Development and Simulation of a Task Assignment Model for Multirobot Systems

B.B.Biswal bbbiswal@nitrkl.ac.in
Professor, Department of Mechanical Engineering
National Institute of Technology, Rourkela
769008, India

B.B.Choudhury bbcnit@gmail.com
Ph.D Scholar, Department of Mechanical Engineering
National Institute of Technology, Rourkela
769008, India

Abstract

Multirobot systems (MRS) hold the promise of improved performance and increased fault tolerance for large-scale problems. A robot team can accomplish a given task more quickly than a single agent by executing them concurrently. A team can also make effective use of specialists designed for a single purpose rather than requiring that a single robot be a generalist. Multirobot coordination, however, is a complex problem. An empirical study is described in the present paper that sought general guidelines for task allocation strategies. Different task allocation strategies are identified, and demonstrated in the multi-robot environment. A simulation study of the methodology is carried out in a simulated grid world. The results show that there is no single strategy that produces best performance in all cases, and that the best task allocation strategy changes as a function of the noise in the system. This result is significant, and shows the need for further investigation of task allocation strategies.

Keywords: Multirobot, task allocation, allocation strategies, auction algorithms

1. INTRODUCTION

The study of MRS has received increased attention in the recent years. This is not surprising as continually improving technology has made the deployment of MRS consisting of increasingly larger number of robots possible. It is obvious that, at least in some important respects, multiple robots will be superior to a single robot in achieving a given task. Potential advantages of MRS over a SRS (Single robot systems) include reduction of total system cost by employing multiple simple and cheap robots as opposed to a single, complex and expensive robots. Furthermore, the inherent complexity of certain task environment may require the use of multiple robots as the demand for capability is quite substantial to be met by a single robot. Finally, multiple robots are assumed to increase system robustness by taking advantage of inherent parallelism and redundancy.

Multirobot teamwork is a complex problem consisting of task division, task allocation, coordination, and communication. The most significant concept in multi-robot systems is cooperation. It is only through cooperative task performance that the superiority of robot groups can be demonstrated. The cooperation of robots in a group can be classified into two categories of implicit cooperation and explicit cooperation. In the implicit cooperation case each robot performs individual tasks, while the collection of these tasks is toward a unified mission. This type of group behavior is also called asynchronous cooperation, as it requires no synchronization in time or space. The explicit cooperation is the case where robots in a team work synchronously with respect to time or space in order to achieve a goal. One example of such cooperation is transportation of heavy objects by multiple robots, each having to contribute to the lifting and moving of the object. This task requires the robots to be positioned suitably with respect to each other and to function simultaneously. Regardless of the type of cooperation, the goal of the team must be transformed in to tasks to be allocated to the individual robots.

There is no general theory of task allocation in uncertain multi-robot domains. In this paper, an attempt is made to empirically derive some guidelines for selecting task allocation strategies for multi-robot systems with implicit cooperation. The explored strategies are individualistic in that they do not involve explicit cooperation and negotiation among the robots. However, they are a part of a large class approaches that produce coherent and efficient cooperative behavior. Given the empirical nature of this work and the scope of the problem addressed, these guidelines are necessarily incomplete, though they provide useful insight. The choice of task allocation strategy is far from trivial and that no optimal task allocation strategy exists for all domains. It can be very difficult to identify the optimal task allocation strategy even for a particular task. These results are derived through the use of a framework developed for understanding the task allocation problem, which illustrates a common approach to decomposing the problem. The approach presented in this paper can be advantageously used in real-world problems.

2. RELATED WORK

Multirobot systems are becoming increasingly more capable and the types of achievable applications for teams of robots are becoming progressively more complex. Many approaches to multirobot coordination rely on a mechanism for task allocation to determine an efficient assignment of tasks to robots. However, existing techniques do not fully consider the complexity of the tasks to be allocated. For the most part, tasks are assumed to be atomic units that can be performed by one or more robots on the team. In practice, this usually means that tasks are either acquired from a central planner that decomposes the mission goals, or that tasks are specified as input by a system user. In any case, existing task allocation algorithms consider the tasks only in terms of the level of description provided by the user or the planner. Another main issue in task allocation is the study of multi-robot systems in hardware with small population sizes (e.g., under twenty), versus the study of issues in multi-agents systems in simulation with large population sizes. It should be noted that the effects of team size and its scaling are integral issues in robot group studies, and the reliability of the simulation results remains to be seen.

One main issue in task allocation is the division of the tasks into homogeneous and heterogeneous tasks. Goldberg and Mataric [1, 2, 3] studied homogeneous and heterogeneous task allocation for a foraging task, namely trash collection. Their implementation ranges from homogenous system where all robots have the same task to a grouping, which divides the robots in different groups, and each group is assigned to do a different task. They use inference, spatial, and temporal parameters to evaluate different methods. The results show that although the grouping system is suitable for reducing interference, the best performance is obtained through homogenous task allocation, i.e., the fastest collection of trash than others. In another work, Parker [4] showed that augmenting homogenous task allocation by making robots more team-aware, results in systems that are substantially more efficient. Dudek et al. [5] worked out a general taxonomy to characterize multi-agent systems, consisting of the number of agents, communication (range, bandwidth and topology), reconfigurability, processing mechanism, and differentiation.

Berstas [6] presents an algorithm that can be utilized in task allocation in multi-robot applications, especially suitable for parallel computation. This approach attempts to find the best assignment between tasks and users, while maximizing the total benefit. It iterates between users and during iteration it tries to assign a task to a user who offers the most. The majority of multirobot systems that utilize an explicit task allocation mechanism assume either that a static set of tasks is given to the system as input [7, 8, 9, 10], or that tasks arrive dynamically, either from external [8, 9] or internal [11, 12] sources. In any case, such approaches search for an efficient assignment of the current task set to robots, assuming that all tasks are indivisible. When this type of mechanism is applied to complex tasks, a robot assigned a task can decompose it and then execute the resulting simple tasks [7]. In reality, however, it may be beneficial to allocate subcomponents of these tasks to more than one, and generally the preferred task decomposition will depend on the subtask assignments. Therefore, treating tasks as atomic entities during allocation is not always prudent.

A common alternative among systems that explicitly handle complex tasks is a two-stage approach: first decompose all tasks and then distribute the resulting set of subtasks [12, 13, 14]. The main drawback of this approach is that task decomposition is performed without knowledge of the eventual task allocation; therefore the cost of the final plan cannot be fully considered. Since there is no backtracking, costly mistakes in the central decompositions cannot be rectified. In some instances, the central plan is left intentionally vague, which allows for a limited amount of flexibility in modifying it later. For example, in GOFER Project [14], the central planner produces a general plan structure for which individual robots can later instantiate some variables; while in the "mapping algorithm" of Simmons et al. [11], is an on-line approach to likelihood maximization that uses hill climbing to find maps that are maximally consistent with sensor data and odometry. Ostergaard and Mataric [15] propose an algorithm for task allocation that assigns tasks dynamically to a suitable and capable robot. Task allocation is dynamic and happens on a needed basis. Task allocation is one of the main problems in multirobot systems. Guerrero and Oliver[16] propose a methodology to allocate tasks in a multirobot systems by considering among other factors, to get a good task allocation, and to take into account the physical interference effects between robots, that is, when two or more robots want to access to the same point at the same time. Lian and Murray [17] discuss a design methodology of cooperative trajectory generation for multi-robot systems. The trajectory of achieving cooperative tasks, i.e., with temporal constraints, is constructed by a nonlinear trajectory generation (NTG) algorithm. In this paper three scenarios of robot tasking from home base to target position. Stenz and Dias [18] implement task allocation as a free market system. Some of the important features of this approach are dynamical task allocation, group learning, and minimum communication dependability. Shen, Tzeng and Liu [19] implement workflow modelers, during workflow design and specify the performers of a task by their organizational role. However, during workflow enactment, numerous agents with different skills and expertise may share the same role in an organization, making it hard to select appropriate individuals based merely on the assignment relation between a role and a task. The Alliance approach [20] is focused on small to medium size robot teams. It is a fault-tolerant, behavior-based architecture that assigns tasks dynamically. Its behavior-based controller uses different sets of behavior for different tasks. This architecture assumes a heterogeneous team of robots. Each robot needs to run an Alliance process as a requirement in order to cooperate. Each task consists of a target location that needs to be visited by a robot. The objective of the allocation is to minimize the total cost, that is, the sum of the travel costs of all robots for visiting a target and finding an optimal allocation is an NP-hard problem, even in known environments. The PRIM ALLOCATION [21], is a simple and fast approximate algorithm for allocating targets to robots which provably computes allocations whose total cost is at most twice as large as the optimal total cost. Skrzypczyk [22] discusses a problem of planning and coordination in a multi robot system and considers a team of robots that performs a global task in a human-made workspace of complex structure. A hybrid architecture of the team motion control system is considered in the work. The system is split into two layers: the planner module and the behavior based collision free motion controller that is designed to perform several elementary navigation tasks. The role of the planner is to plan and coordinate execution of elementary tasks by individual agents to obtain performance of global task. The method of elementary tasks planning based on N-person game. An algorithm of multi robot workspace exploration is presented as an example of application of the proposed method. Simulation of the algorithm is carried out, and its result is presented and discussed in the paper. Mosteo and Montano[23] discuss a novel approach in networked robotics for optimal allocation with interchangeable objective functions, from minimizing the worst-case cost of any agent in a multi-robot team in time-critical missions, to minimizing the team usage of resources. They propose a general model for flexible mission planning, using hierarchical task networks as descriptive framework, the multiple traveling salesmen as optimization model, and distributed simulated annealing for solution search in very large solution spaces. This proposal does not discard viable solutions, hence the optimal one for the model may be eventually found. Boneschanscher [24] presents a task assigner for a flexible assembly cell (FAC) incorporating multiple robots and a transport system. The FAC can assemble a wide range of products in small batches. Parts are fed on pallets and assembled on fixtures, which both can route through the cell. The FAC has a limited buffer capacity. The task assigner determines a schedule for each batch, with minimum assembly time as the main objective. Task assignment is done for a limited time horizon, using a goal directed search. The time horizon is determined by the limited buffer capacity of the FAC. While assigning tasks to resources in the cell, the task assigner determines an appropriate assembly sequence and allocates tools such as grippers to workstations in the cell. It is evident that the allocation strategy is not a generalist but is situation driven. The present method attempts to develop and implement a suitable model for an implicit cooperation environment based upon the capability of the candidates to handle the tasks.

3. DYNAMIC TASK ASSIGNMENT

In the context of multi-robot coordination, dynamic task allocation can be viewed as the selection of appropriate actions [25] for each robot at each point in time so as to achieve the completion of the global task by the team as a whole. From a global perspective, in multi-robot coordination, action selection is based on the mapping from the combined robot state space to the combined robot action space. For homogeneous robots, it is the mapping;

$$S^{|R|} \rightarrow A^{|R|}$$

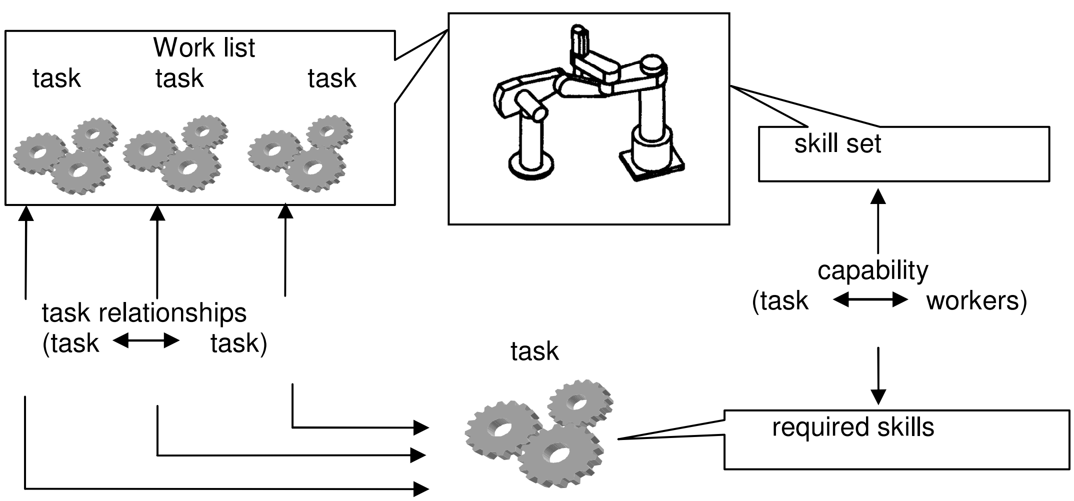

FIGURE 1: Evaluation Criteria

where, S is the state space of a robot, $|R|$ is the number of robots, and A is the set of actions available to a robot [26]. In practice, even with a small number of robots, this is an extremely high-dimensional mapping, a key motivation for decomposing and distributing control. Based on the approach introduced in [27], the task allocation problem is decomposed into the following three steps:

1. each robot bids on a task based on its perceived fitness to perform the task;
2. an auctioning mechanism decides which robot gets the task;
3. the winning robot's controller performs one or more actions to execute the task.

The above decomposition is aimed at constructing a general formulation for the multi-robot coordination problem. In this formulation, a bidding function determines each robot's ability to perform a task based on that robot's state. Next, the task allocation mechanism determines which robot should perform a particular task based on the bids. Finally, the robot controllers determine appropriate actions for each robot, based on the robot's current task engagement. This partitioning, as illustrated in Figure 1, serves two purposes: it reduces the dimensionality of the coordination problem, and it reduces the amount of inter-robot communication required.

We now have the mapping

$$B^{|R||T|} \rightarrow T^{|R|}$$

Instead of mapping, namely from all robots' bids B for all tasks T to a task assignment for each robot, this overall mapping is called the task allocation strategy for the system as a whole. The overall mapping is treated here as a global, centralized process (as depicted in Figure 2)**,** but distributed auctioning mechanisms [27, 28], blackboard algorithms [29], and cross-inhibition of behaviors [30] are some validated methods for distributing the task allocation function. In this methodology, the focus is on what the task allocation function should be, rather than on how it should be distributed. The above framework is a general way that dynamic task allocation for multi-robot systems can be formulated.

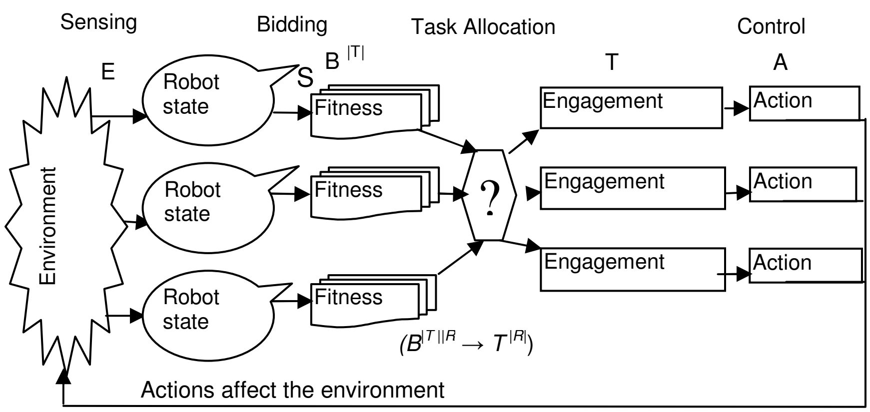

FIGURE 2: Reducing the Dimensionality of Multi-Robot Coordination

3.1 Auction Algorithm

The auction algorithm is an intuitive method for solving the classical assignment problems. It outperforms substantially its main competitors for important types of problems, both in theory and practice, and is also naturally well suited for parallel computation. In the process, the user submits jobs to the auctioneer to start the process. An auctioneer is responsible for submitting and monitoring jobs on the user's behalf. The auctioneer creates an auction and sets additional parameters of the auction such as job length, the quantity of auction rounds, the reserve price and the policy to be used. The auctioneer informs the robots (Robot-1, Robot-2 and Robot-3) that an auction is about to start. Then, the auctioneer creates a call for proposals, sets its initial price, and broadcasts calls to all the robots (Robot-1, Robot-2 and Robot-3). Robots formulate bids for selling a service to the user to execute the job. The robots evaluate the proposal; they decide not to bid because the price offered is below what they are willing to charge for the service. This makes the auctioneer to increase the price and send a new call for proposal with this increase in the price. Meanwhile, the auctioneer keeps updating the information about the auction. In the second round, Robots are decided to bid. The auctioneer clears the auction according to the policy specified beforehand. Once the auction clears, it informs the outcome to the user and the robots. The flowchart for the process is presented in Figure 3.

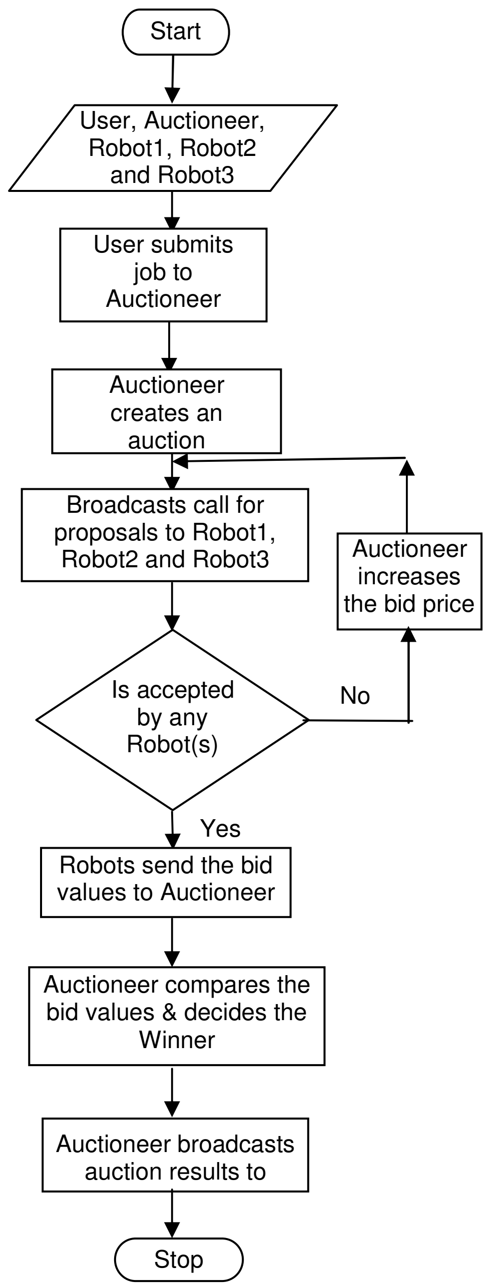

FIGURE 3: Flowchart of the Auction for Task Allocation

The algorithm described here can be utilized in task allocation in multi-robot applications, and is particularly suitable for parallel computation. This approach attempts to find the best assignment between tasks and robots, while maximizing the total benefit. It iterates between robots and in each iterations tries to assign a task to a robot who offers the most. In consecutive iterations, other robots may bid for other tasks and if more than one bids are available for the same task, it will increase the cost of task until finally just one task-robot pair match takes place, (iterative improvement). The iteration terminates when all robots are pleased with their match, otherwise an unhappy robot will bid higher for another task and this process will continue. Although auction algorithm may have some similarities to the free market approach, there is a little difference. One difference is that in the free market approach, agents can cooperate in order to gain a maximum profit for all of them, however in the auction algorithm every robot is considered rival. Another dissimilarity is that the auction algorithm uses an exclusive mathematical model for all the applications, while the free market approach does not. In addition, the free market technique is based on the collection of heterogeneous agents, while in the auction algorithm the robot set is homogeneous.

3.2 Task Allocation Strategies

The dynamic task allocation problem, i.e., the mapping from bids to tasks, can be performed in numerous ways. The focus is limited here to Markovian systems, where the task allocation mapping for a given robot is based on the mapping between that robot's current task assignments and every other robot's current bid on each task, to the given robot's new task assignment, as shown in Figure 4. Given each robot's bid on each task and each robot's current task engagement, each robot's new task assignment need to be determined. The effects of two key aspects of distributed control, commitment and coordination, on performance are explored.

Given the large space of possibilities, only the extreme cases of each: no commitment and full commitment, and no coordination and full coordination are considered. The combination of these extremes results in four task allocation strategies as shown in Figure 5. Along the commitment axis, a fully committed strategy meant a robot would complete its assigned task before considering any new engagements, while a fully opportunistic strategy allowed a robot to drop an ongoing engagement at any time in favor of a new one. Along the coordination axis, the uncoordinated (individualistic) strategy meant each robot performed based on its local information, while a coordinated strategy simply implemented mutual exclusion, so only one robot could be assigned to a task, and no redundancies were allowed. It is noted that this notion of coordination is simple, and it is not intended to represent explicit cooperation and coordination strategies (i.e., the fixed time-cost was 0). During the process three new tasks appear every twelve time-steps at random positions on the grid. The tasks are structured so that one robot is sufficient for completion of an individual task assignment.

Current engagement	Bids	A	B	C	D	New engagement
A	R1	6	4	2	5	?
--	R2	4	1	0	3	?
C	R3	7	2	3	2	?

FIGURE 4: An Example Task Allocation Scenario

Commitment 🡇	Coordination🡆	
	Individual	Mutually Exclusive
Commitment	Strategy.1	Strategy.2
Opportunity	Strategy.3	Strategy.4

FIGURE 5: The Four Task Allocation Strategies

Thus, mutual exclusion is the simplest yet effective form of coordination. As an example, the fully committed mutually exclusive strategy is as follows:

1. If a robot is currently engaged in a task, and its bid on that task is greater than zero, remove the row and column of the bid from the table, and set the robot's new assignment to its current one.
2. Find the highest bid in the remaining table. Assign the corresponding robot to the corresponding task. Remove the row and column of the bid from the table.

3. Repeat from step 2 until there are no more bids. In case of individualistic (uncoordinated) strategies, the same algorithm is run on a separate table for each robot. In the opportunistic (uncommitted) case, step 1 above is skipped.

4. GRID WORLD EXPERMENTAL FRAME WORK

A simplified version of the above described multi-robot task in a grid world is illustrated in Figure 6. As the base case of the grid world implementation, a 10×10 grid inhabited by 10 robots is considered. Robots bid on tasks depending on their capability (expressed by a number) to those tasks. The bid was set to 20 – d, where d is the Manhattan distance to the task. In each time-step, any robot assigned to a particular task selects that task. When a robot selects a task, that task goes off the list and new tasks are added to it. In order to explore the parameter space of the task, we focused on commitment and coordination. In the context of emergency handling, commitment means that robots stay focused on a single task, until the task is over. The opposite, opportunism*,* means that robots can switch tasks, if for example another task is found with greater intensity or priority. In the experiments, coordination is linked to communication, namely the ability of robots to communicate about who should service which tasks, as opposed to individualism*,* where robots have no awareness of each other. Communication is used to prevent multiple robots from trying to accomplish the same task; robots inhibit others from engaging in the same task. The goal is to reduce interference among robots, and to prevent loss of coverage in some areas because all the robots rush to perform task in another area. Deciding the level of commitment and collaboration are key aspects of the multi-robot task allocation problem. Four experiments were designed resulting from the combinations in varying the two parameters, coordination and commitment. The results of the grid world simulation are presented in Figure 7. On one axis we test commitment versus opportunism, and on the other we test individualism versus mutual exclusion.

		R		T					
								R	
						R			
	R								
			T						
								T	

FIGURE 6: An Example 10 x 10 Grid World with Four Robots and Three Tasks.

Strategy:	I,O	I,C	M,O	M,C
Results:	980	1045	435	722

FIGURE 7: Results from Base Case Grid World

5. BLACK BOARD ALGORITHM

In order to ensure reasonable scalability and robustness, communication among the robots is done through a "blackboard"[29]. To simulate experiments with inter-robot communication, each robot sends its relevant state information to the blackboard, and the blackboard information is read by all the robots. In the case of no communication, the blackboard just contains information from one robot (itself). The information on the blackboard is the current engagement of each robot. Intuitively, if all robots have the same blackboard information available and execute the same algorithm, they should all come to the same conclusion as to which robot should pursue which task.

To facilitate validation of the experiments, all parameters are held constant, except the way the information on the blackboard is handled. The algorithm for deciding on the allocation of the tasks to individual robots is as follows:

Step 1: All robots engaged in a task cannot have their engagement set to “none”
Step 2: In case of commitment, all entries in the blackboard for robots already pursuing a task is set to zero, along with all entries for task already being pursued. In case of opportunism, this step is skipped.
Step 3: The highest non-zero score in the table is checked, and the robot corresponding to this entry is assigned to the task corresponding to this entry.

Results	Individual		Mutual Exclusion	
Commitment	2063	1	2325	2
	2016	2	1919	1
	1786	2	2008	1
Opportunism	1087	0	2061	2
	928	0	1406	1
	1917	0	1078	0
			1322	0

TABLE 1: Quantitative Results

This algorithm has the effect that in the case of commitment robots keep themselves engaged in pursuing an task until it is fixed, while in the case of opportunism, robots keep switching engagement.

6. DISCUSSION

The grid world results are interesting if they actually represent real world system behavior. The fact that the best performing task allocation strategy changes as we vary noise parameters in the grid world implies that it can be very difficult to decide *apriori* which task allocation strategy should be used in a given task for any real world implementation. The quantitative results of the experiments are presented in Table 1. The experiments clearly show that the opportunistic strategy worked significantly better than the commitment-based strategy. This might be because the time to reach a task was significantly larger than the time to complete a task, once a robot was there. This choice of parameters favors opportunism over commitment since the former effectively uses the presence of robots near emergencies by harnessing them immediately. In other regions of the parameter space of the emergency handling task (e.g., where the ratio of time-to-reach-task to time-to-complete-task is small) opportunism might not be as effective. The present study excluded the case where several robots would be required to do a task in a cooperative fashion, a regime in which performance might improve with commitment.

The four task allocation strategies we examined are *extreme*, in that they take into consideration only the complete presence or absence of commitment and coordination in the given context. Arguably, the best strategy for any particular task would most likely be a carefully balanced compromise. However, as stated previously, the goal of this work was not to attempt to find the best strategy (which is necessarily task- and parameter-specific), but rather to gain some insight into task allocation in general. The four strategies we explored provide a reasonable span of strategy space and provide leading insights for further study. In practice, the robot capability ratings can be obtained from the databases. Therefore, one can automatically select appropriate candidate for a given task by using the proposed matching procedure and databases.

7. CONCLUSION

The paper describes an empirical study that sought general guidelines for task allocation strategies in systems of multiple cooperating robots. Four distinct task allocation strategies are identified that aim at studying tradeoffs between commitment and coordination. The data from the simulations show that there is no single strategy that produces best performance in all cases, and that the best task allocation strategy changes as a function of the noise in the system. This result is significant, and shows the need for further investigation of task allocation strategies. The described work is a small step toward the larger goal of principled analysis and synthesis of multi-robot coordination strategies for complex and uncertain domains, such as space exploration. The entire exercise has relevance to real world distributed robotic systems.

8. REFERENCES

1. Goldberg, D. and Mataric, M. J. "*Robust behavior-based control for distributed multi-robot collection tasks*". In T. Balch and L.E. Parker (Eds.) Robot Teams: From Diversity to Polymorphism, 2000

2. Goldberg, D. and Mataric, M.J. "Design *and evaluation of robust behavior-based controllers for distributed multi-robot collection tasks*". USC Institute for Robotics and Intelligent Systems Technical Report IRIS-00-387(2000)

3. D. Goldberg and M. J. Mataric, "*Interference as a tool for designing and evaluation of robust behavior-based controllers*". In Proceedings of the AAAI-97, Providence, Rhode Island, July, 637-642, 1997

4. Parker, L. E.. L-ALLIANCE: "*A Mechanism for adaptive action selection in heterogeneous Multi-Robot teams",* ORNL/TM-13000 (1995)

5. G. Dudek, M. Jenkin, E. Milios, and D. Wilkes. "*A taxonomy for swarm robots*". In Proceedings of the IEEE/RSJ International conference on Intelligent Robotics and Systems, 441-447, 1993

6. D. P. Bertsekas. "Auction algorithms for network flow problems: A tutorial introduction". Computational Optimization and Applications*,* 7-66, (1992)

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

8. M. B. Dias. *TraderBots: "A new paradigm for robust and efficient multirobot coordination in dynamic environments".* Ph.D thesis, Robotics Institute, Carnegie Mellon University, January 2004.

9. B. P. Gerkey and M. J. Mataric. Sold!: "*Auction methods for multi-robot control*". IEEE Transactions on Robotics and Automation Special Issue on multi-robot Systems, 18(5), 2002

10. M. Golfarelli, D. Maio, and S. Rizzi. "*A task-swap negotiation protocol based on the contract net paradigm".* Technical Report 005-97, CSITE (Research Center for Informatics and Telecommunication Systems), University of Bologna (1997)

11. R. Simmons, D. Apfelbaum, W. Burgard, D. Fox, S. Thrun, and H. Younes. "*Coordination for multi-robot exploration and mapping*". In Proceedings of the National Conference on Artificial Intelligence, 2000

12. R. Zlot, A. Stentz, M. B. Dias, and S. Thayer. "*Multi-robot exploration controlled by a market economy*". In Proceedings of the International Conference on Robotics and Automation, 2002

13. R. Aylett and D. Barnes. "*A multi-robot architecture for planetary rovers*". In Proceedings of the 5th ESA Workshop on Advanced Space Technologies for Robotics and Automation, 1998

14. P. Caloud, W. Choi, J.-C. Latombe, C. L. Pape, and M. Yim. "*Indoor automation with many mobile robots*". In Proceedings of the International Workshop on Intelligent Robotics and Systems (IROS), 1990

15. E. Ostergaard, and M.J. Mataric. "*Distributed multi-robot task allocation for emergency handling*". In Proceedings of International Conference on Intelligent Robots and Systems, 2001

16. J. Guerrero and G. Oliver."*Physical interference impact in multi-robot task allocation auction methods*". In Proceedings of IEEE Workshop on Distributed Intelligent Systems. pp.19-24, 2006

17. F.Li Lian and R. Murray. "*Cooperative task planning of multi-robot systems with temporal constraints*", In Proceedings of International Conference on Robotics & Automation, 2003

18. A. Stentz, and M.B. Dias. "*A free market architecture for coordinating multiple robots*". Carnegie Mellon Robotics Institute Tech Report CMU-RI-TR-99-42, December. (1999)

19. M. Shen, G.H.Tzeng and D.R.Liu . "*Multi-criteria task assignment in workflow management systems*". In Proceedings of the 36th Hawaii International Conference on System Sciences, 2002

20. L.E. Parker. "*ALLIANCE: architecture for fault tolerant multi-robot cooperation*". In Proceedings of IEEE Transactions on Robotics and Automation, Vol. 14, No. 2, 220-240, 1998

21. M. G. Lagoudakis, M. Berhault, S. Koenig, P. Keskinocak and A.J**.** Kleywegt, "*Simple auctions with performance guarantees for multi-robot task allocation*". In Proceedings of IEEEIRSI International Conference on Intelligent Robots and Systems, 2004

22. K.Skrzypczyk. "*Game theory based task planning in multi robot systems*". In proceedings of 16th European Simulation Symposium, 2004

23. A. R. Mosteo and Luis Montano."*Simulated annealing for multi-robot hierarchical task allocation with flexible constraints and objective functions*". Workshop on Network Robot Systems: Toward Intelligent Robotic Systems Integrated with Environments". IROS, 2006

24. N. Boneschanscher, “*Task assignment for a small batch flexible manufacturing assembly cell incorporating multiple robots*”. In Proceedings of IEEE, 1990

25. Maes, P. “*Modeling adaptive autonomous agents*”. Artificial Life, I, 1(2),135-162, 1994

26. Mataric, M.J., Sukhatme, G.S., et al. “*Multirobot Task Allocation in uncertain Environment”.*Autonomous Robots, Vol. 14, 255-263, 2003

27. Gerkey,B.and Mataric, M.J. “*Principled communication for dynamic multi-robot task allocation*”. *Experimental Robotics VII, LNCIS 271,* D. Rus and S. Singh (Eds.), Springer-Verlag: Berlin, 353-362,2001

28. M.B.Dias, and A.T.Stentz, “*A free market architecture for distributed control of a multirobot system*”. In Proceedings of the 6th International Conference on Intelligent Autonomous Systems (IAS-6), 115— 122, 2000

29. Corkill, D.D. “*Blackboard systems*”. AI Expert, 6(9), 40-47, 1991

30. B.Werger, and M.Mataric, “*Broadcast of local eligibility for multi-target observation*”. In Proceedings of the 5th International Symposium on Distributed Autonomous Robotic Systems (DARS), Knoxville, TN, Oct. 4-6, 347-356, 2000