Submitted to *the Intl. J. of Robotics Research*
(Also Technical Report CRES-03-013, University of Southern California, Los Angeles, July 2003)

A formal framework for the study of task allocation in multi-robot systems

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

Abstract

Despite more than a decade of experimental work in multi-robot systems, important theoretical aspects of multi-robot coordination mechanisms have, to date, been largely untreated. To address this issue, we focus on the problem of multi-robot task allocation (MRTA). Most work on MRTA has been ad hoc *and empirical, with many coordination architectures having been proposed and validated in a proof-of-concept fashion, but infrequently analyzed. With the goal of bringing objective grounding to this important area of research, we present a formal study of MRTA problems. A domain-independent taxonomy of MRTA problems is given, and it is shown how many such problems can be viewed as instances of other, well-studied, optimization problems. We demonstrate how relevant theory from operations research and combinatorial optimization can be used for analysis and greater understanding of existing approaches to task allocation, and show how the same theory can be used in the synthesis of new approaches.*

1 Introduction

Over the past decade, a signi£cant shift of focus has occurred in the £eld of mobile robotics as researchers have begun to investigate problems involving multiple, rather than single, robots. From early work on loosely-coupled tasks such as homogeneous foraging (e.g., Matarić (1992)) to more recent work on team coordination for robot soccer (e.g., Stone & 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 signi£cant achievements have been made along these axes, it is no longer suf£cient to show, for example, a pair of robots observing targets or a large group of robots ¤ocking as examples of coordinated robot behavior. Today we reasonably expect to see increasingly larger robot teams engaged in concurrent and diverse tasks over extended periods of time.

1.1 Multi-Robot Task Allocation (MRTA)

As a result of the growing focus on multi-robot systems, multi-robot coordination has received signi£cant attention. In particular, *multi-robot task allocation* (MRTA) has recently risen to prominence and become a key research topic in its own right. As researchers design, build, and use cooperative multi-robot systems, they invariably encounter the fundamental question: “which robot should execute which task?” in order to cooperatively achieve the global goal.

In this work, we are concerned with methods for *intentional* cooperation (Parker 1998). In this model, robots cooperate explicitly and with purpose, often through task-related communication. As compared with *emergent* cooperation (e.g., Deneubourg, Theraulaz & Beckers (1991)), intentional cooperation is usually better suited to the kinds of real-world tasks that humans might want robots to do. If the robots are deliberately cooperating with each other, then, intuitively we expect that humans can deliberately cooperate with them, which is a long-term goal of multi-robot research. Furthermore, intentional cooperation has the potential to better exploit the capabilities of heterogeneous robot teams. In this work, the use of intentional cooperation is at the level of *task allocation*, and need not propagate to the level of task execution. Importantly, we do not prescribe or proscribe any particular method for implementing the details of a task. For example, if a foraging task is assigned to a team of robots because they are best £t for the job, they can execute the task in any way they wish, from probabilistic swarming to classical planning.

1.2 Toward formal analysis

The question of task allocation must be answered, even for relatively simple multi-robot systems, and its importance grows with the complexity, in size and capability, of the system under study. The empirically validated methods demonstrated to date remain primarily *ad hoc* in nature, and relatively little has been written about the general properties of cooperative multi-robot systems. After a decade of research, while cooperative architectures have been proposed, the £eld still lacks a prescription for how to design a MRTA system. Similarly, there has been little attempt to evaluate or compare the proposed architectures, either analytically or empirically.

In this paper we present a particular framework for studying MRTA, based on organizational theory from several £elds, including operations research, economics, scheduling, network ¤ows, and combinatorial optimization. We show how this framework can be used to classify MRTA problems, and evaluate and compare proposed solutions. For the simpler (and mostly widely studied) problems, we provide a complete analysis and prescribe provably optimal, yet tractable, algorithms for their solution. For more dif£cult problems, we suggest candidate approximation algorithms that have enjoyed success in other application domains. There are also some extremely dif£cult MRTA problems for which there do not currently exist good approximations; in such cases we provide formal characterizations of the problems but do not suggest how they should be solved.

Our framework is not meant to be £nal or exhaustive and indeed it has some limitations. However, we believe that the ideas we present constitute a starting point toward a more complete understanding of problems involving MRTA, as well as other aspects of multi-robot coordination.

2 Related work

Research in multi-robot systems has been focused primarily on construction and validation of working systems, rather than more general analysis of problems and solutions. As a result, in the literature, one can £nd many *architectures* for multi-robot coordination, but relatively few formal *models* of multi-robot coordination. We do not attempt here to cover the various proposed and demonstrated architectures; rather we treat the prominent models. For discussion and analysis of several key architectures, see Section 5.1.3.

Formal models of coordination in multi-robot systems tend to target medium- to large-scale systems composed of simple, homogeneous robots, such as the CEBOTS (Fukuda, Nakagawa, Kawauchi & Buss 1988). Agassounon & Martinoli (2002) explored the tradeoffs between using a coarse, macroscopic model of such systems and using detailed, microscopic models of the individuals. Lerman & Galstyan (2002) presented a physics-inspired macroscopic model of a cooperative multi-robot system and showed that it accurately described the behavior of physical robots engaged in stick-pulling and foraging tasks. That kind of model is *descriptive* but not *prescriptive*, in that it does not guide the design of control or coordination mechanisms.

Though simple and elegant, such models are insuf£cient for domains involving complex tasks or requiring precise control. To study complex tasks, Donald, Jennings & Rus (1997) proposed the formalism of *information invariants*, which models the information requirements of a coordination algorithm and provides a mechanism to perform reductions between algorithms. Spletzer & Taylor (2001) developed a prescriptive control-theoretic model of multi-robot coordination and showed that it can be used to produce precise multi-robot box-pushing. Mason (1986) had earlier applied a similar control-theoretic model to box-pushing with dexterous manipulators.

Relatively little work has been done on formal modeling, analysis, or comparison of multi-robot coordination at the level of task allocation. Chien, Barrett, Estlin & Rabideau (2000) developed a baseline geological scenario and used it to compare three different planning approaches to coordinating teams of planetary rovers. Klavins (2002) showed how to apply the theory of communication complexity to the study of multi-robot coordination algorithms. Finally, Jennings & Kirkwood-Watts (1998) described the method of *dynamic teams*, concentrating on programmatic structures that enable the speci£cation of multi-robot tasks.

Our goal in this paper is to £ll a gap in the existing literature on multi-robot coordination. We neither construct a formal model in support of a particular coordination architecture, nor compare different architectures in a particular task domain. Rather, we develop a task- and architecture-independent analytical framework, based on optimization theory, in which to study task allocation problems.

3 Utility

To treat task allocation in an optimization context, one must decide what exactly is to be optimized. Ideally the goal is to directly optimize overall system performance, but that quantity is often dif£cult to measure during system execution. Furthermore, when selecting among alternative task allocations, the impact on system performance of each option is usually not known. Consequently, some kind of performance estimate, such as *utility*, is needed.

Utility is a unifying, if sometimes implicit, concept in economics, game theory, and operations research, as well as in multi-robot coordination. It is based on the notion that each individual can internally estimate the value (or the cost) of executing an action. Depending on the context, utility is also called £tness, valuation, and cost. Within multi-robot research, the formulation of utility can vary from sophisticated planner-based methods (e.g., Botelho & Alami (1999)) to simple sensor-based metrics (e.g., Gerkey & Matarić (2002*b*)). We posit that utility estimation of this kind is carried out somewhere in every autonomous task allocation system, for the heart of any task allocation problem is comparison and selection among a set of available alternatives. Since each system uses a different method to calculate utility, we give the following generic and practical de£nition of utility for multi-robot systems.

It is assumed that each robot is capable of estimating its £tness for every task it can perform. This estimation includes two factors, which are both task- and robot-dependent:

- expected quality of task execution, given the method and equipment to be used (e.g., the accuracy of the map that will be produced using a laser range-£nder),

- expected resource cost, given the spatio-temporal requirements of the task (e.g., the power that will be required to drive the motors and laser range-£nder in order to map the building).

Given a robot R and a task T, if R is capable of executing T, then one can de£ne, on some standardized scale, Q_{RT} and C_{RT} as the quality and cost, respectively, expected to result from the execution of T by R. This results in a combined, nonnegative utility measure:

$$U_{RT} = \begin{cases} Q_{RT} - C_{RT} & \text{if } R \text{ is capable of executing} \\ & T \text{ and } Q_{RT} > C_{RT} \\ 0 & \text{otherwise} \end{cases}$$

For example, given a robot A that can achieve a task T with quality $Q_{AT} = 20$ at cost $C_{AT} = 10$ and a robot B that can achieve the same task with quality $Q_{BT} = 15$ at cost $C_{BT} = 5$, there should be no preference between them when searching for ef£cient assignments, for:

$$U_{AT} = 20 - 10 = 10 = 15 - 5 = U_{BT}.$$

Regardless of the method used for calculation, the robots' utility estimates will be inexact due to sensor noise, general uncertainty, and environmental change. These unavoidable characteristics of the multi-robot domain will necessarily limit the ef£ciency 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 at the task allocation level. When we discuss "optimal" allocations, 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 that used in optimal scheduling (Dertouzos & Mok 1983).

It is important to note that utility is an extremely ¤exible measure of £tness, that can encompass arbitrary computation. The only constraint on a utility estimators is that they must each produce a single scalar value that can be compared for the purpose of ordering candidates for tasks. For example, if the metric for a particular task is distance to a location and a candidate robot employs a probabilistic localization mechanism, then a reasonable utility estimate might be to calculate the distance to the target using the center of mass of the current probability distribution. Other mechanisms, such as planning and learning, can likewise be incorporated into utility estimation. Regardless of 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.

4 Combinatorial optimization

Before entering into a discussion of task allocation problems as being primarily concerned with optimization, it will be necessary to provide some theoretical background. The £eld of combinatorial optimization provides a set-theoretic framework, based on *subset systems*, for describing a wide variety of optimization problems (Ahuja, Magnanti & Orlin 1993):

De£nition 1 (Subset System) : A *subset system* (E, F) is a £nite set of objects E and a nonempty collection F of subsets, called *independent sets*, of E that satis£es the property that if $X \in F$ and $Y \subseteq X$ then $Y \in F$.

That is, any subset of an independent set is also an independent set. A general maximization problem can be de£ned in the following way:

De£nition 2 (Subset Maximization) : Given a a subset system (E, F) and a utility function $u : E \to \mathbb{R}_+$, £nd an $X \in F$ that maximizes the total utility:

$$u(X) = \sum_{e \in X} u(e) \qquad (1)$$

The elements of F are usually not given directly, or at least are inconvenient to represent explicitly. Instead, it is assumed that an *oracle* is available that, given a candidate set X, can decide whether $X \in F$. The job of such an oracle, given a proposed solution, is to verify the feasibility of that solution. For many problems, this veri£cation is computationally trivial when compared to the complexity of the optimization problem.

Given a maximization problem over a subset system, one can de£ne algorithms that attempt to solve it. Of particular interest is the canonical *Greedy algorithm* (Ahuja et al. 1993):

Algorithm 1 (The Greedy algorithm) :

1. Reorder the elements of $E = \{e_1, e_2, \ldots, e_n\}$ such that $u(e_1) \geq u(e_2) \geq \ldots \geq u(e_n)$.
2. Set $X := \emptyset$.
3. For $j = 1$ to n:
   if $X \cup \{e_j\} \in F$ then $X = X \cup \{e_j\}$

This algorithm is an abstraction of the familiar and intuitive greedy algorithm for solving a problem: repeatedly take the best valid option. While the Greedy algorithm performs well on some optimization problems, it can do quite poorly on others. In particular, it performs well on certain subset systems that can be further classi£ed as *matroids*:

De£nition 3 (Matroid) : A subset system (E, F) is a *matroid* if, for each $X, Y \in F$ with $|X| > |Y|$, there exists an $x \in X \setminus Y$ such that $Y \cup \{x\} \in F$.

That is, given two independent sets X and Y, with X larger than Y, Y can be "grown" by adding to it some element from X. With respect to the current discussion, an equivalent de£nition of a matroid is that a subset system (E, F) is a matroid if and only if the Greedy algorithm *optimally* solves the associated maximization problem (Korte & Vygen 2000). In the parlance of algorithmic analysis, matroids satisfy the *greedy-choice property*, which is a prerequisite for a greedy algorithm to produce an optimal solution (Cormen, Leiserson & Rivest 1997). Matroids are of particular interest precisely because their associated optimization problems are amenable to greedy solution.

While the Greedy algorithm does not *optimally* solve every maximization problem, it is useful to know how *poor* the greedy solution can be. For such purposes it is common to report a *competitive factor* for the sub-optimal algorithm For a maximization problem, an algorithm is called α-*competitive* if, for any input, it £nds a solution whose total utility is never less than $\frac{1}{\alpha}$ of the optimal bene£t.

5 A taxonomy of MRTA problems

We propose a taxonomy of MRTA problems based on axes laid out below. Our goals here are two-fold: 1) to show how various MRTA problems can be positioned in the resulting problem space; and 2) to explain how organizational theory relates to those problems and to proposed solutions from the robotics literature. In some cases, it will be possible to construct provably optimal solutions, while in others only approximate solutions are available. There are also some dif£cult MRTA problems for which there do not currently exist good approximations. When designing a multi-robot system, it is essential to understand what kind of task allocation problem is present in order to solve it in a principled manner.

We propose the following three axes for use in describing MRTA problems:

- **single-task robots (ST)** vs. **multi-task robots (MT)**: ST means that each robot is capable of executing as most one task at a time, while MT means that some robots can execute multiple tasks simultaneously.

- **single-robot tasks (SR)** vs. **multi-robot tasks (MR)**: SR means that each task requires exactly one robot to achieve it, while MR means that some tasks can require multiple robots.

- **instantaneous assignment (IA)** vs. **time-extended assignment (TA)**: IA means that the available information concerning the robots, the tasks, and the environment permits only an instantaneous allocation of tasks to robots, with no planning for future allocations. TA means that more information is available, such as the set of all tasks that will need to be assigned, or a model of how tasks are expected to arrive over time.

We denote a particular MRTA problem by a triple of two-letter abbreviations drawn from this list. For example, a problem in which multi-robot tasks must be allocated once to single-task robots is designated ST-MR-IA.

These axes are not meant to be exhaustive, but to allow for a taxonomy that is both broad enough and detailed enough to meaningfully characterize many practical MRTA problems. Furthermore, this taxonomy will often allow for a prescription of solutions. The following sections present the combinations allowed by these axes, discussing for each which MRTA problem(s) it represents and what organizational theory pertains. Section 6 treats some important MRTA problems that are not captured by this taxonomy.

5.1 ST-SR-IA: Single-task robots, single-robot tasks, instantaneous assignment

This problem is the simplest, as it is actually an instance of the *Optimal Assignment Problem* (OAP) (Gale 1960), which is a well-known problem that was originally studied in game theory and then in operations research, in the context of personnel assignment. A recurring special case of particular interest in several £elds of study, this problem can be formulated in many ways. Given the application domain of MRTA, it is £tting to describe the problem in terms of jobs and workers.

De£nition 4 (Optimal Assignment Problem) : Given are m workers, each looking for one job and n prioritized jobs, each requiring one worker. Also given for each worker is a nonnegative skill rating (i.e., utility estimate) that predicts his/her performance for each job; if a worker is incapable of undertaking a job, then the worker is assigned a rating of zero for that job. The goal is to assign workers to jobs so as to maximize overall expected performance, taking into account the priorities of the jobs and the skill ratings of the workers.

The OAP can be cast in many ways, including as an integral linear program (Gale 1960): £nd n^2 nonnegative integers α_{ij} that maximize

$$U = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_{ij} U_{ij} w_j \tag{2}$$

subject to

$$\begin{aligned} \sum_{i=1}^{n} \alpha_{ij} = 1, \quad 1 \leq j \leq n \\ \sum_{j=1}^{n} \alpha_{ij} = 1, \quad 1 \leq i \leq n. \end{aligned} \tag{3}$$

The sum (2) is the overall system utility, while (3) enforces the constraint of working with single-worker jobs and single-job workers (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 (2) subject to (3)), an optimal assignment is constructed by assigning worker i to job j only when $\alpha_{ij} = 1$.

The ST-SR-IA problem can be posed as an OAP in the following way: given m robots, n prioritized tasks, and utility estimates for each of the mn possible robot-task pairs, assign at most one task to each robot. If the robots' utilities can be collected at one machine (or distributed to all machines), then a centralized linear programming approach (e.g., Kuhn's (1955) Hungarian method) will £nd the optimal allocation in $O(mn^2)$ time.

Alternatively, a distributed auction-based approach (e.g., Bertsekas's (1990) Auction algorithm) will £nd the optimal allocation, usually requiring time proportional to the maximum utility and inversely proportional to the minimum bidding increment. In order to understand such economically-inspired algorithms, it is necessary to consider the concept of linear programming duality. As do all maximum linear programs, the OAP has a *dual* minimum linear program, which can be stated as follows: £nd m integers u_i and n integers v_j that minimize:

$$P = \sum_{i=1}^{m} u_i + \sum_{j=1}^{n} v_j \tag{4}$$

subject to:

$$u_i + v_j \geq U_{ij}, \ \forall i, j. \tag{5}$$

The Duality Theorem states that the original problem (called the *primal*) and its dual are equivalent, and that the total utility of their respective optimal solutions are the same (Gale 1960).

Optimal auction algorithms for task allocation usually work in the following way. Construct a price-based *task market*, in which tasks are sold by brokers to robots. Each task j is for sale by a broker, which places a value c_j on the task. Also, robot i places a value h_{ij} on task j. The problem then is to establish task *prices* p_j, which will in turn determine the allocation of tasks to robots. To be *feasible*, the price p_j for task j must be greater than or equal to the broker's valuation c_j; otherwise, the broker would refuse to sell. Assuming that the robots are acting sel£shly, each robot i will elect to buy a task t_i for which its pro£t is maximized:

$$t_i = \underset{j}{\operatorname{argmax}} \, \{h_{ij} - p_j\}. \tag{6}$$

Such a market is said to be at *equilibrium* when prices are such that no two robots select the same task.

At equilibrium, each individual's pro£t in this market is maximized. Furthermore, the pro£ts made by the robots and the pro£ts made by the brokers form an optimal solution to the dual of the OAP:

$$\begin{aligned} u_i &= h_{it_i} - p_{t_i}, \ \forall i \\ v_j &= p_j - c_j, \ \forall j. \end{aligned} \tag{7}$$

Thus, the allocation produced by the market at equilibrium is optimal (Gale 1960).

In MRTA problems, separate valuations are not given in this manner, but only combined utility estimates for robot-task pairs. However, task valuations can be de£ned for the robots and brokers as follows:

$$\begin{aligned} h_{ij} &= \alpha_{ij} \\ c_j &= 0. \end{aligned} \tag{8}$$

The solution to the corresponding dual problem then becomes:

$$\begin{aligned} u_i &= \alpha_{it_i} - p_{t_i} \\ v_j &= p_j. \end{aligned} \quad (9)$$

Note that setting c_j to 0 implicitly states that the brokers always prefer to sell their tasks, regardless of how much they are paid. In other words, it is always better to execute a task than not execute it, regardless of the expected performance. In economic terminology, those are *lexicographic* preferences with regard to the tasks (Pearce 1999). Such preferences violate important assumptions concerning the nature of utility values that are made when building or analyzing general economic systems. Fortunately, in constructing the market corresponding to the ST-SR-IA problem, no assumptions are made concerning the robots' preferences, and so lexicographic preferences do not present a problem. On the other hand, the behavior of more complex, long-lived economies (such as the markets suggested by Dias & Stentz (2001) and Gerkey & Matarić (2002*a*)) may depend strongly the nature of the robots' preferences, especially if the synthetic economies are meant to interact with the human economy.

The two approaches (i.e., centralized and distributed) to solving the OAP represent a tradeoff between solution time and communication overhead. Centralized approaches generally run faster than distributed approaches, but incur a higher communication overhead. In order to explore this tradeoff, we implemented, in ANSI C, the Hungarian method (Kuhn 1955) and the Auction algorithm (Bertsekas 1990).[1] As a simpli£cation for testing, our implementation of the Auction algorithm is contained within a single process, and thus is not truly distributed. The communication overhead of this implementation, as measured by the number of required bid messages, is unaffected. However, the solution times reported here for the Auction algorithm should be considered lower bounds, because they do not include message transmission delays that would be seen in practice with a distributed implementation.

We conducted performance tests on our implementations of the Hungarian method and Auction algorithm on a Pentium III-700MHz running Linux 2.4. Shown in Figures 1 & 2 are the mean computation and communication overheads, respectively, for symmetric assignment problems (i.e., problems where $m = n$) with uniformly randomly distributed utilities.[2] By linear regression, we

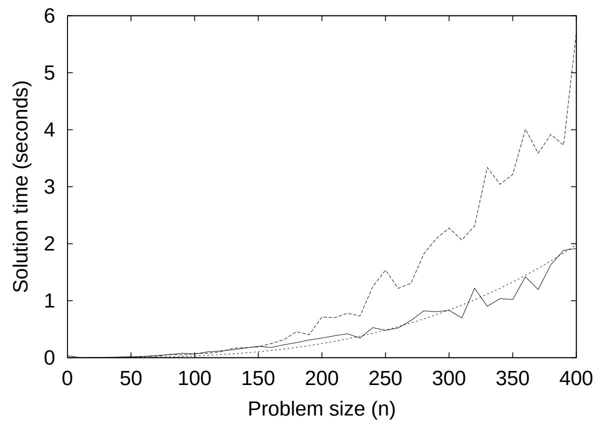

Figure 1: *Comparison of the computational overhead of assignment algorithms. The solid line and dashed line show the amount of time required by the Hungarian method and Auction algorithm, respectively, to solve randomly generated symmetric ($n \times n$) instances of the Optimal Assignment Problem (OAP) on a 700MHz Pentium III running Linux 2.4. For comparison, the dotted line is $10^{-10}n^{3.89}$; the Hungarian method is known to exhibit a running time of $O(n^3)$.*

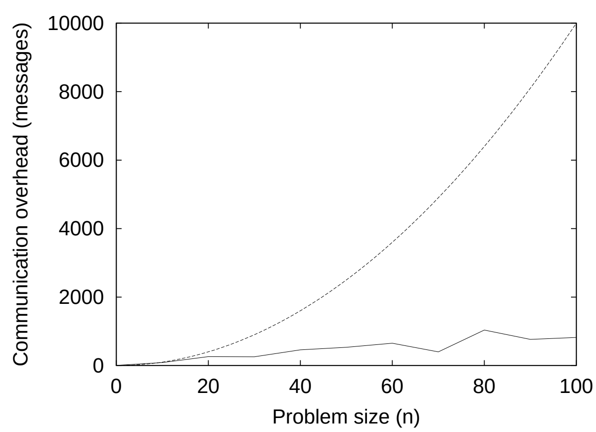

Figure 2: *Comparison of the communication overhead of assignment algorithms. The solid line and dashed line show the number of messages sent by the Hungarian method and Auction algorithm, respectively, when solving randomly generated symmetric ($n \times n$) instances of the Optimal Assignment Problem (OAP).*

[1]The code for these implementations is available from: http://robotics.usc.edu/∼gerkey.

[2]Of course, real MRTA problems are unlikely to exhibit uniformly randomly distributed utilities. Nonetheless, these results are indicative of the running time one can expect from applying these algorithms to MRTA problems.

have determined the running time of our implementation of the Hungarian method to be approximately $O(n^{3.89})$ with a constant coef£cient on the order of 10^{-10} (seconds), which roughly agrees with the theoretically determined running time of $O(n^3)$. As Figure 1 shows, this algorithm is ef£cient enough to be used on line to optimally solve MRTA problems with hundreds of robots and hundreds of tasks.

To implement a centralized assignment algorithm, n^2 messages are required to transmit the utility of each robot for each task; an auction-based solution usually requires far fewer (sometimes fewer than n) messages to reach equilibrium, as shown in Figure 2. With the addition of simple optimizations, such as buffering multiple utility values and transmitting them in one message, this gap in communication overhead will only become apparent in large-scale systems. Furthermore, the time required to transmit a message cannot be ignored, especially in wireless networks, which can induce signi£cant latency. Thus, for small- to medium-scale systems, say $n < 200$, a broadcast-based centralized assignment solution is likely the better choice. Not surprisingly, many MRTA architectures implement some form of this approach (Parker 1998, Werger & Matarić 2000, Castelpietra, Iocchi, Nardi, Piaggio, Scalzo & Sgorbissa 2001, Weigel, Auerback, Dietl, Dümler, Gutmann, Marko, Müller, Nebel, Szerbakowski & Thiel 2001, Østergård, Matarić & Sukhatme 2001).

5.1.1 Variant: iterated assignment

Few MRTA problems exhibit exactly the above one-time assignment structure. However, many problems can be framed as *iterated* instances of ST-SR-IA. Consider the cooperative multi-object tracking problem known as CMOMMT, studied by Parker (1999) and Werger & Matarić (2000), which consists of coordinating robots to observe multiple unpredictably moving targets. When presented with new sensor inputs (e.g., camera images) and consequent utility estimates (e.g., perceived distance to each target), the system must decide which robot should track which target.

Werger & Matarić's (2000) MRTA architecture, Broadcast of Local Eligibility (BLE), solves this iterated assignment problem using the following algorithm:

Algorithm 2 (BLE assignment algorithm) :

1. If any robot remains unassigned, £nd the robot-task pair (i, j) with the highest utility. Otherwise, quit.
2. Assign robot i to task j and remove them from consideration.
3. Go to step 1.

This algorithm is an instance of the canonical Greedy algorithm (Algorithm 1). The OAP is *not* a matroid (see Section 4) and so the Greedy algorithm will not necessarily produce an optimal solution. The Greedy algorithm is known to be 2-competitive for the OAP (Avis 1983), and thus so is BLE. That is, in the worst case, BLE will produce a solution whose bene£t is $\frac{1}{2}$ of the optimal bene£t. Exactly this algorithm, operating on a global blackboard, has been used in a study of the impact of communication and coordination on MRTA (Østergård et al. 2001). A very similar assignment algorithm is also used by Botelho & Alami's (1999) MRTA architecture M+.

Parker's (1998) MRTA architecture L-ALLIANCE, which can also perform iterated allocation, learns its assignment algorithm from experience. The resulting algorithm is similar to, but potentially more sophisticated than, the Greedy algorithm. If well-trained, the L-ALLIANCE assignment algorithm can outperform the Greedy algorithm (Parker 1994), but is not guaranteed to be optimal.

Another domain in which the iterated OAP arises is robot soccer. Since many of the robots are interchangeable, it is often advantageous to allow any player to take on any role within the team, according to the current situation in the game. The resulting coordination problem can be cast as an iterated assignment problem in which the robots' roles are periodically reevaluated, usually at a frequency on the order of 10Hz. This utility-based dynamic role assignment problem and has been studied by many (Stone & Veloso 1999, Weigel et al. 2001, Castelpietra et al. 2001, Emery, Sikorski & Balch 2002, Vail & Veloso 2003).

It is common in the robot soccer domain for each robot to calculate its utility for each role and periodically broadcast these values to its teammates. The robots can then execute, in parallel, some centralized assignment algorithm. For example, Castelpietra et al.'s (2001) assignment algorithm consists of ordering the roles in descending priority and then assigning each to the available robot with the highest utility. This algorithm is yet another instance of the Greedy algorithm. Vail & Veloso (2003) also employ the Greedy algorithm with £xed priority roles. Weigel et al. (2001) employ a similar but slightly more sophisticated algorithm that tries to address the problem of excessive role-swapping by imposing stricter prerequisites for reassignment. Among other things, the algorithm requires that both robots "want" to exchange roles in order to maximize their respective utilities, recalling the conditions for equilibrium in markets (see Section 5.1). However, Weigel et al.'s (2001) algorithm is not guaranteed to produce optimal assignments of roles, a fact that can easily be shown by counterexample.

Since the number of robots involved in many iterated MRTA problems today is small ($n \leq 11$ for robot soccer, which is more than for most current multi-robot systems), $O(n^3)$ *optimal* assignment algorithms could easily replace the suboptimal *ad hoc* assignment algorithms that are typically used. As the performance results in the previous section show, the Hungarian method can be used to solve typical problems in less than $1ms$ per iteration with the moderately powerful computers found on today's robots.

Since there is some additional cost for running an optimal algorithm (if only in the work involved in the implementation), one might ask whether the optimal solution provides a suf£cient bene£t. For example, it is known that for arbitrary assignment problems, the Greedy algorithm's *worst-case* behavior is to produce a solution with half of the optimal utility. However, it is not known how the algorithm can be expected to perform on typical MRTA problems, which exhibit some structure and are unlikely to present pathological utility combinations. Anecdotal evidence suggests that the Greedy algorithm works extremely well on such problems. An interesting avenue of research would be to analytically determine how well the Greedy algorithm will perform on the kinds of utility landscapes that are encountered in MRTA problems.

5.1.2 Variant: online assignment

In some MRTA problems, the set of tasks is not revealed at once, but rather the tasks are introduced one at a time. If robots that have already been assigned cannot be reassigned, then this problem is a variant of SR-ST-IA, known as *online assignment* (Kalyanasundaram & Pruhs 1993). Instead of being initially given, the robot-task utility matrix is revealed one column (or row) at a time. If previously assigned robots *can* be reassigned, then the problem reduces to an instance of the iterated SR-ST-IA problem, which can be optimally solved with standard assignment algorithms.

The MRTA problems solved by Gerkey & Matarić's (2002*b*) MURDOCH system, in which tasks are randomly injected into the system over time, are instances of the online assignment problem. The MURDOCH assignment algorithm can be stated as follows:

Algorithm 3 (MURDOCH assignment algorithm) :

1. When a new task is introduced, assign it to the most £t robot that is currently available.

This simple algorithm is yet another instance of the Greedy algorithm (Algorithm 1), and is known in the context of network ¤ows as the *Farthest Neighbor* algorithm. Not surprisingly, the online assignment problem is *not* a matroid (see Section 4); the Greedy algorithm is known to be 3-competitive with respect to the optimal *post hoc* of¤ine solution. Furthermore, this performance bound is the best possible for any online assignment algorithm (Kalyanasundaram & Pruhs 1993). Thus, without a model of the tasks that are to be introduced, and without the option of reassigning robots that have already been assigned, it is impossible to construct a better task allocator than MURDOCH.

5.1.3 Analysis of some existing approaches

Presumably because it is the simplest case of MRTA, the ST-SR-IA problem has received the most attention from the research community. Having now developed a formal framework in which to study to this MRTA problem, it is possible to apply that framework toward an analysis of some of the key task allocation architectures from the literature. In this section six approaches to MRTA are analyzed, focusing on the following three characteristics[3]:

1. computation requirements
2. communication requirements
3. solution quality

The theoretical aspects of multi-robot coordination mechanisms are vitally important to the study, comparison, and objective evaluation, as the large-scale and long-term system behavior is strongly determined by the fundamental characteristics of the underlying algorithm(s). First, the methodology that is used in the analysis is explained.

Methodology Computational requirements, or running time, are determined in the usual way, as the number of times that some dominant operation is repeated. For the MRTA domain that operation is usually either a calculation or comparison of utility, and running time is stated as a function of m and n, the number of robots and tasks, respectively. Since modern robots have signi£cant processing capabilities on board and can easily work in parallel, in this analysis 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 each robot must select the task with the highest utility, then the running time is $O(n)$, because each robot performs n comparisons, in parallel. Note that this analysis does *not* measure or consider the actual running time of the utility calculation, in large part because that information is not generally reported. Rather it is assumed that the utility calculations are computationally similar enough to be meaningfully compared.

Communication requirements are determined as the total number of inter-robot messages sent over the network.

[3] This analysis was originally presented in Gerkey & Matarić (2003)

In the analysis 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. Further, we assume that a perfect shared broadcast communication medium is used 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 $O(m)$, because each robot makes a single broadcast.

Solution quality is reported in terms of a competitive factor (see Section 4).

Results & discussion Next, six MRTA architectures that have been validated on either physical or simulated robots are analyzed. While there are a great many architectures in the literature, this summary attempts to gather a set of approaches that is representative of the work to date.

The previous sections of this paper present components of the analysis, and are not repeated here. Tables 1 & 2 summarize the results for the iterated assignment architectures and online assignment architectures, respectively. Perhaps the most signi£cant trend in these results is how similar the architectures look when examined within this framework. For example, the iterated architectures listed in Table 1, which assign all 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 £tness of the others, thereby effectively distributing the utility calculations. More striking are the results in Table 2, which lists architectures that assign tasks in a sequential manner: with respect to computational and communication requirements, these architectures are *identical*. In terms of solution quality, Dias & Stentz's (2001) and Chaimowicz et al.'s (2002) approaches, which allow reassignment of tasks, can potentially perform better than MURDOCH.

These results are particularly interesting because they suggest that there is some common methodology underlying many existing approaches to MRTA. This trend is dif£cult or impossible to discern from reading the technical papers describing the work, as each architecture is described in different terms, and validated in a different task domain. However, viewed in the formal framework described here, fundamental similarities of the various architectures become obvious. These similarities are encouraging because they suggest that, regardless of the details of the robots or tasks in use, the various authors are all studying a common, fundamental problem in autonomous coordination. As a corollary, there is now a formal grounding for the belief that these *ad hoc* architectures may have properties that allow them to be generalized and applied widely.

Of course, the described framework does not capture all relevant aspects of the systems under study. For example, in the ALLIANCE architecture, the robots' computational load is increased to handle modeling of other robots, but this analysis does not consider that extra load. Such details, which are currently not widely discussed in the literature, will likely become more important as the £eld moves toward improved 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 online assignment architectures listed in Table 2 are all economically-inspired, built around task *auctions*. The designers of such architectures generally justify their approach with a loose analogy to the ef£ciency of the free market as it is used by humans. With a formal analysis, it is possible to gain a clearer understanding of why auction-based allocation methods work in practice. Speci£cally, is well known that synthetic economic systems can be used to solve a variety of optimization problems. As explained in Section 5.1, an appropriately constructed price-based market, at equilibrium (i.e., when the prices are such that no two utility-maximizing robots would select the same task), produces *optimal* assignments. The previously described economically-inspired architectures approximate such a market to varying degrees.

5.2 ST-SR-TA: Single-task robots, single-robot tasks, time-extended assignment

When the system consists of more tasks than robots, or if a model of how tasks will arrive exists, then the robots' *future* utilities for the tasks can be predicted with some accuracy, and the problem is an instance of ST-SR-TA. This problem is one of building a time-extended *schedule* of tasks for each robot, with the goal of minimizing total weighted cost. Using Brucker's (1998) terminology, this problem is an instance of the class of scheduling problems

$$R \,||\, \sum w_j C_j.$$

That is, the robots execute tasks in parallel (R) and the optimization criterion is the weighted sum of execution costs ($\sum w_j C_j$). Problems in this class are strongly $\mathcal{NP}$-hard (Bruno, Coffman & Sethi 1974). Even for relatively small problems, the exponential space of possible schedules precludes enumerative solutions.

[4] In addition to solving the ST-SR-IA problem, the ALLIANCE architecture is also capable of building time-extended task *schedules* in order to solve a form of the ST-SR-TA problem (see Section 5.2.1).

Name	**Computation / iteration**	**Communication / iteration**	**Solution quality**
ALLIANCE[4] (Parker 1998)	$O(mn)$	$O(m)$	at least 2-competitive
BLE (Werger & Matarić 2000)	$O(mn)$	$O(mn)$	2-competitive
M+ (Botelho & Alami 1999)	$O(mn)$	$O(mn)$	2-competitive

Table 1: *Summary of selected iterated assignment architectures for MRTA. Shown here for each architecture are the computational and communication requirements, as well as solution quality.*

Name	**Computation / task**	**Communication / task**	**Solution quality**
MURDOCH (Gerkey & Matarić 2002*b*)	$O(1)$ / bidder $O(n)$ / auctioneer	$O(n)$	3-competitive
First-price auctions (Dias & Stentz 2001)	$O(1)$ / bidder $O(n)$ / auctioneer	$O(n)$	at least 3-competitive
Dynamic role assignment (Chaimowicz et al. 2002)	$O(1)$ / bidder $O(n)$ / auctioneer	$O(n)$	at least 3-competitive

Table 2: *Summary of selected online assignment architectures for MRTA. Shown here for each architecture are the computational and communication requirements, as well as solution quality.*

A means of treating ST-SR-TA is to ignore the time-extended component and approximate the problem as an instance of the ST-SR-IA problem (Section 5.1), followed by an instance of the online assignment problem (Section 5.1.2). For example, given m robots and n tasks, with $m > n$, the following approximation algorithm can be used:

Algorithm 4 (ST-SR-TA approximation algorithm) :

1. Optimally solve the initial $m \times n$ assignment problem.
2. Use the Greedy algorithm to assign the remaining tasks in an online fashion, as the robots become available.

The performance of this algorithm is bounded below by the normal Greedy algorithm, which is 3-competitive for online assignment. The more tasks that are assigned in the £rst step, the better this algorithm will perform. As the difference between the number of robots and the number of tasks that are initially presented decreases (i.e., $(n - m) \rightarrow 0$), performance approaches optimality, wherein all tasks are assigned in one step. Thus, although it is not guaranteed to produce optimal solutions, Algorithm 4 should work well in practice, especially for ST-SR-TA problems with short time horizons.

Another way to approach this problem is to employ an iterative task allocation system, such as Dias & Stentz's (2001) free market. The robots would opportunistically exchange tasks over time, thereby modifying their schedules. This idea is demonstrated by the multi-robot exploration system described by Zlot, Stentz, Dias & Thayer (2002). However, without knowledge of the exact criteria used to decide when and with whom each robot will trade, it is impossible to determine the algorithmic characteristics (including solution quality) of this method.

5.2.1 Variant: ALLIANCE Ef£ciency Problem

Parker (1995) formulated a related MRTA problem called the ALLIANCE Ef£ciency Problem (AEP). Given is a set of tasks making up a mission, and the objective is to allocate a subset of these tasks to each robot so as to minimize the maximum time taken by a robot to serially execute its allocated tasks. Thus in order to solve the AEP, one must construct a time-extended schedule of tasks for each robot. This problem is an instance of the class of scheduling problems:

$$R \,||\, C_{max}.$$

Problems in this class are known to be strongly $\mathcal{NP}$-hard (Garey & Johnson 1978). Parker (1995) arrived at the same conclusion regarding the AEP, by reduction from the $\mathcal{NP}$-complete problem PARTITION.

To attack the AEP, Parker (1998) used a learning approach, in which the robots learn both their utility estimates and their scheduling algorithms from experience. When trained for a particular task domain, this system has the potential to outperform Algorithm 4 (but it is not guaranteed to do so).

5.3 ST-MR-IA: Single-task robots, multi-robot tasks, instantaneous assignment

Many MRTA problems involve tasks that require the combined effort of multiple robots. In such cases, we must consider *combined* utilities of groups of robots, which are in general *not* sums over individual utilities; utility may be de£ned arbitrarily for each potential group. For example, if a task requires a particular skill or device, then any group of robots without that skill or device has zero utility with respect to that task, regardless of the capabilities of the other robots in the group. This kind of problem is signi£cantly more dif£cult than the previously discussed MRTA problems, which were restricted to single-robot tasks. In the multi-agent community, the ST-MR-IA problem is referred to as *coalition formation*, and has been extensively studied (e.g., Sandholm & Lesser (1997), Shehory & Kraus (1998)).

It is natural to think of the ST-MR-IA problem as splitting the set of robots into task-speci£c coalitions. A relevant concept from set theory is that of a set partition. A family X is a *partition* of a set E if and only if the elements of X are mutually disjoint and their union is E:

$$\begin{aligned} \bigcap_{x \in X} &= \emptyset \\ \bigcup_{x \in X} &= E. \end{aligned} \tag{10}$$

With the idea of partitions in mind, a well-known problem in combinatorial optimization called the (maximum utility) Set Partitioning Problem, or SPP (Balas & Padberg 1976) is relevant:

De£nition 5 (Set Partitioning Problem (SPP)) : Given a £nite set E, a family F of acceptable subsets of E, and a utility function $u : F \rightarrow \mathbb{R}_+$, £nd a maximum-utility family X of elements in F such that X is a partition of E.

The ST-MR-IA problem can be cast as an instance of SPP, with E as the set of robots, F as the set of all feasible coalition-task pairs, and u as the utility estimate for each such pair.

Unfortunately, the SPP is strongly $\mathcal{NP}$-hard (Garey & Johnson 1978). Fortunately, the problem has been studied in depth (e.g., Atamtürk, Nemhauser & Savelsbergh (1995)), especially in the context of solving crew scheduling problems for airlines (e.g., Marsten & Shepardson (1981), Hoffman & Padberg (1993)). As a result, many heuristic SPP algorithms have been developed.

It remains to be seen whether such heuristic algorithms are applicable to MRTA problems. Some approximation algorithms, including those of Hoffman & Padberg (1993) and Atamtürk et al. (1995), have been shown to produce high-quality solutions to many instances of SPP. Even with hundreds of rows/columns and using mid-1990s workstation-class machines, these algorithms require at most a few tens of seconds to arrive at a solution. With ever-increasing computational power available on robots, it seems plausible that SPP approximation algorithms could be used to solve small- and medium-scale instances of the ST-MR-IA problem. To this end, a potentially important question is whether and how these algorithms can be parallelized. Shehory & Kraus (1998) showed how to implement a parallel SPP algorithm for coalition formation in a multi-agent context. Another important point is that, in order to apply certain SPP algorithms to ST-MR-IA problems, it may be necessary to enumerate a set of feasible coalition-task combinations. In the case that the space of such combinations is very large, there is a need to prune the feasible set; pruning can take advantage of sensor-based metrics such as physical distance (e.g., if two robots are more than 50 meters apart, then disallow any coalitions that contain them both).

5.4 ST-MR-TA: Single-task robots, multi-robot tasks, time-extended assignment

The ST-MR-TA class of problems includes both coalition formation and scheduling. To produce an optimal solution, all possible schedules for all possible coalitions must be considered. This problem is $\mathcal{NP}$-hard. If the coalitions are given, with no more than one coalition allowed for each task, the result is an instance of a multiprocessor scheduling problem:

$$MPTm \mid\mid \sum w_j C_j.$$

Even with two processors ($MPT2 \mid\mid \sum w_j C_j$), this problem is strongly $\mathcal{NP}$-hard (Hoogeveen, van del Velde & Veltman 1994), as is the unweighted version ($MPT2 \mid\mid \sum C_j$) (Cai, Lee & Li 1998). With three processors, the maximum £nishing time version ($MPT3 \mid\mid C_{max}$) is also strongly $\mathcal{NP}$-hard (Hoogeveen et al. 1994).

A means of treating ST-MR-TA is to ignore the time-extended component and approximate the problem as an instance of iterated ST-MR-IA. A greedy approximation algorithm akin to Algorithm 4 can be employed. Unfortunately, the quality of such an approximation is dif£cult to determine. Another approach is to employ a leader-based mechanism to dynamically form coalitions and build task schedules for them, as described by Dias & Stentz (2002). However, the performance and overhead of this method will also be dif£cult, if not impossible, to predict without detailed information about the implementation (how many and which robots will be leaders, how does a leader select among candidate coalitions? For how long do coalitions persist, etc.).

5.5 MT-SR-IA & MT-SR-TA: Multi-task robots, single-robot tasks

The MT-SR-IA and MT-SR-TA problems are currently uncommon, as they assume robots that can each concurrently execute multiple tasks. Today's mobile robots are generally actuator-poor. Their ability to affect the environment is typically limited to changing position, so they can rarely execute more than one task at a time. However, there are sensory and computational tasks that £t the MT-SR-IA or MT-SR-TA models quite well.

Solving the MT-SR-IA problem is equivalent to solving the ST-MR-IA problem (see Section 5.3), with the robots and tasks interchanged in the SPP formulation. Likewise, the MT-SR-TA problem is equivalent to the ST-MR-TA problem (see Section 5.4). Thus the analysis and algorithms provided for the multi-robot task problems also directly apply here to the multi-task robot problems.

5.6 MT-MR-IA: Multi-task robots, multi-robot tasks, instantaneous assignment

When a system consists of both multi-task robots and multi-robot tasks, the result is an instance of the MT-MR-IA problem. A relevant concept from set theory is the set cover. A family X is a *cover* of a set E if and only if the union of elements of X is E:

$$\bigcup_{x \in X} = E. \tag{11}$$

As compared with a partition (see Section 5.3), the subsets in a cover need not be disjoint. A well-known problem in combinatorial optimization called the (minimum cost) Set Covering Problem, or SCP (Balas & Padberg 1972), is relevant:

De£nition 6 (Set Covering Problem (SCP)) : Given a £nite set E, a family F of acceptable subsets of E, and a cost function $c : F \to \mathbb{R}_+$, £nd a minimum-cost family X of elements in F such that X is a cover of E.

The MT-MR-IA problem can be cast as an instance of the SCP, with E as the set of robots, F as the set of all feasible (and possibly overlapping) coalition-task pairs, and c as the cost estimate for each such pair.

Though super£cially similar to the SPP, the SCP is in fact a "distant relative," with the solution space of the SCP being far less constrained (Balas & Padberg 1976). The two problems are similar in that the SCP is also strongly $\mathcal{NP}$-hard (Korte & Vygen 2000).

Chvátal (1979) developed a greedy approximation algorithm for the SCP. The competitive factor for this algorithm is logarithmic in the size of the largest feasible subset (i.e., $\max_{f \in F} |f|$), and the running time is polynomial in the number of feasible subsets (i.e., $|F|$). Bar-Yehuda & Even (1981) present another heuristic set covering algorithm, whose competitive factor is the maximum number of subsets to which any element belongs (i.e., $\max_{e \in E} |\{f \in F : e \in f\}|$), and whose running time is the sum of the sizes of the feasible subsets (i.e., $\sum_{f \in F} |f|$) (Korte & Vygen 2000).

The important trend to note is that these heuristic algorithms perform well when the space of feasible subsets is limited, and that they perform poorly in the most general case of the SCP, with all subsets allowed. For MRTA, this result suggests that such algorithms would best be applied in environments in which the space of possible coalitions is naturally limited, as is the case with heterogeneous and/or physically distantly separated robots. In the case of equally-skilled collocated robots, these algorithms would tend to run slowly and produce poor-quality solutions.

To the authors' knowledge, set covering algorithms have not been applied to MRTA problems, and it is an open question as to whether such an application would be bene£cial. However, Shehory & Kraus (1996) successfully adapted and distributed Chvátal's (1979) approximation algorithm for use in multi-agent systems, which suggests that SCP algorithms may indeed be viable for MRTA problems.

5.7 MT-MR-TA: Multi-task robots, multi-robot tasks, time-extended assignment

The MT-MR-TA problem is an instance of a scheduling problem with multiprocessor tasks and multipurpose machines:

$$MPTmMPMn \mid\mid \sum w_j C_j.$$

This problem is strongly $\mathcal{NP}$-hard, because it includes as a special case the strongly $\mathcal{NP}$-hard scheduling problem $MPT2 \mid\mid \sum w_j C_j$. We are not aware of any heuristic or approximation algorithms for this dif£cult problem.

6 Other problems

Although the taxonomy given in the previous sections covers many MRTA domains, several potentially important problems are excluded. Next we describe some problem domains that are not captured by the taxonomy.

6.1 Interrelated utilities

Consider the problem of assigning target points to a team of robots that are cooperatively exploring an unknown environment. Many targets (e.g., the frontiers of Yamauchi (1998)) may be known at one time, and so is is possible to build a schedule of targets for each robot. Unfortunately, this problem is not an instance of ST-SR-TA, because the cost for a robot to visit target C depends on whether that robots £rst visits target A or target B. Instead, this problem is an instance of the multiple traveling salesperson problem (MTSP); even in the restricted case of one salesperson, MTSP is strongly $\mathcal{NP}$-hard (Korte & Vygen 2000). If, as is often the case with exploration, it is possible to discover *new* targets over time, then the problem is an instance of the dynamic MTSP, which is clearly at least as dif£cult as the classical MTSP.

Given the dif£culty of the multi-robot exploration problem, it is not surprising that researchers have not attempted to solve it directly or exactly. A heuristic approximation is offered by Zlot et al. (2002), who use TSP heuristics to build target schedules and derive costs that are used in Dias & Stentz's (2001) market-based task allocation architecture. When a robot discovers a new target, it inserts the new target into its schedule, but retains the option of later auctioning the target off to another, closer robot.

The multi-robot exploration problem is an example of a larger class of problems, in which a robot's utility for a task may depend on which other tasks that robot executes. These problems in turn form part of another, more general class of problems in which a robot's utility for a task may depend on which other tasks *any* robot executes. That is, each robot-task utility can depend on the overall allocation of tasks to robots. Such interrelated utilities can sometimes be tractably captured with factored Partially Observable Markov Decision Processes (POMDPs), assuming that a world model is available (Guestrin, Koller & Parr 2001).

For mobile robots, this situation can arise any time that physical interference contributes signi£cantly to task performance. For example, consider a multi-robot resource transportation problem in which each robot must choose which of a predetermined number of source-sink roads to travel. The decision of which road to travel should take into account the congestion caused by other robots. Taking the position that interference effects are dif£cult or impossible to adequately model *a priori*, Dahl, Matarić & Sukhatme (2002) developed a reinforcement learning approach to the multi-robot resource transportation problem. The robots do not communicate with each other directly, but rather through physical interactions, with each robot maintaining and updating an estimate of the utility for each available road. This approach was shown to produce higher-quality solutions than those produced without learning, and added no communication overhead.

6.2 Task constraints

In addition to an assumption of independent utilities, our taxonomy also assumes independent tasks. If instead there are constraints between the tasks, such as sequential or parallel execution, then this taxonomy will not suf£ce. Although the topic of job constraints is addressed by the scheduling literature (Brucker 1998), the addition of such constraints generally increases problem dif£culty, and tractable algorithms exist for only the simplest kinds of constraints. A possible way to approach this problem is with the dynamic constraints satisfaction method described by Modi, Jung, Tambe, Shen & Kulkarni (2001).

7 Summary & future work

In the £eld of mobile robotics, the study of multi-robot systems has grown signi£cantly in size and importance. Having solved some of the basic problems concerning single-robot control, many researchers have shifted their focus to the study of multi-robot coordination. There are by now a plethora of examples of demonstrated coordinated behavior in multi-robot systems, and almost as many proposed coordination architectures. However, despite more than a decade of research, the £eld so far lacks a theoretical foundation that can explain or predict the behavior of a multi-robot system. Our goal in this paper has been to provide a candidate framework for studying such systems.

The word "coordination" is somewhat imprecise, and has been used inconsistently in the literature.[5] In order to be precise about the problem with which we are concerned, we de£ned a smaller problem: multi-robot task allocation (MRTA). That is, given some robots and some tasks, which robot(s) should execute which task(s)? This restricted problem is both theoretically and practically im-

[5] The words "coordination," "cooperation," and "collaboration" are often used when discussing multi-robot systems, but their relationships are rarely established. Some researchers (e.g., Emery et al. (2002)) assert that each word represents a unique concept, while some dictionaries (e.g., Barnhart (1952)) de£ne at least one of the three words in terms of the other two.

portant, and is supported by the signi£cant body of existing work that focuses on MRTA, in one form or another.

To date, the majority of research in MRTA has been experimental in nature. The standard procedure, followed by a large number of researchers, has been to construct a MRTA architecture and then validate it in one or more application domains. This proof-of-concept method has led to the proposal of many MRTA architectures, each of which has been experimentally validated to a greater or lesser extent, sometimes in simulation and sometimes with physical robots. These research efforts are undeniably useful, as they demonstrate that successful multi-robot coordination is possible, even in relatively complex environments. However, to date it has not been possible to draw general conclusions regarding the underlying MRTA problems, or to establish a prescriptive strategy that would dictate how to achieve task allocation in a given multi-robot system.

We view MRTA problems as fundamentally organizational in nature, in that the goal is to allocate limited resources in such a way as to ef£ciently achieve some task(s). In this paper we have shown how MRTA problems can be studied in a formal manner by adapting to robotics some of the theory developed in relevant disciplines that study organizational and optimization problems. These disciplines include operations research, economics, scheduling, network ¤ows, and combinatorial optimization.

As an example of how such theory can be applied to understanding task allocation problems and solutions, consider the idea of using synthetic markets as distributed coordination mechanisms. As we noted, it is common to justify such a use of market-based techniques by reference to the ef£cacy of the free market as it is experienced by humans, as for example stated by the economic theorist Hayek (1945):

> "If we can agree that the economic problem of society is mainly one of rapid adaptation to changes in the particular circumstances of time and place, it would seem to follow that the ultimate decisions must be left to the people who are familiar with these circumstances, who know directly of the relevant changes and of the resources immediately available to meet them."

Though not without a certain romantic appeal, the invocation of this justi£cation for the success of market methods in synthetic systems (e.g., Dias & Stentz (2001), Gerkey & Matarić (2002*a*)) is far less principled than that provided by operations research: the OAP is a maximum problem for a linear program; any such problem can be transformed into a *dual* minimum problem, and the solution to such a dual is the same as the prices that result from the evolution of a certain price-based market system (Section 5.1). In this light, the fact that market-based techniques result in ef£cient allocations of tasks is not surprising.

Using such connections to relevant optimization theory, we have presented in this paper an analytical framework in which to study MRTA problems. We have provided formal characterizations of a wide range of such problems, in the larger context of a taxonomy. For the easier problems, we have provided provably optimal algorithms that can be used in place of commonly-employed *ad hoc* or greedy solutions. For the more dif£cult problems, we have, wherever possible, provided suggestions toward their heuristic solution. Thus, this work can be used to aid further research into multi-robot coordination by allowing for the formal classi£cation of MRTA problems, and by sometimes prescribing candidate solutions.

The presented MRTA formalism is very general, in that it relies only on domain-independent theory and techniques. Thus, for example, the taxonomy given in Section 5 should apply equally well in multi-agent and multi-robot systems. However, in exchange for such generality, this formalism is only capable of providing coarse characterizations of MRTA problems and their proposed solutions. Consider the analysis showing that MURDOCH, as an implementation of the canonical Greedy algorithm, is 3-competitive for the online assignment problem. This kind of competitive factor gives an algorithm's *worst-case* behavior, which may be quite different from its *average-case* behavior. In this respect, the bounds established for existing MRTA architectures, in terms of computational overhead, communication overhead, and solution quality, are relatively loose.

One way to tighten these bounds is to add domain-speci£c information to the formalism. By capturing and embedding models of how real MRTA domains behave and evolve over time, it should be possible to make more accurate predictions about algorithmic performance. For example, while the classical theory of the OAP makes no assumptions about the nature of the utility matrices that form the input, MRTA problems are likely to exhibit signi£cant structure in their utility values. Far from randomly generated, utility values generally follow one of a few common models, determined primarily by the kind of sensor data that are used in estimating utility. If only "local" sensor information is used (e.g., can the robot currently see a particular target, and if so, how close is it?), then utility estimates tend to be strongly bifurcated (e.g., a robot will have very high utility for those targets that it can see, and zero utility for all others). On the other hand, if "global" sensor information is available (e.g., how close is the robot to a goal location?), then utility estimates tend to be smoother (e.g., utility will fall off smoothly in space away from the goal). A promising avenue for future research would be to characterize this "utility landscape" as it is encountered in MRTA domains, and then classify different MRTA problems according to the shapes of their landscapes, and make predictions about, for example, how well a greedy assignment algorithm should be expected to work, as opposed to a more costly optimal assignment algorithm.

Acknowledgments

The research reported here was conducted at the Interaction Lab, part of the Center for Robotics and Embedded Systems (CRES) at the University of Southern California. This work was supported in part by the Intel Foundation, DARPA Grant DABT63-99-1-0015 (MARS), and Of£ce of Naval Research Grants N00014-00-1-0638 (DURIP) and N00014-01-1-0354. We thank Herbert Dawid, Andrew Howard, Richard Vaughan, and Michael Wellman for their insightful comments.

References

Agassounon, W. & Martinoli, A. (2002), A Macroscopic Model of an Aggregation Experiment using Embodied Agents in Groups of Time-Varying Sizes, *in* 'Proc. of the IEEE Conf. on System, Man and Cybernetics (SMC)', Hammamet, Tunisia, pp. 250–255.

Ahuja, R. K., Magnanti, T. L. & Orlin, J. B. (1993), *Network Flows: Theory, Algorithms, and Applications*, Prentice Hall, Upper Saddle River, New Jersey.

Atamtürk, A., Nemhauser, G. & Savelsbergh, M. (1995), 'A Combined Lagrangian, Linear Programming and Implication Heuristic for Large-Scale Set Partitioning Problems', *J. of Heuristics* **1**, 247–259.

Avis, D. (1983), 'A Survey of Heuristics for the Weighted Matching Problem', *Networks* **13**, 475–493.

Balas, E. & Padberg, M. W. (1972), 'On the Set-Covering Problem', *Operations Research* **20**(6), 1152–1161.

Balas, E. & Padberg, M. W. (1976), 'Set Partitioning: A Survey', *SIAM Review* **18**(4), 710–760.

Bar-Yehuda, R. & Even, S. (1981), 'A linear-time approximation algorithm for the weighted vertex cover problem', *J. of Algorithms* **2**, 198–203.

Barnhart, C. L., ed. (1952), *The American College Dictionary*, Random House, New York.

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

Botelho, S. C. & 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)', Detroit, Michigan, pp. 1234–1239.

Brucker, P. (1998), *Scheduling Algorithms*, 2^{nd} edn, Springer-Verlag, Berlin.

Bruno, J. L., Coffman, E. G. & Sethi, R. (1974), 'Scheduling Independent Tasks To Reduce Mean Finishing Time', *Communications of the ACM* **17**(7), 382–387.

Cai, X., Lee, C.-Y. & Li, C.-L. (1998), 'Minimizing Total Completion Time in Two-Processor Task Systems with Prespeci£ed Processor Allocations', *Naval Research Logistics* **45**(2), 231–242.

Castelpietra, C., Iocchi, L., Nardi, D., Piaggio, M., Scalzo, A. & Sgorbissa, A. (2001), Communication and Coordination among heterogeneous Mid-size players: ART99, *in* P. Stone, T. Balch & G. Kraetzschmar, eds, 'RoboCup 2000, LNAI 2019', Springer-Verlag, Berlin, pp. 86–95.

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

Chien, S., Barrett, A., Estlin, T. & Rabideau, G. (2000), A comparison of coordinated planning methods for cooperating rovers, *in* 'Proc. of Autonomous Agents', Barcelona, Spain, pp. 100–101.

Chvátal, V. (1979), 'A greedy heuristic for the set cover problem', *Mathematics of Operations Research* **4**, 233–235.

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

Dahl, T. S., Matarić, M. J. & Sukhatme, G. S. (2002), Adaptive spatio-temporal organization in groups of robots, *in* 'Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)', Lausanne, Switzerland, pp. 1044–1049.

Deneubourg, J.-L., Theraulaz, G. & Beckers, R. (1991), Swarm-made architectures, *in* 'Proc. of the European. Conf. on Arti£cial Life (ECAL)', Paris, France, pp. 123–133.

Dertouzos, M. L. & 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. & Stentz, A. (2001), A Market Approach to Multirobot Coordination, Technical Report CMU-RI-TR-01-26, The Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania.

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

Donald, B., Jennings, J. & Rus, D. (1997), 'Information invariants for distributed manipulation', *The Intl. J. of Robotics Research* **16**(5), 673–702.

Emery, R., Sikorski, K. & Balch, T. (2002), Protocols for Collaboration, Coordination, and Dynamic Role Assignment in a Robot Team, *in* 'Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)', Washington, DC, pp. 3008–3015.

Fukuda, T., Nakagawa, S., Kawauchi, Y. & Buss, M. (1988), Self organizing robots based on cell structures – CEBOT, *in* 'Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)', IEEE Computer Society Press, pp. 145–150.

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

Garey, M. R. & Johnson, D. S. (1978), '"Strong" NP-Completeness Results: Motivation, Examples, and Implications', *J. of the ACM* **25**(3), 499–508.

Gerkey, B. P. & Matarić, M. J. (2002*a*), A market-based formulation of sensor-actuator network coordination, *in* 'Proc. of the AAAI Spring Symp. on Intelligent Embedded and Distributed Systems', Palo Alto, California, pp. 21–26.

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

Gerkey, B. P. & Matarić, M. J. (2003), Multi-Robot Task Allocation: Analyzing the Complexity and Optimality of Key Architectures, *in* 'Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)', Taipei, Taiwan. To appear.

Guestrin, C., Koller, D. & Parr, R. (2001), Multiagent Planning with Factored MDPs, *in* 'Proc. of Advances in Neural Information Processing Systems (NIPS)', Vancouver, Canada, pp. 1523–1530.

Hayek, F. A. (1945), 'The Use of Knowledge in Society', *The American Economic Review* **35**(4), 519–530.

Hoffman, K. L. & Padberg, M. W. (1993), 'Solving Airline Crew Scheduling Problems by Branch-and-Cut', *Management Science* **39**(6), 657–682.

Hoogeveen, J., van del Velde, S. & Veltman, B. (1994), 'Complexity of scheduling multiprocessor tasks with prespeci£ed processor allocations', *Discrete Applied Mathematics* **55**, 259–272.

Jennings, J. S. & Kirkwood-Watts, C. (1998), Distributed Mobile Robotics by the Method of Dynamic Teams, *in* 'Proc. of the Intl. Symp. on Distributed Autonomous Robotic Systems (DARS)', Karlsruhe, Germany.

Kalyanasundaram, B. & Pruhs, K. (1993), 'Online Weighted Matching', *J. of Algorithms* **14**, 478–488.

Klavins, E. (2002), Communication Complexity of Multi-Robot Systems, *in* 'Proc. of the Intl. Workshop on the Algorithmic Foundations of Robotics (WAFR)', Nice, France. To appear.

Korte, B. & Vygen, J. (2000), *Combinatorial Optimization: Theory and Algorithms*, Springer-Verlag, Berlin.

Kuhn, H. W. (1955), 'The Hungarian Method for the Assignment Problem', *Naval Research Logistics Quarterly* **2**(1), 83–97.

Lerman, K. & Galstyan, A. (2002), 'Mathematical Model of Foraging in a Group of Robots: Effect of Interference', *Autonomous Robots* **13**(2), 127–141.

Marsten, R. E. & Shepardson, F. (1981), 'Exact Solution of Crew Scheduling Problems Using the Set Partitioning Model: Recent Successful Applications', *Networks* **11**, 165–177.

Mason, M. T. (1986), 'Mechanics and planning of manipulator pushing operations', *The Intl. J. of Robotics Research* **5**(3), 53–71.

Matarić, M. J. (1992), Designing Emergent Behaviors: From Local Interactions to Collective Intelligence, *in* J.-A. Meyer, H. Roitblat & S. Wilson, eds, 'From Animals to Animats 2, Second International Conference on Simulation of Adaptive Behavior (SAB-92)', MIT Press, pp. 432–441.

Modi, J., Jung, H., Tambe, M., Shen, W.-M. & Kulkarni, S. (2001), A Dynamic Distributed Constraint Satisfaction Approach to Resource Allocation, *in* 'Proc. of the Intl. Conf. on Principles and Practices of Constraint Programming', Paphos, Cyprus.

Østergård, E. H., Matarić, M. J. & Sukhatme, G. S. (2001), Distributed multi-robot task allocation for emergency handling, *in* 'Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)', Wailea, Hawaii, pp. 821–826.

Parker, L. E. (1994), Heterogeneous Multi-Robot Cooperation, PhD thesis, MIT EECS Department.

Parker, L. E. (1995), L-ALLIANCE: A Mechanism for Adaptive Action Selection in Heterogeneous Multi-Robot Teams, Technical Report ORNL/TM-13000, Oak Ridge National Laboratory.

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.

Pearce, D. W., ed. (1999), *The MIT Dictionary of Modern Economics*, 4th edn, The MIT Press, Cambridge, Massachusetts.

Sandholm, T. W. & Lesser, V. R. (1997), 'Coalitions among computationally bounded agents', *Arti£cial Intelligence* **94**(1), 99–137.

Shehory, O. & Kraus, S. (1996), Formation of overlapping coalitions for precedence-ordered task-execution among autonomous agents, *in* 'Proc. of the Intl. Conf. on Multi Agent Systems (ICMAS)', Kyoto, Japan, pp. 330–337.

Shehory, O. & Kraus, S. (1998), 'Methods for task allocation via agent coalition formation', *Arti£cial Intelligence* **101**(1–2), 165–200.

Simon, H. A. (2001), *The Sciences of the Arti£cial*, 3rd edn, MIT Press, Cambridge, Massachusetts.

Spletzer, J. R. & Taylor, C. J. (2001), A Framework for Sensor Planning and Control with Applications to Vision Guided Multi-robot Systems, *in* 'Proc. of Computer Vision and Pattern Recognition Conf. (CVPR)', Kauai, Hawaii, pp. 378–383.

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

Vail, D. & Veloso, M. (2003), Dynamic Multi-Robot Coordination, *in* A. Schultz et al., eds, 'Multi-Robot Systems: From Swarms to Intelligent Automata, Volume II', Kluwer Academic Publishers, the Netherlands, pp. 87–98.

Weigel, T., Auerback, W., Dietl, M., Dümler, B., Gutmann, J.-S., Marko, K., Müller, K., Nebel, B., Szerbakowski, B. & Thiel, M. (2001), CS Freiburg: Doing the Right Thing in a Group, *in* P. Stone, T. Balch & G. Kraetzschmar, eds, 'RoboCup 2000, LNAI 2019', Springer-Verlag, Berlin, pp. 52–63.

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

Yamauchi, B. (1998), Frontier-Based Exploration Using Multiple Robots, *in* 'Proc. of Autonomous Agents', Minneapolis, Minnesota, pp. 47–53.

Zlot, R., Stentz, A., Dias, M. B. & Thayer, S. (2002), Multi-Robot Exploration Controlled by a Market Economy, *in* 'Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)', Washington, DC, pp. 3016–3023.