TASK ALLOCATION METHODOLOGIES FOR MULTI-ROBOT SYSTEMS

Khashayar R. Baghaei

Arvin Agah

Technical Report ITTC-FY2003-TR-20272-01

November 2002

Information and Telecommunication Technology Center

The University of Kansas, Lawrence, Kansas 66045

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 ten such task allocation methodologies for multi-robot systems, providing a review of numerous approaches.

KEYWORDS

Multi-robot systems, task allocation, distributed robotics.

1. 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: large objects must be handled, large areas must be covered, iterative tasks must be performed, and robustness and fault tolerance is required. In addition, the study of multi-robot systems can help in better understanding of social sciences, life sciences and cognitive sciences [1]. There are a number of certain situations that lends themselves well to the task decomposition and allocation among multiple robots [2].

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.* [29] present a general taxonomy to characterize multi-agent systems, consisting of the number of agents, communication (range, bandwidth and topology), 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. What is the reason for robots to function in a group? Do all robots have a unique goal like a soccer team or do they have a multiple goals such as a free market system? Do robots act in a self-centered manner or as team-aware individuals? How do the robots help or impede one another? All these issues 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.

2. TASK ALLOCATION

One main issue in task allocation is the division of the tasks into homogeneous versus heterogeneous tasks. Goldberg and Mataric [24, 25, 26] 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 [32] 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.* [37,38] 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 [3,28]. If formation is treated like a coordination problem a static task allocation usually works well. Balch and Arkin [2, 3, 5] 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 [4] 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 next section.

3. TASK ALLOCATION METHODOLOGIES

3.1 Murdoch: Publish/Subscribe System

Gerkey and Mataric present Murdoch [6, 7, 8], 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).

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.

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

Werger and Mataric [9, 10, 11] 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 advantages 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.

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

Stenz and Dias [12] 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 but this individuality. 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.

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.

3.4 Auction Algorithm

Berstas [22] 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.

3.5 Alliance

The Alliance approach [13, 14, 15, 16, 17, 18] 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 robots communicate explicitly and globally. An extended approach, which incorporates learning, is called L_Alliance [31, 32].

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.

3.6 Task Acquisition using Multiple Objective Behavior Coordination

Pirjanian and Mataric present a task allocation approach for deliberative behavior-based architecture [19, 20, 21] 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.

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

In the FA/C Distributed Problem Solving approach [33] 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.

3.8 Distributed Multi-Robot Task Allocation for Emergency Handling

Ostergaard and Mataric [23] 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.

3.9 Team Formation-Based Task Allocation

Stone and Veloso [27] 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.

3.10 Ants Algorithms

The basic idea of Ants algorithm [35, 36] 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.

4. 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.

ACKNOWLEDGEMENTS

This work was sponsored in part by a grant from the Kansas NASA EPSCoR Program (KNEP), entitled: “An autonomous mobile radar for temperate ice thickness measurements.”

REFERENCES

[1] Y.U. Cao, A.S. Fukunaga, A.B. Kahng, and F. Meng. (1995) Cooperative mobile robotics: Antecedents and directions. In *Proceedings of IEEE/RSJ IROS*, Pittsburgh, PA, Vol. 1, 226-234.

[2] R.C. Arkin and T. Balch. (1998). 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.

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

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

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

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

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

[8] B.B. Werger and M.J. Mataric (2000). Broadcast of local eligibility: behavior-based control for strongly cooperative robot teams. In *Proceedings of the Fourth International Conference on Autonomous Agents*, CM Press, 21-22.

[9] B.B. Werger and M.J. Mataric. (2000). Broadcast of local eligibility for multi-target observation. In *Proceedings of DARS 4*, 347-356.

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

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

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

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

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

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

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

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

[18] L.E. Parker. (1994). An experiment in mobile robotic cooperation. In *Proceedings of the ASCE Specialty Conference on Robotics for Challenging Environments*, February, 131-139.

[19] P. Pirjanian (1998). Multiple objective action selection & behavior fusion using voting. Ph.D. Dissertation, *Department of Medical Informatics and Image Analysis, Aalborg University*, Denmark.

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

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

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

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

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

[25] D. Goldberg and M.J. Mataric. (2000). 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.

[26] D. Goldberg and M.J. Mataric. (1997). 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.

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

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

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

[30] G. Sukthanker and K. Sycara. (2001). Team-aware robotic deminig agents for military simulation. http://www-2.cs.cmu.edu/~softagents/iaai00/iaai00.html.

[31] L.E. Parker. (1997). 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.

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

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

[34] O. Shehory and S. Kraus. (1998). Methods for task allocation via agent coalition formation. *Artificial Intelligence*, Vol. 101, No. 1-2, 165-200.

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

[36] K. Lerman. (2000). Design and mathematical analysis of agent-based systems. *Proceedings of the Formal Methods Approaches for Agent-based Systems*, to appear.

[37] M. Dorigo and L.M. Gambardella. (1996). Ant colonies for the traveling salesman problem. *Universit'e Libre de Bruxelles, Belgium*, IRIDIA, Technical Report TR/IRIDIA/1996-3.

[38] M. Dorigo, V. Maniezzo, and A. Colorni. (1996). 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.