Proceedings of the IEEE Sponsored Conference on Computational Intelligence, Control And Computer Vision In Robotics & Automation, March 10-11, 2008, NIT Rourkela, P 99-106

Task Allocation Methodologies for Multi-Robot Systems

B.B.Choudhury and B.B. Biswal

Task Allocation Methodologies for Multi-Robot Systems

B.B.Choudhury and B.B. Biswal

Abstract-- **One of most important aspects in the design of multi-robot systems is the allocation of tasks among the robots in a productive and efficient manner. Task allocation methodologies must ensure that not only the global mission is achieved, but also the tasks are well distributed among the robots. An effective task allocation approach considers the available resources, the entities to optimize (time, energy, quality), the capabilities of the deployable robots, and appropriately allocates the tasks accordingly. This paper presents some important task allocation methodologies for multi-robot systems, providing a review of numerous approaches.**

***Keywords*--Distributed robotics, Multi-robot systems, Task allocation.**

I. Introduction

One of the important recent trends in robotics is the study of teams of multi-robot systems. Research performed under such titles as distributed robotic systems, swarm robotics, sociorobotics, decentralized robotics, multi-agent robotics, and cellular robotics, has focused on the investigation of issues and applications of systems composed of groups of robots. The general idea is that teams of robots, deployed to achieve a common goal, are not only able to perform tasks that a single robot is unable to, but also can outperform systems of individual robot, in terms of efficiency and quality. In addition, groups of robots provide a level of robustness, fault tolerance, and flexibility, as the failure of one robot does not result in the unsuccessfulness of the mission, as long as the remaining robots share the tasks of the failed robot. Examples of tasks appropriate for robot teams are large area surveillance, environmental monitoring, autonomous reconnaissance, large object transportation, planetary exploration, and hazardous waste cleanup.

Applications of robot teams are in four basic areas: handling large objects, covering large areas, performing iterative tasks, providing robustness and fault tolerance. There are a number of certain situations that lends themselves well to the task decomposition and allocation among multiple robots [1].

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 robots performs individual tasks, while the collection of these tasks is toward a unified mission. For example, when multiple robots are engaged in collecting rock samples and returning them to a common place, the team is accomplishing a global mission while cooperating implicitly. This type of group behavior is also called asynchronous cooperation, as it required 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 into tasks to be allocated to the individual robots.

Multi-robot teamwork is a complex problem consisting of task division, task allocation, coordination, and communication. Dudek *et al.* [2] present a general taxonomy to characterize multi-agent systems, consisting of the number of agents, communication, reconfigurability, processing mechanism, and differentiation. Simultaneous self-centered actions of robots do not result in multi-robot system teamwork. These actions must be defined in a framework of system's goal, which unites or separates robots in groups. The reason for robots to function in a group, whether all robots to function in a group, whether all robots have a unique goal like a soccer team or they have multiple goals such as a free market system, the behavior of the group of robots amongst themselves (help or impede one another) are some of the important issues which can be addressed by the proper task allocation mechanism. The focus of this paper is on reviewing different task allocation methodologies for multi-robot systems.

B.B.Choudhury, Ph.D Scholar, (bbcnit@gmail.com)
Prof.B.B. Biswal, Professor, (bbbiswal@nitrkl.ac.in)
Department of Mechanical Engineering,
National Institute of Technology, Rourkela 769008, India

II. Task Allocation

One main issue in task allocation is the division of the tasks into homogeneous versus heterogeneous tasks. Goldberg and Mataric [3] studied homogeneous and heterogeneous task allocation for a foraging task such as trash collection. Their implementation ranged 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 used inference, spatial, and temporal parameters to evaluate different methods. Experimental result showed 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, Sukthanker and Sycara [4] showed that augmenting homogenous task allocation by making robots more team-aware, results in systems that are substantially more efficient.

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. Construction, maintenance, and utilization of large groups of robots have proven to be infeasible due to time and budget requirements. It is too difficult to build a team of 100 robots, make sure that all are functioning and perform experiments with them. Instead, the researchers have been conducting the hardware experiments with only a few robots, and then they have augmented their hardware studies with computer modeling and simulation of robot groups with large populations. 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. In some simulation and analytical studies, the focus is on complex emergent behavior of a collection of simple robots, i.e., collective behavior. These works use mathematics to predict and design working group of robots. Lerman *et al.* [5] propose a mathematical methodology based on viewing large colonies of robots (swarms) as stochastic systems, Markov property, to predict their emergent behavior. Because Markov property holds in many multi-agent systems (e.g. behavior-based or reactive control) this analysis can be useful in many applications. Mathematical analysis has the following capabilities: predicating the collective emergent behavior, and the understanding of effects of each parameter on system.

In terms of applications, certain missions are more suitable for the study of task allocation. Geometric formation is one such application. In geometric formation, a team of mobile robots attempts to achieve and maintain a geometrical shape, while moving toward their target. Some multi-robot missions such as exploration require team formation. For example, army mechanized scout platoon or dynamic radar surface coverage that are based on maintaining a constant distance from one another. This type of problem has been studied by multi-robot system researchers [6,7]. If formation is treated like a coordination problem a static task allocation usually works well. Balch and Arkin [1,6, 8] proposed a method of team formation where the task allocation takes place during system design. In this static approach, all of the robots have a predefined and similar task. This task can be expressed as "while avoiding to bump to any object, including other robots go to target point and meanwhile maintain formation." This work used schema-based architecture [9] to implement motor schema navigation. Perceiving sensed data activates schemas in parallel. These asynchronous processes start behaviors, and the result of these behaviors (a vector format) will be multiplied by an importance weight. The sum of these factors is used to generate a global output for the control of the actuators (e.g. motors in mobile robots). Each robot maintains the formation by calculating its proper position in the group and executing a motor schema to move toward the goal position.

The task allocation methodologies for multi-robot systems are presented in the following sections.

III Task allocation methodologies

A. Functionally-Accurate Cooperative (FA/C) Distributed Problem Solving

In the FA/C Distributed Problem Solving approach [10] each robot has just partial data for solving the imperfect and temporal sub-problems. By considering the issue of reduction of the costly communication, this approach provides a structure to cooperate the interactions among the robots. In this architecture, robots work interdependently, where each robot is aware of its present situation and produces partial and intermediate stage results.

This approach works even in cases where the lack of data has resulted in system inconsistency and uncertainty, through the resolution between interdependent middle data received from the other robots. As a result, this architecture has a very low bandwidth, is more reliable, and increases the effective agent operation time, i.e., less agent idle time. But it uses exhaustive computation. Fig. 1 shows an example of two-agent system.

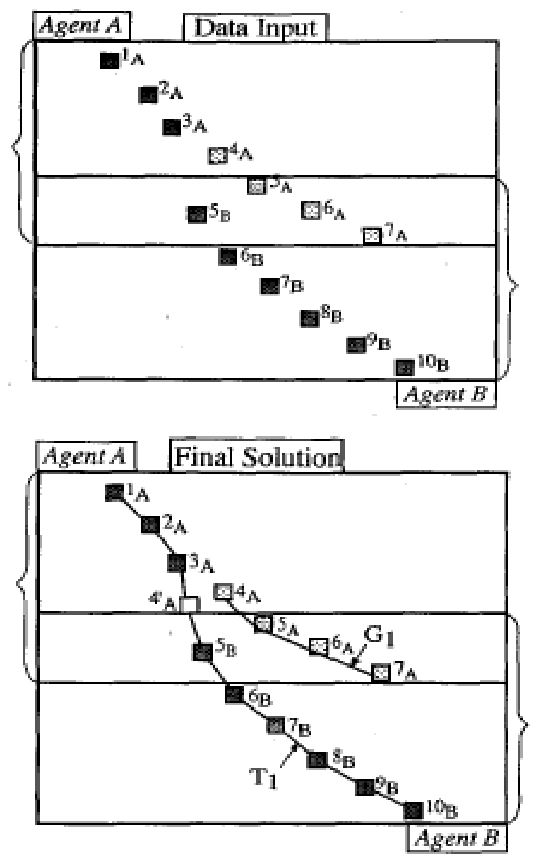
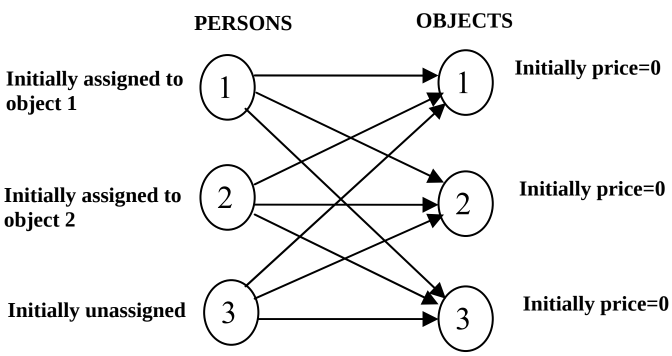


Fig. 1. An example of a two-agent distributed aircraft monitoring scenario

B. Auction Algorithm

Bertsekas[11] 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 in each iteration tries to assign a task to a user who offers the most. In consecutive iterations, other users may bid for other tasks and if more than one bids for the same task, it will increase the cost of task until finally just one task-user pair match takes place, i.e., iterative improvement. The iteration terminates when all users are happy with their match, otherwise an unhappy user will bid higher for another task and this process will continue.

Although auction algorithm may have some similarities to the free market approach, there are a few differences. One difference is that in the free market approach, robots can cooperate in order to gain a maximum profit for all of them, however in the auction algorithm every robot is considered a rival. Another dissimilarity is that the auction algorithm uses a unique 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 robots, while in the auction algorithm the robot set is homogeneous. The auction algorithm is presented in Fig.2.



Here a_{ij} =c>0 for all(i,j) with i=1,2,3 and j=1,2 and a_{ij}=0 for all (i,j) with i=1,2,3 and j=3

Fig. 2. The Auction algorithm

C. Alliance

The Alliance approach [12, 13, 14, 15, 16] 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 just needs to run an Alliance process as a requirement in order to cooperate. Task allocation between different robots with different structures takes place in the Alliance. The Alliance architecture is shown in Fig. 3. The robots communicate explicitly and globally. An extended approach, which incorporates learning, is called L.Alliance [17, 18].

The selection of a suitable action is based on a concept called motivation. Motivation is mathematically modeled with two functions of impatience and acquiescence. Each robot has a partial knowledge of its own and other robots' state. This partial knowledge plus impatience and acquiesce is used to calculate the level of activation as a probability value computed for each action. Impatience happens when a robot perceives that another robot (considering its effect on the environment) has not achieved enough. Acquiesce happens when a robot understands its incapability to complete a task using its sensory feedback.

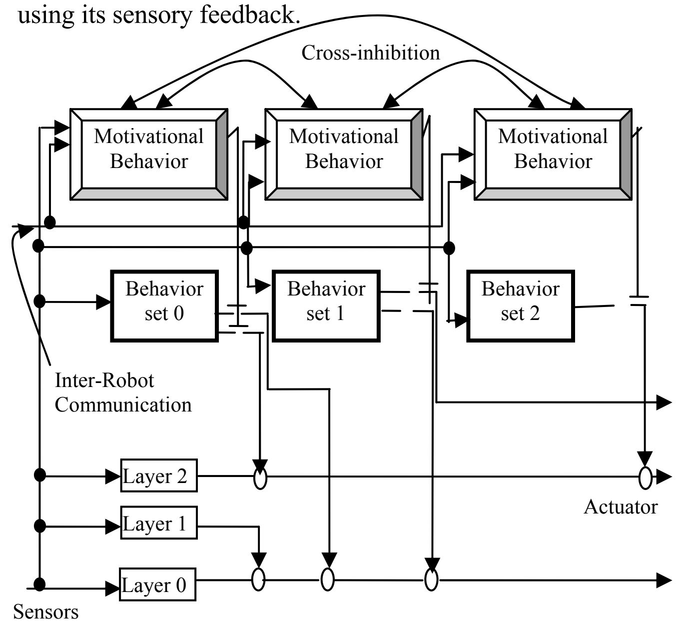


Fig. 3. The alliance architecture

D. Task Acquisition using Multiple Objective Behavior Coordination

Pirjanian and Mataric present a task allocation approach for deliberative behavior-based architecture [19, 20] for multi-robot systems. This methodology uses a behavior-based architecture for single robot control, and a deliberative task planning system for team interaction and task allocation. In this approach the whole system does not have a unique goal but each robot may have its own individual goal. The proposed architecture enables each robot to select its action, and to maximize each robot's achievement while also maximizing the group gain. The concept of optimality may not exist in many situations. Therefore this approach looks for solutions that are just partially optimal. Action selection is performed through voting and global communication among agents. The Multiple Objective Behavior Coordination (MOBC) is the main thesis of this approach and proposes command fusion among robots.

E. A Free Market Architecture for Distributed Control of a Multi-Robot Systems

Stenz and Dias [21] 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. This approach is based on free market, i.e., the opposite to the central market system. This technique manages the robots as economical entities. Each individual robot acts based on its own benefit and is self-centered. This results a good group performance, where separate profits are added and thus generate the total team profit. The robots may cooperate in order to improve the overall benefits.

The system's performance is measured based on the revenue/cost balance. The ultimate goal is to maximize the value of the revenue minus the cost. The functions for defining the revenue and the cost for the team, and also the method for task and related cost distribution among robots must be determined. A robot can gain revenue by contributing to the team's goal and also through trading services and goods with other robots.

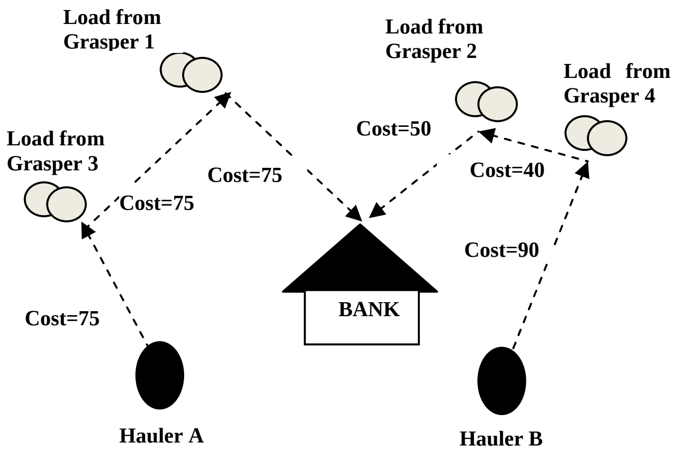


Fig. 4. The winning TSP tour from robot A.

The core of this approach is based on two functions. One function maps the result of each action to a revenue value and a second function that maps each method for performing a single task to its cost values. The calculation of the minimum value of the difference between the two functions results in a factor for selecting the most suitable task. For each given application, the human user must customize and change these two functions based on provided requirements. An example of the problem is presented in Fig. 4.

F. Team Formation-Based Task Allocation

Stone and Veloso [22] use a dynamic task allocation, targeted for the robot soccer application. In this approach all homogeneous robots are set to function with a predefined strategy and have predefined tasks. Later on, a robot uses its perception of the world, and can decide to change its tasks. Because of the changes in the robot's internal state, its external behavior will change and finally its effect on environment will change. This yields new task requirement for other robots in environment. Each change in inner state of a robot is communicated to other robots in order to generate a new team formation. A global goal and a set of tasks are assigned to the robot team in certain periods, such as half times in soccer.

G. Murdoch: Publish/Subscribe System

Gerkey and Mataric present Murdoch [23,24], a dynamic task allocation mechanism using a communication method called publish/subscribe for performing distributed control and multi-robot coordination. The proposed system is successfully tested on both tightly coupled and loosely coupled systems. The whole system is seen as a collection of resources that must be assigned to tasks. Because of the uncertainty associated with the status of each robot, task assignments do not address robots directly. Instead, the technique uses publish/subscribe paradigm, which provides anonymous task-robot sets. Each robot subscribes to a set of tasks based on resources that it can deploy. Robots can subscribe to different message lists based on their capabilities (e.g., mobile, sonar, vision, etc.), and based on current state information (e.g., seeing an object, pushing one, or so on). The subjects of the message could be the robot's capability or its status.

Murdoch declares and defines different tasks based on the tasks' subjects, referred to as subject-base addressing. These include the robot's capabilities (ownership of resources for a task) or robot status (e.g., the robot's energy). The system then publishes that message, addressed by the content, instead of their targets in entire network. Robots who subscribed to that subject will receive the associated message. Thus tasks are divided at the behavior abstraction level instead of robot abstraction level. For instance, a task requiring sonar, laser, and vision publishes using the tuple of sonar laser camera as shown in Fig. 5 to push the box along the desired trajectory.

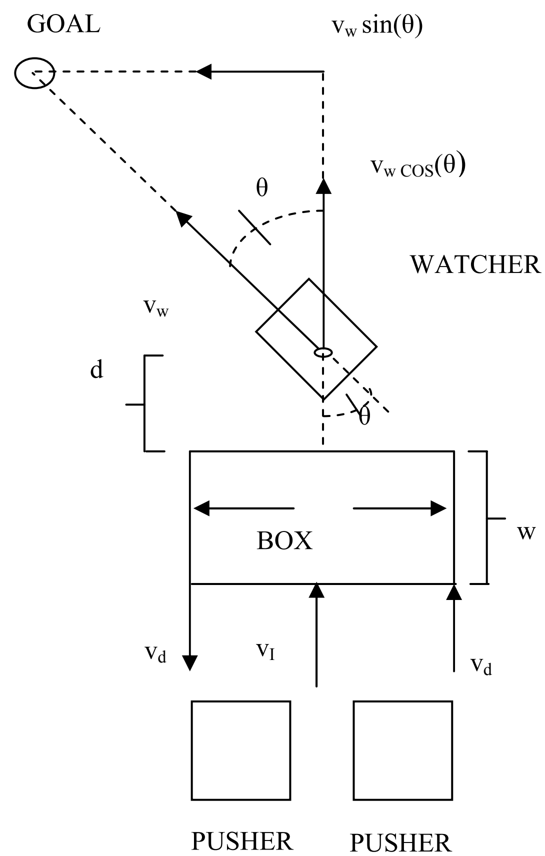


Fig. 5. The model used to derive the pushing velocities for moving the box along the desired trajectory

A best-fit selection algorithm is used to choose the best among robots that are registered for a particular subject. The human user or another component of the system must perform task decomposition. Each task is accompanied with a metric as a measure of fitness. This metric is application-dependant and can be related to the robot's state or other computation. Afterwards, each registered robot measures its own fitness based on the metric and communicates the score *to the others.* The winner gains a time limit within which to accomplish the given task.

H. Broadcast of Local Eligibility (BLE) using Port Arbitration Behavior (PAB)

Werger and Mataric [25, 26, and 27] present a task allocation methodology based on calculating the local efficiency of a robot for a task, and then communicating it with other robots. The most efficient robot inhibits others and performs the task. This approach is based on behavior-based control, and uses a technique called Port Arbitration Behavior (PAB), which is an architecture for conflict resolution among robots. PAB uses a collection of behavior production modules (BPM), which are the programmed code that produce a robot behavior. Each BPM can be considered as a control software component. Each of these components has an interface port that is accessible from the other components. These ports can be connected to each other with unidirectional communication paths (called connections).

In multi-robot applications the separation of BPMs from the connections is advantageous in terms of the code reusability. Therefore in different coordination methods, only the connections attributes must be changed.

Each BPM computes its own local eligibility and send it to all other robots. This communication is very simple. Each robot then sends its calculated eligibility value to all other robots. Each robot compares its computed local eligibility in relation to others. If one recognizes that it has the best eligibility, then it produces the desired behavior and inhibits the other robots' behaviors. This is achieved by inhibiting the others' port via the connection between them. One issue of concern for the designers is in finding a function to compute local eligibility for any given application. The function must be at least partially dependant on the sensor outputs. The scheme of the process is shown in Fig. 6.

Initial

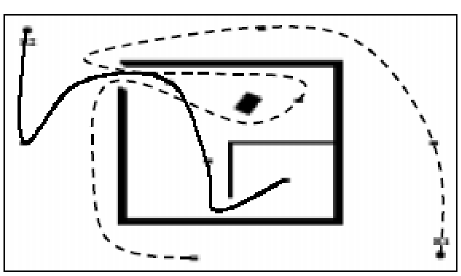


Fig. 6. Initial assignments and final tours for 2 robots and 8 cities

I. Distributed Multi-Robot Task Allocation for Emergency Handling

Ostergaard and Mataric [28] propose an algorithm for task allocation that assigns dynamically to each "emergency" situation a suitable and capable robot to handle it. Task allocation is dynamic and happens on a needed basis. The assignment of tasks to the robots are done based on two factors. The first factor is the commitment, defined based on whether a robot should finish its assigned task or should move on to the next more beneficial task. The second factor is coordination, defined as the awareness of other robots and whether or not to communicate.

Task allocation utilizes a robot as a blackboard upon which the rest of the robots write to and read from. Each robot reads data from the blackboard often and if the intensity of the signal that was sent by a goal and received by it is at the maximum value, it then selects that task and starts running it. This approach assumes that if all robots have same information, then they collectively choose the most efficient task for system performance. Thereafter they update the blackboard.

This distributed task selection functions based on selecting the task which sends the maximum intensity of the sensed signals.

Definition 1: The Emergency Handling
Task consists of:

- An environment, E
- A set of robots, R
- A set of alarms, A
- A set of tools, T
- A capability function, c: $R \rightarrow T$
- A requirement function, $s : A \rightarrow T$

One robot can carry $|c(r_j)|$ tools, where $0 \leq |c(r_j)| < |T|, 0 \leq i < |R|$.

Each alarm can require $|S\ (a_i)|, 0 < |s(a_i)| \leq min(|R|,|T), 0 < i < |A|$ tools to be fixed. We require that all alarms can be handled with one or more of the available tools. Robots are heterogeneous if they are equipped with different tools or have different capabilities.

Otherwise, the robots are homogeneous.

Fig. 7. The emergency handling

J. Ants Algorithms

The basic idea of Ants algorithm [29, 30] is based on adaptability of groups of ants to their environment changes. The method is based on some biologic facts about ants, where they leave some amount of pheromone on their trail, and they prefer to follow the paths with most pheromone on it. This approach can be considered as task allocation, since each path/trail can be thought of as a task which must be selected with a probability function. This methodology is based on a few assumptions, including the fact that ants walk in a direct path, moving in a two-dimensional dimension. Another assumption is that when a group of ants encounters an obstacle, they divide into two equal sub-groups. An important feature of this approach is the indirect communication between ants, resulting in emergent behavior.

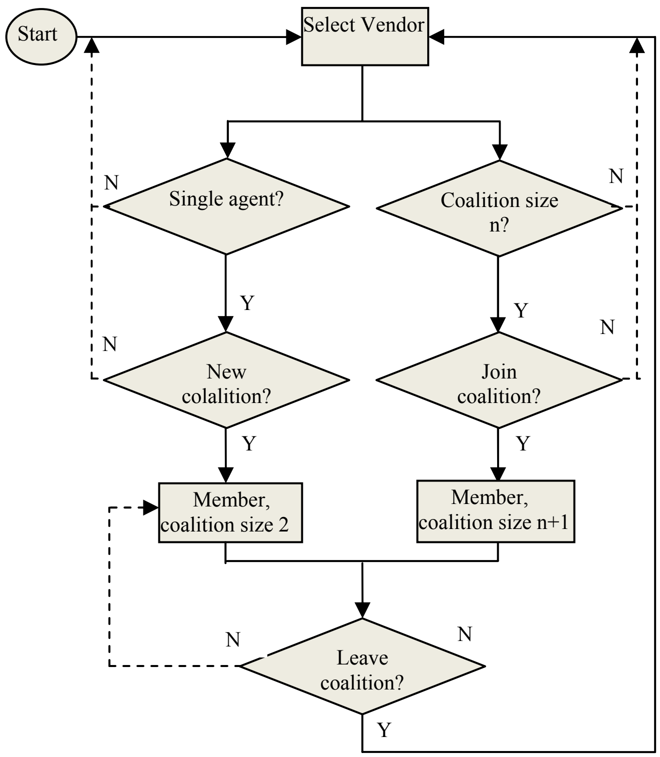


Fig. 8. Single agent controller.

K. Task Allocation in uncertain environment

Multiple cooperating robots hold the promise of improved performance and increased fault tolerance for large-scale problems such as planetary survey and habitat construction. Multi-robot coordination, however, is a complex problem. This problem in the framework of multi-robot dynamic task allocation under uncertainty and describe an empirical study that sought general guidelines for task allocation strategies in multi-robot systems. MATARIC et.al[30] identified four distinct task allocation strategies, and demonstrate them in two versions of the multirobot emergency handling task. In this paper describe an experimental setup to compare results obtained from a simulated gridworld to those obtained from physical mobile robot experiments. Data resulting from eight hours of experiments with multiple mobile robots are compared to the trend identified in simulation. 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 and their application to planetary exploration.

L. Robot exploration with combinatorial auctions

Berhauld , Huang and Keskinocaki [31] propose 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. The results of Team Bots, 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.

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.

M. Cooperative task planning of multi-robot systems with temporal constraints

Lian and Murray [32] 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.

- A single robot is tasking from the home base position to the target position. The target position and the designated action at the position is simply instructed by an upper-level command unit.

- In the second case, three robots might he instructed by the same activity command, and need to move together in a designated formation. Hence, the controller at each individual robot should generate a set of feasible, real-time trajectories which guarantee the group of robot to move in the designated formation.
- The third case considers a more general scenario where multiple robots from different home bases are commanded to either one common target or multiple targets. At some location, these robots are commanded to move together and have a certain level of formation interaction. Conceptually, this scenario can be viewed as a combination of the first two cases.

For a given system dynamics and a set of state and input constraints, and to minimize a pre-specified cost function, the NTG algorithm first makes use of the differential flatness property to find a new set of outputs in a lower dimensional space and then parameterizes the outputs by the B-spline basis representation.

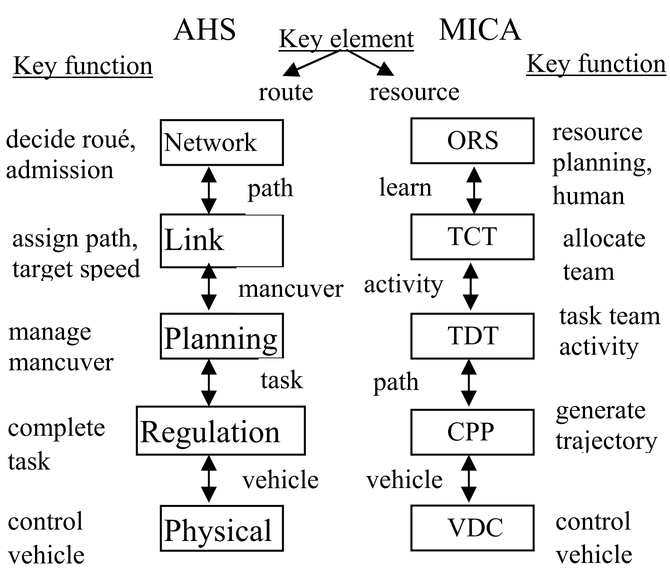


Fig. 9. The AHS and MICA hierarchies and their key elements and functions.

N. Integer programming for combinatorial auction winner determination

Andersson,Tenhunen and Ygge[33] propose on Combinatorial auctions are important as they enable bidders to place bids on combinations of items; compared to other auction mechanisms, they often increase the efficiency of the auction, while keeping risks for bidders low. However, the determination of an optimal winner combination in combinatorial auctions is a complex computational problem. In this paper we (i) compare recent algorithms for winner determination to traditional algorithms, (ii) present and benchmark a mixed integer programming approach to the problem, which enables very general auctions to be treated efficiently by standard integer programming algorithms and (iii) discuss the impact of the probability distributions chosen for benchmarking.

O. Physical interference impact in multi-robot task allocation auction methods

Guerrero and Oliver [34] present Task allocation is one of the main problems in multirobot systems. Among other factors, to get a good task allocation, we have to take into account the physical interference effects between robots, that is, when two or more robots want to access to the same point at the same time. This paper analyzes interference impact using auction methods, one of the most popular task allocation systems. This paper shows how the performance of the auction utility function can be improved if interference impact is included in it. We also provide a framework to simplify one of the main problems of all auction systems which is finding a good utility function.

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

Also, in general, utility functions are not linear, so the learning process can be very hard. To simplify the process, some parameters can be analyzed previously, using an ideal environment, and then modified during the execution of the task in 3 steps:

Individual utility: during the first stage, we evaluate the characteristics of each single robot without taking into account the others. Here it will be include some characteristics like velocity, acceleration, etc.

Group utility: in this step, the robot will take into account the other ones to create a coalition or working group. Here some parameters, like interference effect, will be included. That is, the robots will calculate the utility function of the group.

Inter-Group utility: finally, the robots have to take into account that the decision of one group can affect to other groups. This inter-group dependency must be included in the utility function during the final step.

IV. Conclusion

Although numerous important results have been obtained by the researchers in the area of multi-robot systems, a great deal of work remains to be done in order for the group behavior of robots to be fully understood and utilized in real world applications. The concept of task allocation remains an essential component of this challenge. A survey of this field was included in this paper. Productive, efficient, and dynamic approaches to allocating tasks among robots will result in further utilization of multi-robot systems.

Task allocation and decomposition methodologies will serve as design guidelines to allow multi-robot systems gain efficiency. It is important to invest time to understand different methodologies before applying them to real world applications. We believe that a comprehensive and integrated survey will help accelerate this understanding. The intent of this article was to provide readers with a global perspective on the research literature on multi-agent task allocation systems.

V. References

[1] R.C.Arkin and T. Balch, "Cooperative multiagent robotic systems", In D. Kortenkamp, R.P. Bonasso, and R. Murphy (Eds.) *Artificial Intelligence and Mobile Robots: Case Studies of Successful Robot Systems,* AAAI Press/The MIT Press, Cambridge, MA, 277-296, 1998.

[2] 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.

[3] D. Goldberg and M.J. Mataric, "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.

[4] L.E. Parker. "L-ALLIANCE: A Mechanism for Adaptive Action Selection in Heterogeneous Multi-Robot Teams", *ORNL/TM-13000,* 1995.

[5] M. Dorigo, V. Maniezzo, and A. Colorni, "The Ant System: Optimization by a Colony of Cooperating Agents", *IEEE Transactions on Systems, Man, and Cybernetics,* Part B, vol. 26, No. 1,29-41,1996.

[6] T. Balch and R.C. Arkin, " Behavior-based formation control for multirobot teams", *IEEE Transactions on Robotics and Automation,* December, Vol. 14, 926-939,1998.

[7] J. Fredslund and M.J. Mataric, "A general, local algorithm for robot formation", In *Proceedings of IEEE International Symposium on Computational Intelligence for Robotics and Automation,* Banff, Canada, 2001.

[8] B.P. Gerkey and M.J. Mataric, "Murdoch: Publish/Subscribe Task Allocation For Heterogeneous Agents", In *Proceedings of the Fourth International Conference on Autonomous Agents,* ACM Press, Barcelona, Spain, 203-204, 2000.

[9] R.C. Arkin, "Motor schema based mobile robot navigation", *International Journal of Robotics Research,* Vol. 8, No. 4, 92-112, 1989.

[10] V.R. Lesser, "A retrospective view of FA/C distributed problem solving", *IEEE Transactions on Systems, Man, and Cybernetics,* Vol. 21, 1347-1362, 1991.

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

[12] L.E. Parker, " ALLIANCE: an architecture for fault tolerant multi-robot cooperation", *IEEE Transactions on Robotics and Automation,* Vol. 14, No. 2, 220-240, 1998.

[13] L.E. Parker," On the design of behavior-based multi-robot teams", *Journal of Advanced Robotics,* Vol. 10, No. 6,1996.

[14] L.E. Parker, "Designing control laws for cooperative agent teams", In *Proceedings of the IEEE International Conference on Robotics and Automation,* 582-587,1993.

[15] L.E. Parker, " Evaluating success in autonomous multi-robot teams: experiences from ALLIANCE architecture implementations", *Journal of Theoretical and Experimental Artificial Intelligence,* Vol. 13, 95-98, 2001.

[16] L.E. Parker, "Cooperative motion control for multi-target observation", In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems,* 1591-1598, 1997.

[17] L.E. Parker, " L-ALLIANCE: task-oriented multi-robot learning in behavior-based systems", *Advanced Robotics, Special Issue on Selected Papers from IROS'96*, Vol. 11, No.4, 305-322, 1997.

[18] L.E. Parker, "L-ALLIANCE: A Mechanism for Adaptive Action Selection in Heterogeneous Multi-Robot Teams", ORNL/TM-13000, 1995.

[19] P. Pirjanian, "Multiple objective action selection in behavior-based control", In *Proceedings of the 6th Symposium for Intelligent Robotic Systems,* Edinburgh, Scotland, July, 83-92,(1998.

[20] P. Pirjanian and H.I. Christensen, "Satisfying action selection", In *Proceedings of the SPIE Conference on Intelligent Systems and Advanced Manufacturing,* Boston, Massachusetts, November, 153-164, 1998.

[21] 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.

[22] P. Stone and M. Veloso, " Task decomposition, dynamic role assignment, and low-bandwidth communication for real-time strategic teamwork", *Artificial Intelligence,* Vol. 110, No. 2, June, 241-273, 1999.

[23] B.P. Gerkey and M.J. Mataric., "Pusher-watcher: an approach to fault-tolerant tightly-coupled robot coordination", In *Proceedings of IEEE/RSJ IROS,* 2001.

[24] B.P. Gerkey and M.J. Mataric, "Sold!: Market methods for multi-robot control",IEEE transactions on Robotics and Automation, Special Issue on Multi Robot Systems, 2001.

[25] B.B. Werger and M.J. Mataric, "Broadcast of local eligibility for multi-target observation", In *Proceedings ofDARS 4,* 347-356, 2000.

[26] B. Werger and M.J. Mataric, "From insect to internet: Situated control for networked robot teams",*Annals of Mathematics and Artificial Intelligence,* Vol. 31, No. 1-4, 173-198, 2001.

[27] M. B. Dias, and A. Stentz, "A Free Market Architecture for Distributed Control of a Multi-Robot System", In *Proceedings of the 6th International Conference on Intelligent Autonomous Systems,* July 2000.

[28] E. Ostergaard, and M.J. Mataric, "Distributed multi-robot task allocation for emergency handling", In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems,* Maui, Hawaii, 2001.

[29] K. Lerman and A. Gastyan , "A general methodology for mathematical analysis of multi-agent systems", *USC Information Sciences,* Technical Report ISI-TR-529, 2001.

[30] M.J. Matric, G. S. Sukhatme and E. H. Østergaard., "Multi-Robot Task Allocation in Uncertain Environments", Autonomous Robots 14, 255–263, 2003

[31] Berhauld , Huang and Keskinocaki, "Robot Exploration with Combinatorial Auctions*", In Proc. of Int.Conf. on Intelligent Robots and Systems* ,vol.2,pp.1957-1962, 2003.

[32] F.Li Lian and R. Murray, "Cooperative Task Planning of Multi-Robot Systems with Temporal Constraints*", In Proceedings. of International Conference on Robotics &Automation.,*pp-14-19,2003.

[33]. A. Andersson,M. Tenhunen and F.Ygge.(2000) Integer Programming for Combinatorial Auction Winner Determination. Proc. of Fourth International Conference on Multiagent Systems .pp.39-46.

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