To appear in *Proceedings of the NRL Workshop on Multi-Robot Systems*
Washington, DC, March 17 - 19, 2003

A FRAMEWORK FOR STUDYING MULTI-ROBOT TASK ALLOCATION

Brian P. Gerkey, Maja J Matarić
Computer Science Department
University of Southern California
Los Angeles, California 90089-0781, USA
bgerkey@cs.usc.edu, mataric@cs.usc.edu

Abstract Although many multi-robot task allocation (MRTA) architectures can be found in the literature, relatively little has been said regarding the fundamental theoretical characteristics of the task allocation problem. We present a formal, but practical, framework for studying MRTA. In constructing our framework, we borrow from the Operations Research community and show that MRTA can be understood as an instance of the Optimal Assignment Problem. We use this framework to analyze several recently proposed approaches to MRTA, describing their fundamental characteristics in such a way that they can be objectively studied, compared, and evaluated. In so doing, we demonstrate the utility of such frameworks in formalizing robotics research, which we argue is vital to the development of the field.

1. Introduction

Over the past decade, a significant shift of focus has occurred in the field of mobile robotics as researchers have begun to investigate problems involving multiple, rather than single, robots. From early work on simple loosely-coupled tasks such as foraging (Arkin et al., 1993) to recent work on sophisticated team skills for robot soccer (Stone and Veloso, 1999), the complexity of the multi-robot systems being studied has increased. This complexity has two primary sources: larger team sizes and greater heterogeneity of robots and tasks. As significant achievements have been made along these axes, the bar has been raised; it is no longer a sufficient demonstration of multi-robot coordination to show, for example, only two robots observing targets (Parker, 1999), or a large group of robots only flocking (Matarić, 1995). Rather today we reasonably expect to see larger and larger robot teams engaged in concurrent and diverse tasks over extended periods of time. As if to underscore this point, the validation task for the current DARPA Software for Distributed

Robots program (SDR-II) involves deploying 100 robots to achieve a complex multi-faceted task in an unknown environment, possibly over a twenty-four-hour period.

As a result of this focus on multi-robot systems, multi-robot coordination has received significant attention. In particular multi-robot task allocation (MRTA) has recently risen to prominence. Originally a side-show to other problems, MRTA has now become a key research issue in its own right. As researchers design, build, and use cooperative multi-robot systems, they invariably encounter the question: "which robot should execute which task?" This question must be answered, even for relatively simple multi-robot systems, and the importance of task allocation grows with the complexity, in size and capability, of the system under study. Yet the empirically validated methods remain primarily *ad hoc* in nature, and relatively little has been said regarding the general properties of cooperative multi-robot systems. After a decade of research, while countless such architectures have been proposed, we lack even a primitive prescription for how to design a MRTA system.

This dearth of formal work on multi-robot coordination is, in part, to be expected. A new field, such as Robotics, generally begins as an experimental science. It is not until after sufficient collective experience that one can begin to analyze the assembled evidence in order to find general trends and make formal statements describing and predicting system behavior. We suggest that research in multi-robot coordination has reached this point and that there is now the potential to move the field from a primarily experimental science to a more formal, analytical one.

Of course, we have already seen one incarnation of Robotics as a formal undertaking in the time of so-called Good Old-Fashioned Artificial Intelligence (Russell and Norvig, 1995). Proponents of this paradigm suggested that we could program robots by having them build abstract symbolic models of their environments and solve problems by reasoning from first principles about that model. Then system designers could, for example, prove theorems regarding their robots' behavior. Unfortunately this way of thinking, which brought unparalleled success with tasks like chess and checkers, has been a spectacular failure in *embodied* domains, including Robotics, and we are not suggesting a return to it. Rather, we propose to follow in the footsteps of the natural sciences: accrue experimental evidence regarding some system or phenomenon until one can propose a plausible descriptive model. Such a model will of course not be perfect, as it will not capture every salient detail of system. However, because it is constructed *from the experimental evidence*, such an imperfect model, like the laws of Newtonian physics, will still be a powerful tool for describing multi-robot systems and provide a common framework in which to study them.

There are many ways in which one could build this kind of model or framework. In this paper we present a particular framework for studying MRTA, based on the Optimal Assignment Problem, that we have developed by borrowing and adapting ideas and tools that come originally from the field of Operations Research. We have described some parts of this framework elsewhere (Gerkey and Matarić, 2002a); we revisit the salient aspects here, but omit some technical details for brevity and clarity. The rest of this paper is organized as follows. In Section 2 we describe our formal framework. We apply the framework to the evaluate proposed task allocation architectures in Section 3. We address limitations of our approach in Section 4 and conclude in Section 5.

2. Framework

When studying the problem of multi-robot task allocation we take inspiration from Operations Research, a field that concerns itself with human organizations. In particular we claim that multi-robot task allocation can be reduced to an instance of the *Optimal Assignment Problem* (OAP) (Gale, 1960), a well-known problem from Operations Research. A recurring special case of particular interest in several fields of study, this problem can be formulated in many ways. Given our application domain, it is fitting to describe the problem in terms of jobs and workers. There are n workers, each looking for one job, and m available jobs, each requiring one worker. The jobs can be of different priorities, meaning that it is more important to fill some jobs than others. Each worker has a nonnegative skill rating estimating his/her performance for each potential job (if a worker is incapable of undertaking a job, then the worker is assigned a rating of zero for that job). The problem is to assign workers to jobs in order to maximize the overall expected performance, taking into account the priorities of the jobs and the skill ratings of the workers. This problem was first formally studied in the context of assigning naval personnel to jobs based on the results of aptitude tests (Thorndike, 1950).

Our multi-robot task allocation problem can be posed as an assignment problem in the following way: given n robots, m prioritized (i.e., weighted) single-robot tasks, and estimates of how well each robot can be expected to perform each task, assign robots to tasks so as maximize overall expected performance. However, because the problem of task allocation is a dynamic decision problem that varies in time with phenomena including environmental changes, we cannot be content with this static assignment problem. Thus we complete our reduction by iteratively solving the static assignment problem over time.

Of course, the cost of running the assignment algorithm must be taken into account. At one extreme, a costless algorithm can be executed arbitrarily fast, ensuring an efficient assignment over time. At the other extreme, an expensive algorithm that can only be executed once will produce a static assignment that is only initially efficient and will degrade over time. Finally there is the question of how many tasks are considered for (re)assignment at each iteration. In order to create and maintain an efficient allocation, the assignment algorithm must consider (and potentially reassign) every task in the system. Such an inclusive approach can be computationally expensive and, indeed, some implemented approaches to MRTA use heuristics to determine a subset of tasks that will be considered in a particular iteration.

Together, the cost of the static algorithm, the frequency with which it is executed, and the manner in which tasks are considered for (re)assignment will determine the overall computational and communication overhead of the system, as well as the solution quality. Thus it is these characteristics of MRTA architectures in which we are interested. Before continuing with a formal statement of our problem, we undertake a necessary aside regarding *utility*.

2.1 Utility

Utility is a unifying, if sometimes implicit, concept in Economics (Edgeworth, 1967), Game Theory (Neumann and Morgenstern, 1964), and Operations Research (Bertsekas, 1990), as well as multi-robot coordination. The idea is that each individual can somehow internally estimate the value (or the cost) of executing an action. It is variously called fitness, valuation, and cost. Within multi-robot research, the formulation of utility can vary from sophisticated planner-based methods (Botelho and Alami, 1999) to simple sensor-based metrics (Gerkey and Matarić, 2002b). We posit that utility estimation of this kind is carried out somewhere in every autonomous task allocation system.

Regardless of the method used for calculation, the robots' utility estimates will be inexact for a number of reasons, including sensor noise, general uncertainty, and environmental change. These unavoidable characteristics of the multi-robot domain will necessarily limit the efficiency with which coordination can be achieved. We treat this limit as exogenous, on the assumption that lower-level robot control has already been made as reliable, robust, and precise as possible and thus that we are incapable of improving it. When we discuss "optimal" allocation solutions, we mean "optimal" in the sense that, given the union of all information available in the system (with the concomitant noise, uncertainty, and inaccuracy), it is impossible to construct a solution with higher overall utility; this notion of optimality is analogous to optimal scheduling (Dertouzos and Mok, 1983).

It is important to note that utility is an extremely flexible measure of fitness, into which arbitrary computation can be placed. The only constraint on utility estimators is that they must each produce a single scalar value such that they can be compared for the propose of ordering candidates for tasks. For example, if the metric for a particular task is distance to a location and the robots involved employ a probabilistic localization mechanism, then one reasonable utility estimation would be to calculate the center of mass of the current probability distribution. Other mechanisms, such as planning and learning, can likewise be incorporated into utility estimation. No matter the domain, it is vital that *all* relevant aspects of the state of the robots and their environment be included in the utility calculation. Signals that are left out of this calculation but are taken into consideration when evaluating overall system performance are what economists refer to as *externalities* (Simon, 2001) and their effects can be detrimental, if not catastrophic.

2.2 Formalism

We are now ready to state our MRTA problem as an instance of the OAP. Formally, we are given:

- the set of n robots, denoted $I_1, \ldots, I_n$
- the set of m prioritized tasks, denoted $J_1, \ldots, J_m$ and their relative weights $w_1, \ldots, w_m$
- U_{ij}, the nonnegative utility of robot I_i for task J_j, $1 \leq i \leq n$, $1 \leq j \leq m$

We assume:

- Each robot I_i is capable of executing at most one task at any given time.
- Each task J_j requires exactly one robot to execute it.

These assumptions, though somewhat restrictive, are necessary in order to reduce MRTA to the classical OAP, which is given in terms of single-worker jobs and single-job workers. We address the question of relaxing these assumptions in Section 4. It is worth noting that in most existing MRTA work (including the architectures that we study in Section 3), these same assumptions are made, though often implicitly.

The problem is to find an optimal allocation of robots to tasks. An allocation is a set of robot-task pairs:

$$(i_1, j_1) \ldots (i_k, j_k),\ 1 <= k <= \min(m, n)$$

Given our assumptions, for an allocation to be *feasible* the robots $i_1 \ldots i_k$ and the tasks $j_1 \ldots j_k$ must be unique. The benefit (i.e., expected performance) of an allocation is the weighted utility sum:

$$U = \sum_{m=1}^{k} U_{i_m j_m} w_{j_m}$$

We can now cast our problem as an integral linear program (Gale, 1960): find n^2 nonnegative integers α_{ij} that maximize

$$\sum_{i,j} \alpha_{ij} U_{ij} w_j \tag{1}$$

subject to

$$\begin{aligned} \sum_{i} \alpha_{ij} &= 1, \quad \forall j \\ \sum_{j} \alpha_{ij} &= 1, \quad \forall i \end{aligned} \tag{2}$$

The sum (1) is just the overall system utility, while (2) enforces the constraint that we are working with single-robot tasks and single-task robots (note that since α_{ij} are integers they must all be either 0 or 1). Given an optimal solution to this problem (i.e., a set of integers α_{ij} that maximizes (1) subject to (2)), we construct an optimal task allocation by assigning robot i to task j only when $\alpha_{ij} = 1$.

By creating a *linear* program, we restrict the space of task allocation problems that we can model in one way: the function to be maximized (1) must be linear. Importantly, there is no such restriction on the manner in which the components of that function are derived. That is, individual utilities can be computed in any arbitrary way, but they must be combined linearly.

3. Analysis

Having developed a formal framework in which to study to the MRTA problem, we are in a position to apply that framework. Our first step is to analyze some of the key task allocation architectures from the literature. In this section we examine six approaches to MRTA, focusing on three characteristics:

- computation requirements (Cormen et al., 1997)
- communication requirements (Kushilevitz and Nisan, 1997)
- task consideration

In part because of trends in the research community that stress the importance of experimental validation with physical robots, such theoretical aspects of multi-robot coordination mechanisms have been largely ignored. However, they are vitally important to the study, comparison, and objective evaluation of the mechanisms. The large-scale and long-term behavior of the system will be strongly determined by the fundamental characteristics of the underlying algorithm(s). Thus we endeavor to derive and explicate those characteristics here.

Before we continue, however, it will be necessary to explain the methodology that we use in our analysis.

3.1 Methodology

As we stated earlier, the key to effective task allocation for multi-robot systems is to iterate the assignment, in order to deal with changes in the tasks, the robots, and the environment. The architectures under study achieve this iteration in different ways, along two dimensions. First, while some approaches allow assignment and reassignment of all tasks at each iteration, some never reassign tasks (or at least only reassign them because of robot failure). Second, some approaches periodically consider all tasks simultaneously, while others consider single tasks sequentially as they are offered for (re)assignment. Thus, when we discuss complexities, we state them in terms of iterations, though the details of an "iteration" may vary across architectures.

We determine computation requirements, or running time, in the usual way, as the number of times that some dominant operation is repeated. For our domain that operation is usually either a calculation or comparison of utility, and running time is stated as a function of n and m, the numbers of robots and tasks, respectively. Since modern robots have significant processing capabilities on-board and can easily work in parallel, we assume that the computational load is evenly distributed over the robots, and state the running time as it is *for each robot*. For example, if we need to find for each robot the task with the highest utility, then the running time is $O(m)$, because each robot performs m comparisons, in parallel. Note that we do *not* measure or consider the actual running time of the utility calculation, in large part because that information is not generally reported. Rather we operate under the assumption that the utility calculations are computationally similar enough to be meaningfully compared.

We determine communication requirements as the total number of inter-robot messages sent over the network. We do not consider message sizes, on the assumption that they are generally small (e.g., single scalar utility values) and approximately the same for different algorithms. We also assume that a perfect shared broadcast communication medium is in use and that messages are always broadcast, rather than unicast. So if, for example, each robot must tell every other robot its own highest utility value then the overhead is $O(n)$, because each robot makes a single broadcast.

3.2 Results & Discussion

We have chosen to study six MRTA architectures that have been validated on either physical or simulated robots. Our choices are somewhat subjective, for there are a great many more architectures in the literature. However, we

Name	Computation / Iteration	Communication / Iteration	Task Consideration
ALLIANCE (Parker, 1998)	$O(mn)$	$O(m)$	simultaneous, reassignment
BLE (Werger and Matarić, 2000)	$O(mn)$	$O(mn)$	simultaneous, reassignment
M+ (Botelho and Alami, 1999)	$O(mn)$	$O(mn)$	simultaneous, no reassignment
MURDOCH (Gerkey and Matarić, 2002b)	$O(1)$ / bidder $O(n)$ / auctioneer	$O(n)$	sequential, no reassignment
First-price auctions (Dias and Stentz, 2001)	$O(1)$ / bidder $O(n)$ / auctioneer	$O(n)$	sequential, reassignment
Dynamic role assignment (Chaimowicz et al., 2002)	$O(1)$ / bidder $O(n)$ / auctioneer	$O(n)$	sequential, reassignment

Table 1. Summary of selected MRTA architectures. Shown here are the computational and communication requirements for six key architectures. Note that "iteration" has a different meaning depending on whether tasks are considered simultaneously or sequentially.

believe that we have gathered a set of approaches that is fairly representative of the work to date.

The details of our analysis are presented elsewhere (Gerkey and Matarić, 2002a), and we do not repeat them here. Rather we refer the reader to Table 1, in which our results are summarized. Perhaps the most significant trend in those results is how similar the architectures look when examined within our framework. For example, the architectures listed in the top half of Table 1, which assign available tasks simultaneously, exhibit almost identical algorithmic characteristics. Only the ALLIANCE architecture (Parker, 1998) shows any difference; in this case the decrease in communication overhead is achieved by having each robot internally model the fitness of the other robots, thereby effectively distributing the utility calculations. More striking are the results in the bottom half of Table 1, which lists architectures that assign tasks in a sequential manner: with respect to computational and communication requirements, these architectures are *identical*.

These results are particularly interesting because they suggest that there is some common methodology underlying many existing approaches to MRTA. This trend is difficult or impossible to discern from simply reading the technical papers describing the work, as each researcher tells a different "story" regarding his or her architecture, validates the architecture in a different task domain, and explains it in different terms. However, seen through the lens of our OAP framework, fundamental similarities of the various architectures are immediately obvious. These similarities are encouraging because they suggest that, regardless of the details of the robots or tasks in use, we are all studying a common, deep problem in autonomous coordination. As a corollary, there is some reason to believe that these *ad hoc* architectures may in fact have properties that allow them to be generalized and applied widely.

Without such analysis, it is impossible to objectively compare proposed solutions to robotics problems. Of course we have not captured all relevant aspects of the systems that we have studied. For example, in the ALLIANCE architecture the robots' computational load is increased to handle modeling of other robots, but we do not consider that extra load in our analysis. Such details, which are currently not widely discussed in the literature, will likely become more important as the field demands better cross-evaluation of solutions.

In addition to enabling evaluation, this kind of analysis can be used to explain *why* certain solutions work in practice. For example, the sequential allocation architectures listed in the bottom half of Table 1 are all economically-inspired, built around task *auctions*. While the designers of such architectures generally justify their approach with a loose analogy to the efficiency of the free market as it used by humans, it is possible to gain a clearer understanding of what is happening. When seen through our OAP framework, it is not surprising that auction-based allocation methods work in practice, for it is well known that synthetic economic systems can be used to solve a variety of optimization problems. In fact, an appropriately constructed price-based market (which the previously described architectures approximate to varying degrees) can optimally solve assignment problems. At equilibrium, such a market optimizes costs in the so-called *dual* of the original OAP, resulting in an optimal allocation (Gale, 1960; Bertsekas, 1990).

4. Limitations & Extensions

The framework that we have described in this paper, while useful for understanding the problem of MRTA and analyzing proposed solutions, is by no means perfect or complete. Perhaps the most constraining aspect of our OAP framework is the assumption, detailed in Section 2.2, that we are working with single-robot tasks. Although this assumption holds for much of the current research in MRTA, it does not cover all such work, and will clearly not suffice as more sophisticated task domains are explored in the future.

In seeking to relax this assumption, we inevitably face a problem that is known in the multi-agent community as *coalition formation*. Given a collection of agents (e.g., robots), we want them to autonomously coalesce into teams in order to improve overall task performance. In its most general form, the problem of coalition formation is intractable. To optimally solve this problem for an arbitrary set of tasks, one must search the combinatorial space of possible coalitions. This search is likely to be impractical for even moderately sized *static* coalition formation problems, and the situation is worse for MRTA domains, in which the coalition structures must be *dynamic* in order to respond to changing task requirements. Some heuristic solutions to the coalition formation problem for multi-agent systems have been proposed (e.g., Sandholm and Lesser, 1997; Shehory and Kraus, 1998), but they have not been demonstrated in robotic domains.

Our OAP framework can easily be extended to account for coalitions, while remaining focused on task allocation, in the following way. We assume that we are given an externally generated, non-overlapping coalition structure. Each robot is a member of exactly one coalition, and a coalition contains one or more robots. The multi-robot coalitions can be thought of as special-purpose teams that possess particular skills, such as cooperative object manipulation. Furthermore we assume that, when faced with a potential multi-robot task, members of a coalition can collectively calculate a combined utility estimate. Then we can apply the same formalism as before, but with “coalition” substituted for “robot” in the assignment problem. To respond to changing task requirements, we can employ an online coalition formation system that runs in parallel with the task allocation system, producing new coalition structures (and thus new assignment problems) over time.

Such a solution is not without its disadvantages; for example, some kind of convergence or stability analysis would be required to determine how these two processes would interact. More importantly, the two problems, coalition formation and task allocation, are not truly separable when multi-robot tasks are allowed. In principle, to produce an optimal allocation of such tasks, one must consider the entire space of possible coalitions. Even if optimality is not required, the two problems should be attacked simultaneously. A promising avenue of research in this vein is the extension of auction-based task allocation to provide for “leaders,” whose role is to organize and negotiate on behalf of multi-robot teams (Dias and Stentz, 2002). This leader-based approach to coalition formation dovetails nicely with earlier work on multi-agent coordination systems such as RETSINA (Sycara et al., 1996) and the Open Agent Architecture (Martin et al., 1999), in which “facilitator” or “matchmaker” agents are used to mediate and optimize resource exchanges.

5. Conclusion

With the goal of bringing some objective grounding to an emerging area of research that has, to date, been largely experimental, we have presented a formal framework for studying the problem of multi-robot task allocation (MRTA). We have given a domain-independent statement of the problem and shown it to be an instance of the well-known optimal assignment problem

(OAP). By interpreting them as algorithms for solving the underlying OAP, we have analyzed the computation and communication requirements of several robot task-allocation architectures from the literature, and shown them to be remarkably similar.

We plan to pursue the opportunities provided by the substantial body of work regarding the OAP that is available in other fields, including Operations Research, Economics, and Game Theory. We are currently investigating the applicability to the robot domain of a wide variety of efficient, optimal assignment algorithms, both distributed and centralized. Selected in part by their communication and computation requirements (which are generally well-known), we are adapting, implementing, and experimenting with some of these algorithms in the domain of multi-robot task allocation.

The framework that we have described here is by no means the only way to view multi-robot coordination, and we do not claim that it is the best one. Rather than suggest that everyone use and extend our framework, our goal has been to show that such frameworks are useful and are in fact vital to the further development of the field. Whether studying multi-robot task allocation, simultaneous localization and mapping, or some other domain, a formal framework is required in order to analyze the problems and objectively evaluate solutions, in order to guide research and discussion.

Acknowledgments

The research reported here was conducted at the Interaction Lab, part of the Robotics Research Lab at USC. The work is supported in part by the Intel Foundation, DARPA Grant DABT63-99-1-0015, and Office of Naval Research Grants DURIP N00014-00-1-0638 and N00014-01-1-0354. We thank Michael Wellman, Herbert Dawid, Andrew Howard, and Richard Vaughan for their insightful comments.

References

Arkin, R. C., Balch, T., and Nitz, E. (1993). Communication of behavorial state in multi-agent retrieval tasks. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 588–594.

Bertsekas, D. P. (1990). The Auction Algorithm for Assignment and Other Network Flow Problems: A Tutorial. *Interfaces*, 20(4):133–149.

Botelho, S. and Alami, R. (1999). M+: a scheme for multi-robot cooperation through negotiated task allocation and achievement. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 1234–1239, Detroit, MI.

Chaimowicz, L., Campos, M. F. M., and Kumar, V. (2002). Dynamic Role Assignment for Cooperative Robots. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 293–298, Washington, DC.

Cormen, T. H., Leiserson, C. E., and Rivest, R. L. (1997). *Introduction to Algorithms*. MIT Press, Cambridge, Massachusetts.

Dertouzos, M. L. and Mok, A. K. (1983). Multiprocessor On-Line Scheduling of Hard-Real-Time Tasks. *IEEE Transactions on Software Engineering*, 15(12):1497–1506.

Dias, M. B. and Stentz, A. (2001). A Market Approach to Multirobot Coordination. Technical Report CMU-RI-TR-01-26, The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA.

Dias, M. B. and Stentz, A. (2002). Opportunistic Optimization for Market-Based Multirobot Control. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 2714–2720, Lausanne, Switzerland.

Edgeworth, F. Y. (1967). *Mathematical Psychics: An Essay on the Application of Mathematics to the Moral Sciences*. Augustus M. Kelley, New York. Originally published in 1881.

Gale, D. (1960). *The Theory of Linear Economic Models*. McGraw-Hill Book Company, Inc., New York.

Gerkey, B. P. and Matarić, M. J. (2002a). Multi-Robot Task Allocation: Analyzing the Complexity and Optimality of Key Architectures. Technical Report CRES-02-005, Center for Robotics and Embedded Systems, School of Engineering, University of Southern California.

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

Kushilevitz, E. and Nisan, N. (1997). *Communication Complexity*. Cambridge University Press, Cambridge.

Martin, D. L., Cheyer, A. J., and Moran, D. B. (1999). The open agent architecture: A framework for building distributed software systems. *Applied Artificial Intelligence*, 13(1):91–128.

Matarić, M. J. (1995). Designing and Understanding Adaptive Group Behavior. *Adaptive Behavior*, 4(1):51–80.

Neumann, J. V. and Morgenstern, O. (1964). *Theory of Games and Economic Behavior*. J. Wiley, New York, Third edition.

Parker, L. E. (1998). ALLIANCE: An architecture for fault-tolerant multi-robot cooperation. *IEEE Transactions on Robotics and Automation*, 14(2):220–240.

Parker, L. E. (1999). Cooperative Robotics for Multi-Target Observation. *Intelligent Automation and Soft Computing*, 5(1):5–19.

Russell, S. and Norvig, P. (1995). *Artificial Intelligence: A Modern Approach*. Prentice Hall, Upper Saddle River, New Jersey.

Sandholm, T. W. and Lesser, V. R. (1997). Coalitions among computationally bounded agents. *Artificial Intelligence*, 94(1):99–137.

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

Simon, H. A. (2001). *The Sciences of the Artificial*. MIT Press, Cambridge, Massachusetts, Third edition.

Stone, P. and Veloso, M. (1999). Task Decomposition, Dynamic Role Assignment, and Low-Bandwidth Communication for Real-Time Strategic Teamwork. *Artificial Intelligence*, 110(2):241–273.

Sycara, K., Decker, K., Pannu, A., Williamson, M., et al. (1996). Distributed intelligent agents. *IEEE Expert*, 11(6):36–46.

Thorndike, R. L. (1950). The Problem of Classification of Personnel. *Psychometrika*, 15(3):215–235.

Werger, B. B. and Matarić, M. J. (2000). Broadcast of Local Eligibility for Multi-Target Observation. In Parker, L. E., Bekey, G., and Barhen, J., editors, *Distributed Autonomous Robotic Systems 4*, pages 347–356. Springer-Verlag.