Robot Exploration with Combinatorial Auctions

M. Berhault(1) H. Huang(2) P. Keskinocak(2) S. Koenig(1) W. Elmaghraby(2) P. Griffin(2) A. Kleywegt(2)

(1)College of Computing
{marc.berhault,skoenig}@cc.gatech.edu
Georgia Institute of Technology
801 Atlantic Drive, Atlanta GA 30332

(2)School of Industrial and Systems Engineering
{hhuang,pinar,welmagh,pgriffin,anton}@isye.gatech.edu
Georgia Institute of Technology
765 Ferst Drive, Atlanta GA 30332

***Abstract*— We study how to coordinate a team of mobile robots to visit a number of given targets in partially unknown terrain. Robotics researchers have studied single-item auctions (where robots bid on single targets) to perform this exploration task but these do not take synergies between the targets into account. We therefore design combinatorial auctions (where robots bid on bundles of targets), propose different combinatorial bidding strategies and compare their performance with each other, as well as to single-item auctions and an optimal centralized mechanism. Our computational results in TeamBots, a multi-robot simulator, indicate that combinatorial auctions generally lead to significantly superior team performance than single-item auctions, and generate very good results compared to an optimal centralized mechanism.**

I. Introduction

In this joint work between robotics researchers and industrial engineers, we study exploration tasks where a team of mobile robots needs to visit a number of predetermined targets in a partially unknown terrain. Examples of situations where such exploration tasks occur include environmental clean-up missions, space-exploration missions, and search and rescue missions. An important characteristic of these exploration tasks is that the assignment of targets to robots can turn out to be suboptimal as the robots gain more information about the terrain, for example, when a robot suddenly discovers that it is separated by a wall from one of the targets assigned to it. How to assign and re-assign targets to robots is a difficult problem. Centralized control is inefficient in terms of both the required amount of computation and communication since the central controller is the bottleneck of the system. Decentralized control does not have this disadvantage but can result in suboptimal team performance. Researchers in artificial intelligence and robotics have recently investigated market-based approaches, in particular, auctions, as a means for decentralized control. Auctions are efficient in terms of both the required amount of computation and communication since information is compressed into numeric bids that the robots can compute in parallel and can result in near-optimal solutions [1]. They can be used to solve the exploration tasks as follows: Every robot bids on targets and then has to visit all targets that it wins. As the robots gain more information about the terrain during execution, auctions can be held again to change the assignment of targets to robots. So far, researchers in robotics have studied single-item auctions in the context of the exploration tasks, where the targets are auctioned off one at a time. However, single-item auctions can result in highly suboptimal team performance if there are strong synergies between the items for the bidders. Two items are said to exhibit positive (negative) synergy for a bidder if their combined value for the bidder is larger (smaller) than the sum of their individual values. Consider, for example, the gridworld in Figure 1(a) with two robots (R1 and R2) and four targets (G1, G2, G3, and G4). There is a strong positive synergy between targets G3 and G4 for robot R1 because they are close to each other, and the robot can therefore reach the second target with a short travel distance after it has reached the first one. On the other hand, there is a strong negative synergy between targets G1 and G3 for robot R1 because they are on opposite sides of the robot, and the robot can therefore reach the second target only with a long travel distance after it has reached the first one. Combinatorial auctions attempt to remedy the disadvantages of single-item auctions by allowing bidders to bid on bundles of items. If a bidder wins a bundle, they win all the items in that bundle and hence are able to incorporate their synergies into the bids. For example, if the targets are auctioned off in single-item auctions, then robots R1 and R2 first move to targets G3 and G4, respectively, and then to targets G1 and G2, respectively, for a total travel distance of 33 units. In contrast, if the targets are auctioned off in a combinatorial auction, then robot R2 wins bundle {G1, G2}, first moves to target G2 and then to target G1, and robot R1 wins bundle {G3, G4}, first moves to target G3 and then to target G4, for a total travel distance of only 17.

In this paper, we design combinatorial auctions for the exploration tasks, propose different bidding strategies and compare the resulting team performance with that of the other bidding strategies as well as that of single-item auctions and an optimal centralized mechanism. Our

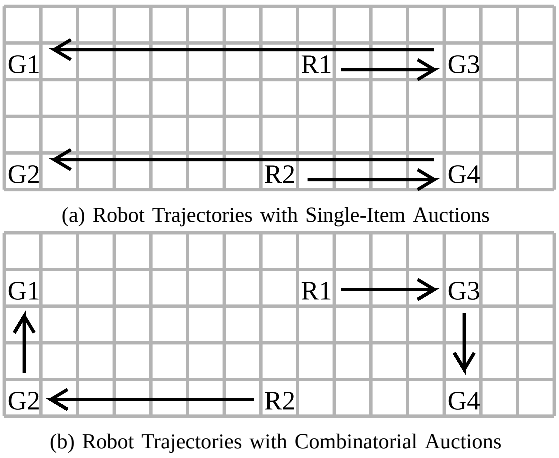


Fig. 1. Motivating Example for Combinatorial Auctions

computational results in TeamBots [2], a multi-robot simulator, indicate that combinatorial auctions generally lead to significantly superior team performance than single-item auctions and generate very good results compared to an optimal centralized mechanism. We also provide insight into the performance of different bidding strategies for combinatorial auctions with respect to criteria such as travel distance, travel time, robot utilization and the amount of communication.

II. Related Work

A wide variety of techniques are being investigated in artificial intelligence and robotics to coordinate teams of robots. An overview can be found in [3]. Although artificial intelligence has studied how to coordinate agents with market-based mechanisms for quite a while [4], robotics has only recently started to investigate how to use them to coordinate teams of robots, and only a limited amount of research exists on the topic. Auction-based systems have been applied to tasks like box pushing [5] and robosoccer [6] but we know of only two applications to exploration tasks, none of which uses combinatorial auctions. Simmons et. al. designed an auction-based system for multi-robot exploration and mapping using single-item first price auctions [7]. The terrain is divided into small cells whose blockage status is unknown in the beginning, and the auctioneer repeatedly auctions off selected frontier cells (unblocked cells bordering cells with unknown blockage status) as targets for the robots. Zlot et. al. designed a similar but decentralized system and studied how different target selection techniques [8] and opportunistic optimization techniques [9] affect the team performance. We extend this research by demonstrating that combinatorial auctions result in better team performance than the single-item auctions used so far. The research on combinatorial auctions in the auction literature has concentrated mainly on auction design (for example, single versus multiple rounds, open versus sealed bids, and bidding rules) and winner determination (computing the optimal allocation of the items to the bidders) [10]. An area that is largely unexplored in the combinatorial auction literature is bidding strategies. Submitting bids for bundles poses a challenging problem for bidders since the number of possible bundles is exponential in the number of items and submitting a bid for every bundle is prohibitively time-consuming. Determining on which bundles to bid and how much to bid remains an open problem. By proposing and testing bidding strategies for our specific application, we therefore contribute to the combinatorial auction literature.

III. Approach

While the auctioneer of a transportation service procurement auction, for example, can determine the participants and the format of the auction but neither the objectives nor the valuations of the auction participants [11], we have complete control over all of these factors. In the following, we first discuss our combinatorial auction mechanism for the exploration tasks and then possible bidding strategies for the robots.

A. Combinatorial Auction Mechanism

For the exploration tasks, robots are a natural choice for the bidders, and targets are a natural choice for the items. The auctioneer is a virtual agent who has sole responsibility for holding auctions and determining their winners but has no other knowledge and cannot control the robots. Initially, no robot owns any targets. Whenever a robot visits a target or gains more information about the terrain, it shares this information with the other robots and the auctioneer starts a new auction that contains all targets that have not yet been visited. (The auctioneer could hold auctions less frequently or with fewer targets, but this would decrease the responsiveness of the robots to new information about the terrain.) Each robot, including the current owner of a target, then generates bids in light of the new information. We use sealed-bid single-round combinatorial auctions. (Alternatively, we could have used multi-round combinatorial auctions, that save bidders from specifying their bids for a large number of bundles in advance, and can be adapted to dynamic environments where bidders and items arrive and depart at different times. However, the auctioneer would then have needed to determine winners in every round and communicate some information about the current bids to the bidders, which would have increased the amount of computation and communication, respectively.) The auctioneer closes the single-round auction after a predetermined amount of time, determines the winning bids, and notifies the winning robots. The winning bids are those that maximize the revenue of the auctioneer with the restriction that each robot wins at most one bundle per auction. This restriction about one winning bundle per robot is because there can be negative synergies between items and thus also between bundles and a robot might not want to win two bundles with negative synergies. Since it is NP-complete to determine the winning bids, we use an approximate winner-determination method that is based on a primal-dual algorithm by Zurel and Nisan [12]. After each auction, the winning robots own the corresponding targets and have the responsibility to visit them, whereas the robots that owned them previously are relieved from that responsibility.

B. Bidding Strategies

The possible bidding strategies of the robots depend on the rewards and costs that they incur. We assume that the robot that visits a target for the first time receives a reward that is the same for all targets and sufficiently large so that all robots bid on all targets. Each robot has to pay the amount of its bids for the bundles that it wins and one dollar for each unit of distance that it travels. Robotics researchers have used the following simple bidding strategy for single-item auctions:

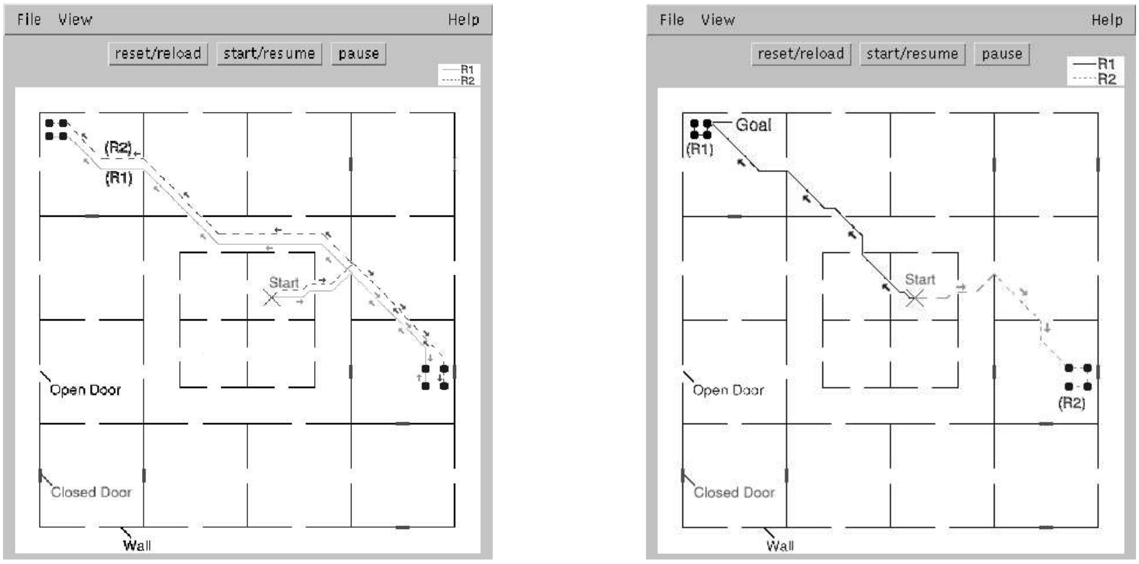
- ***Single***: Each robot bids its surplus for a target, that is, the reward that it receives for the target minus the optimistic travel cost for visiting the target from its current location (that is, the distance from its current location to the target under the assumption that unknown terrain does not contain obstacles).

We use this strategy as a benchmark and generalize it to bundles of targets, where each robot continues to bid its surplus for a bundle, that is, the rewards that it receives for the targets minus the travel cost for visiting all targets from its current location. We estimate the travel cost with a nearest neighbor heuristic, that is, the distance from its current location under the assumption that it repeatedly moves to the closest unvisited target in the bundle and the terrain does not contain obstacles. The distances were calculated with D* Lite [13]. The main question then is on which bundles the robots bid. Since auctions close after a predetermined amount of time, the robots need to compute their bids and communicate this information to the auctioneer within the time limit, which effectively limits the number of bids they can submit. We therefore explore the following bidding strategies:

- ***Three-Combination***: Bid on all bundles with no more than n targets. This strategy quickly becomes infeasible for large n since the number of bundles increases exponentially in n. We therefore used $n = 3$ in our experiments.
- ***Smart-Combination***: Bid on all bundles that contain only one or two targets. Additionally, bid on the $6k$ bundles that have the highest surplus among all bundles containing $2 < l$ targets, where k is the total number of clusters (as explained in the experimental section). We used $l = 3, 4, 5, 6$ in our experiments, that is, the robots bid on $4 \times 6k$ budles containing three, four, five, and six targets, respectively.
- ***Nearest-Neighbor***: Bid on all bundles that correspond to good sequences of targets, where good sequences are recursively defined as follows: Each single target is a good sequence. Appending target t to a good sequence ending in target s yields another good sequence if the surplus of the new sequence is greater than or equal to the surplus of the old sequence and t is the closest target to s among all targets not in the old sequence.
- ***Graph-Cut***: Generate a complete undirected graph whose vertices correspond to the targets. The cost of an edge between two targets corresponds to the optimistic travel cost between them. Generate a bundle that contains all targets in the graph. If the graph contains more than one target, generate additional bundles by using the maximum cut algorithm to split the graph into two connected subgraphs and invoke the algorithm recursively for each of the two subgraphs. Since computing the maximum cut is NP-complete, we used the “Computation Optimization Laboratory: Graph-Partition and Box-Constrained Quadratic Optimization” by Benson, Ye and Zhang to compute an approximation [14].

IV. Experiments

We implemented our combinatorial auction mechanism in Teambots [2], a multi-robot simulator. We tested our four bidding strategies as well as ***Single*** with a team of three robots that navigated in a virtual building composed of rooms connected by doors that were closed with probability 0.2. Figure 2(a) shows the layout of the building, where the gray lines are walls, some of which contain small black lines that represent the closed doors. Two of the most important factors that affect the team performance are the location of the targets and the prior knowledge of the robots. We therefore conducted experiments with six different ways of distributing the targets in the building: The ***Uniform*** terrain contained eight targets that were distributed with uniform probability in the building, while the ***Cluster_i*** terrain (for $i = 1 \ldots 5$) contained i clusters of four targets each that were distributed with uniform probability in the building. We also ran experiments for two different kinds of prior knowledge of the robots: In completely known terrain each robot knew a complete map of the building in advance (including which doors were closed), while in partially unknown terrain each robot knew the locations of the walls, doors and targets to be visited in advance but did not know which doors were closed. The robot that reached a door first discovered its state and broadcast it to the



(a) Robot Trajectories using ***Single*** (b) Robot Trajectories using ***Graph-Cut***

Fig. 2. Robot Trajectories (Screenshots of TeamBots)

other robots. Figure 2, for example, shows the trajectories of a team of two robots in a completely known ***Cluster_2*** terrain, depending on whether they used ***Single*** or ***Graph-Cut*** as their bidding strategy. (We used three robots in the experiments but for clarity we show a screenshot with only two robots.)

Table I reports the results of our experiments. It contains four numbers for each combination of bidding strategy and prior knowledge in each of the six kinds of terrain. All reported numbers are the averages of ten runs with different locations of the targets or clusters: "Number of bids" is the total number of bids submitted by all robots during a run. It measures the amount of communication and computation for determining the bundles, the bids for the bundles, and the winners of the auctions. "Travel cost" is the sum of the travel distances of all robots, while "travel time" (or makespan) is the total time for solving the exploration tasks if both communication and computation are instantaneous. The travel cost and travel time are two different measures for how efficient the allocation of targets to robots is (performance measures). The travel cost roughly determines the amount of energy consumed, and the travel time measures how fast the exploration tasks are solved. The table also reports the optimal travel cost in completely known terrain for comparison purposes. We are mostly interested in the performance of our bidding strategies in partially unknown terrain and therefore would like to compare their travel costs and travel times to the optimal solution; however, finding the optimal solution in unknown terrain is often computationally intractable. Instead, we compute the optimal travel costs in completely known terrain by modeling the problem as a linear integer program and solving it with CPLEX, a commercial software package. (Note that the minimal travel cost in known terrain is always a lower bound on the minimal travel cost in unknown terrain.) Finally, "robot utilization" is the percentage of time during which robots are moving. For example, if one robot is always moving while the other two robots are always idle, robot utilization is 33.3. Similarly, if each robot is moving only one third of the time, robot utilization is again 33.3.

We make the following observations about the travel costs of the five bidding strategies:

- When the targets are distributed uniformly, the total travel costs of all bidding strategies are fairly close to each other.
- When the targets are clustered, the travel costs satisfy ***Graph-Cut*** $\leq$ ***Nearest-Neighbor*** $\leq$ ***Smart-Combination*** $\leq$ ***Three-Combination*** $\leq$ ***Single***, with four exceptions: In partially unknown terrain, the travel cost of ***Nearest-Neighbor*** is smaller than the travel cost of ***Graph-Cut*** in ***Cluster_3***, the travel cost of ***Three-Combination*** is smaller than the travel cost of ***Smart-Combination*** in ***Cluster_4***, and the travel cost of ***Smart-Combination*** is smaller than the travel cost of ***Nearest-Neighbor*** in ***Cluster_5***. In completely known terrain, the travel cost of ***Three-Combination*** is smaller than the travel cost of ***Smart-Combination*** in ***Cluster_3***. Some of these exceptions are due to a single run. Thus, the travel costs of combinatorial auctions are smaller than the travel costs of the single-item auction. For example, the travel cost of ***Graph-Cut*** in terrain ***Cluster_4*** is only about one-third of the travel cost of ***Single***. This is due to the fact that

TABLE I

EXPERIMENTAL DATA

	Partially Unknown Terrain						***Completely Known Terrain***					
Type	***Uniform***	***Cluster_1***	***Cluster_2***	***Cluster_3***	***Cluster_4***	***Cluster_5***	***Uniform***	***Cluster_1***	***Cluster_2***	***Cluster_3***	***Cluster_4***	***Cluster_5***
Single												
number of bids	102.3	28.5	112.8	243.3	412.2	632.1	101.7	29.1	111.0	238.8	392.1	635.1
travel cost	296.9	123.3	206.4	333.0	511.9	434.6	252.4	120.1	197.4	294.9	465.9	426.5
travel time	1554.1	495.0	807.9	1303.6	1979.0	1716.1	1237.8	481.0	771.9	1196.1	1798.1	1690.0
robot utilization	71.0	92.0	93.8	93.6	95.3	93.4	75.8	92.1	93.8	91.8	95.5	92.6
Three-Combination												
number of bids	692.4	70.8	709.2	3090.0	9418.5	20488.5	692.4	72.9	709.2	3079.2	9333.6	20506.5
travel cost	298.8	91.4	149.1	160.5	245.1	263.2	230.4	82.1	123.0	142.4	224.6	247.9
travel time	1736.9	533.5	698.1	728.1	1086.4	1151.8	1166.8	478.8	558.4	625.9	1031.4	1055.7
robot utilization	65.5	63.3	80.0	83.1	84.6	85.1	74.4	63.4	81.9	84.9	82.5	87.5
Smart-Combination												
number of bids	739.8	66.0	743.1	2392.8	5116.5	8620.2	745.5	68.1	739.2	2366.4	5086.8	8731.5
travel cost	303.8	91.3	121.0	158.7	264.9	246.8	246.8	82.0	97.2	152.9	206.5	241.7
travel time	1758.2	531.1	709.9	794.2	1243.1	1153.1	1265.7	480.3	574.9	733.5	969.5	1105.8
robot utilization	64.5	63.4	63.1	75.1	79.3	79.5	72.8	63.2	62.9	77.6	79.9	81.7
Nearest-Neighbor												
number of bids	213.6	60.0	241.4	550.7	1075.4	1690.0	203.6	60.0	239.2	540.5	1005.4	1591.9
travel cost	291.1	50.0	92.5	123.5	228.6	261.2	247.5	45.2	79.8	116.1	178.0	231.8
travel time	1645.2	568.8	716.8	778.6	1378.4	1584.4	1247.8	514.9	569.1	685.0	951.6	1148.9
robot utilization	66.7	32.8	49.8	61.0	61.5	63.1	73.9	32.8	52.4	64.7	72.6	75.8
Graph-Cut												
number of bids	192.0	48.0	182.7	408.0	719.4	1095.6	183.9	48.0	182.7	403.2	717.0	1112.1
travel cost	292.0	49.9	84.4	129.7	185.0	210.0	244.4	45.1	73.1	111.1	170.8	184.1
travel time	2137.1	568.0	809.8	1116.4	1200.1	1470.7	1315.2	514.1	655.9	725.4	911.2	1076.2
robot utilization	53.2	32.8	39.3	49.0	58.6	53.2	69.5	32.8	42.6	60.7	71.2	63.9
Optimal												
travel cost							208.7	47.6	77.4	115.5	171.4	184.4

the single-item auction tends to send all robots to the closest cluster, which often moves them away from the other clusters, as seen in Figure 2(a).

- When the targets are clustered in completely known terrain, the travel costs of ***Graph-Cut*** are approximately equal to the optimal travel costs. This is important because the computation times of ***Graph-Cut*** are very small, whereas the computation times needed to determine the optimal travel costs explode. For example, it took several days to compute the optimal travel costs in completely known ***Cluster_4*** and ***Cluster_5*** terrain. (The travel costs of ***Graph-Cut*** are often smaller than the optimal travel costs due to discretization errors: Our software discretizes terrain into cells and considers a target to be reached once a robot has entered the cell of the target.)

To summarize, ***Graph-Cut*** appears to result in smaller travel costs than the other bidding strategies. The results about the travel times are less clear but it appears that ***Three-Combination*** tends to result in small travel times.

We make the following observation about the numbers of bids of the five bidding strategies:

- When the targets are clustered, the numbers of bids satisfy ***Single*** $\leq$ ***Graph-Cut*** $\leq$ ***Nearest-Neighbor*** $\leq$ ***Smart-Combination*** $\leq$ ***Three-Combination***, with two exceptions: The number of bids of ***Three-Combination*** is smaller than the number of bids of ***Smart-Combination*** in both partially unknown and completely known ***Cluster_2*** terrain.

To summarize, ***Graph-Cut*** results in smaller numbers of bids than the other combinatorial bidding strategies. This is interesting because, in general, it also results in smaller travel costs than the other combinatorial bidding strategies. In general, one would expect larger numbers of bids to result in smaller travel costs because they allow bidders more flexibility in expressing synergies. However, it is undesirable to have a large number of bids since this increases the communication and computation time. Consequently, it is important to develop bidding strategies, such as ***Graph-Cut***, that carefully select which bundles to bid on and thus achieve small travel costs with small numbers of bids.

We make the following observation about the robot utilization of the five bidding strategies:

- When the targets are clustered, the robot utilization satisfies ***Graph-Cut*** $\leq$ ***Nearest-Neighbor*** $\leq$ ***Smart-Combination*** $\leq$ ***Three-Combination*** $\leq$ ***Single*** with one exception: The robot utilization of ***Three-Combination*** is smaller than the robot utilization of ***Smart-Combination*** in partially unknown ***Cluster_1*** terrain.

To summarize, ***Graph-Cut*** appears to utilize fewer robots than the other bidding strategies. In general, one can expect smaller robot utilization to result in smaller travel costs but larger travel times because smaller robot utilization allows for less parallelism and, indeed, ***Graph-Cut*** results in smaller travel costs than the other bidding strategies in general. Similarly, ***Three-Combination*** tends to utilize more robots than the other bidding strategies and appeared to result in small travel times. If robots are costly or it is difficult to operate a large number of robots for other reasons (for example, because one human operator has to control all robots at the same time), it might be desirable for the number of active robots to be small. However, if the travel time is important then it might be desirable for the number of active robots to be large. In this case, one can modify the auction mechanism to utilize more robots. For example, one can impose upper bounds on bundle sizes to ensure a more balanced allocation of targets to robots. The maximum bundle size is three for ***Three-Combination*** and six for ***Smart-Combination***, whereas there is no upper bound on the bundle size of the other two combinatorial bidding strategies. Therefore, the winning bids for ***Nearest-Neighbor*** and ***Graph-Cut*** tend to have large bundle sizes, resulting in a smaller number of winners and thus active robots. This partially explains why ***Three-Combination*** and ***Smart-Combination*** utilize more robots than ***Nearest-Neighbor*** and ***Graph-Cut***.

V. Conclusion

In this paper, we studied how to coordinate a team of mobile robots to visit a number of given targets in partially unknown terrain. Our experimental results show a substantial advantage of combinatorial auctions over single-item auctions. They also show the large influence of the combinatorial bidding strategy on the team performance, where our ***Graph-Cut*** strategy clearly outperformed three other combinatorial strategies with respect to travel cost. Our future work will concentrate on developing more sophisticated bidding strategies, including bidding strategies that minimize the travel time rather than travel cost, and porting our code to ATRV Minis.

Acknowledgments

This research is partly supported by an NSF award under contract ITR/AP-0113881. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the sponsoring organizations, agencies, companies or the U.S. government.

VI. References

[1] A. Stentz and M. Dias, "A free market architecture for coordinating multiple robots," Tech. Rep. CMU-RI-TR-99-42, Robotics Institute, Carnegie Mellon University, Pittsburgh (Pennsylvania), 1999.

[2] T. Balch and A. Ram, "Integrating robotics research with JavaBots," in *Proceedings of the AAAI Spring Symposium*, 1998.

[3] P. Stone and M. Veloso, "Multiagent systems: A survey from a machine learning perspective," *Autonomous Robots*, vol. 8, no. 3, pp. 345–383, 2000.

[4] M. Wellman, "The economic approach to artificial intelligence," *ACM Computing Surveys*, vol. 7, no. 3, pp. 360–362, 1995.

[5] B. Gerkey and M. Mataric, "Sold! Auction methods for multi-robot coordination," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 758–768, 2002.

[6] R. Nair, T. Ito, M. Tambe, and S. Marsella, "Task allocation in the rescue simulation domain: A short note," in *RoboCup-2001: Robot Soccer World Cup V*, A. Birk and S. Coradeschi, Eds., Lecture Notes in Computer Science. Springer, 2002.

[7] R. Simmons, D. Apfelbaum, W. Burgard, D. Fox, M. Moors, S. Thrun, and H. Younes, "Coordination for multi-robot exploration and mapping," in *Proceedings of the National Conference on Artificial Intelligence*, 2000, pp. 852–858.

[8] R. Zlot, A. Stentz, M. Dias, and S. Thayer, "Multi-robot exploration controlled by a market economy," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2002, pp. 3016–3023.

[9] M. Dias and A. Stentz, "Opportunistic optimization for market-based multirobot control," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, 2002, pp. 2714–2720.

[10] S. de Vries and R. Vohra, "Combinatorial auctions: A survey," *INFORMS Journal of Computing*, (forthcoming).

[11] W. Elmaghraby and P. Keskinocak, "Combinatorial auctions in procurement," in *to appear in The Practice of Supply Chain Management*, C. Billington, T. Harrison, H. Lee, and J. Neale, Eds. Kluwer, 2002.

[12] E. Zurel and N. Nisan, "An efficient approximate allocation algorithm for combinatorial auctions," in *Proceedings of the ACM Conference on Electronic Commerce*, 2001.

[13] S. Koenig and M. Likhachev, "Improved fast replanning for robot navigation in unknown terrain," in *Proceedings of the International Conference on Robotics and Automation*, 2002, pp. 968–975.

[14] Y. Ye, "Software: COPL_DSDP," in *ftp://dollar.biz.uiowa.edu/pub/yyye/copldsdp.zip*.