANCE (Parker 1998). We believe it would be interesting to investigate the combination of ASyMTRe and ALLIANCE, as an alternative approach for achieving the merging of coalitions for multi-robot tasks with traditional task allocation techniques.

References

Arkin, R. C. 1987. Motor schema based navigation for a mobile robot: an approach to programming by behavior. In *Proceedings of the IEEE Conference on Robotics and Automation*, 264–271.

Botelho, S., and Alami, R. 1999. M+: A scheme for multi-robot cooperation through negotiated task allocation and achievement. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 1234–1239.

[1] Due to the similarity between our configuration algorithm and the coalition formation algorithm presented in (Shehory 1998), we plan to analyze the bounds on our solution quality in future work. It has been proved in (Shehory 1998) that similar algorithms are of low logarithmic ratio bounds to the optimal solution.

Brown, R. G., and Jennings, J. S. 1995. A pusher/steerer model for strongly cooperative mobile robot manipulation. In *Proceedings of 1995 IEEE International Conference on Intelligent Robots and Systems (IROS '95)*, volume 3, 562–568.

Dias, M. B. 2004. *TraderBots: A New Paradigm for Robust and Efficient Multirobot Coordination in Dynamic Environments*. Ph.D. Dissertation, Robotics Institute, Carnegie Mellon University.

Donald, B. R. 1995. Information invariants in robotics. *Artificial Intelligence* 72:217–304.

Gerkey, B. P., and Mataric, M. J. 2002. Sold! auction methods for multi-robot coordination. *IEEE Transactions on Robotics and Automation* 18(5):758–768.

Gerkey, B., and Mataric, M. J. 2004. A formal analysis and taxonomy of task allocation in multi-robot systems. *International Journal of Robotics Research* 23(9):939–954.

Jones, E.; Browning, B.; Dias, M. B.; Argall, B.; Veloso, M.; and Stentz, A. 2006. Dynamically formed heterogeneous robot teams performing tightly-coordinated tasks. In *Proceedings of IEEE International Conference on Robotics and Automation*.

Kalra, N.; Ferguson, D.; and Stentz, A. 2005. Hoplites: a market-based framework for complex tight coordination in multi-robot teams. In *Proceedings of IEEE International Conference on Robotics and Automation*.

Lin, L., and Zheng, Z. 2005. Combinatorial bids based multi-robot task allocation method. In *Proceedings of IEEE International Conference on Robotics and Automation*.

Luce, R. D., and Raiffa, H. 1957. *Games and Decisions*. John Wiley and Songs, Inc.

Lyons, D. M., and Arbib, M. A. 1989. A formal model of computation for sensory-based robotics. *IEEE Transactions on Robotics and Automation* 5(3):280–293.

Parker, L. E., and Tang, F. 2006. Building multi-robot coalitions through automated task solution synthesis. *Proceedings of IEEE, Special Issue on Multi-Robot Systems*.

Parker, L. E.; Jung, D.; Huntsberger, T.; and Pirjanian, P. 2000. Opportunistic adaptation in space-based robot colonies: Application to site preparation. In *Proceedings of World Automation Congress*.

Parker, L. E. 1994. ALLIANCE: An architecture for fault tolerant, cooperative control of heterogeneous mobile robots. In *Proc. of the 1994 IEEE/RSJ/GI Int'l Conf. on Intelligent Robots and Systems (IROS '94)*, 776–783.

Parker, L. E. 1998. ALLIANCE: An architecture for fault tolerant multi-robot cooperation. *IEEE Transactions on Robotics and Automation* 14(2).

Sandholm, T.; Suri, S.; Gilpin, A.; and Levine, D. 2005. CABOB: a fast optimal algorithm for winner determination in combinatorial auctions. *Management Science* 51(3):374–390. Special issue on Electronic Markets.

Shehory, O., and Kraus, S. 1995. Coalition formation among autonomous agents: strategies and complexity. In Castelfranchi, C., and Muller, J. P., eds., *Lecture Notes in Artificial Intelligence no. 957*. From Reaction to Cognition.

Shehory, O. 1998. Methods for task allocation via agent coalition formation. *Artificial Intelligence* 101(1-2):165–200.

Simmons, R., and Apfelbaum, D. 1998. A task description language for robot control. In *Proceedings of Conference on Intelligent Robotics and Systems*.

Simmons, R.; Singh, S.; Hershberger, D.; Ramos, J.; and Smith, T. 2000. First results in the coordination of heterogeneous robots for large-scale assembly. In *International Symposium on Experimental Robotics*.

Smith, R. G. 1980. The Contract Net Protocol: high-level communication and control in a distributed problem solver. *IEEE Transactions on Computers* C-29(12).

Tang, F., and Parker, L. E. 2005a. ASyMTRe: Automated synthesis of multi-robot task solutions through software reconfiguration. In *Proceedings of IEEE International Conference on Robotics and Automation*.

Tang, F., and Parker, L. E. 2005b. Distributed multi-robot coalitions through ASyMTRe-D. In *Proceedings of IEEE International Conference on Intelligent Robots and Systems*.

Vig, L., and Adams, J. A. 2005. Issues in multi-robot coalition formation. In Parker, L. E.; Schultz, A.; and Schneider, F., eds., *Multi-Robot Systems Volume III: From Swarms to Intelligent Automata*. Kluwer.

Werger, B. B., and Mataric, M. J. 2000. Broadcast of local eligibility for multi-target observation. In *Distributed Autonomous Robotic Systems 4*, 347–356.

Zlot, R., and Stentz, A. 2006. Market-based multirobot coordination for complex tasks. *International Journal of Robotics Research* 25(1). Special Issue on the 4th International Conference on Field and Service Robotics.