Jones and Maja J Mataric´, "Behavior-Based Coordination in Multi-Robot Systems", *Autonomous Mobile Robots: Sensing, Control, Decision-Making, and Applications*, Sam S. Ge and Frank L. Lewis, eds., Marcel Dekker, Inc., 2005.

Behavior-Based Coordination in Multi-Robot Systems

Chris Jones and Maja J Matarić

Computer Science Department

University of Southern California

941 West 37th Place, Los Angeles, CA 90089-0781 USA

cvjones|maja@robotics.usc.edu

The successful deployment of a multi-robot system (MRS) requires an effective method of coordination to mediate the interactions among the robots and between the robots and the task environment in order for a given system-level task to be performed. The design of coordination mechanisms has received increasing attention in recent years and has included investigations into a wide variety of coordination mechanisms. A popular and successful framework for the control of robots in coordinated MRS is *behavior-based control* (1,2). Behavior-based control is a methodology in which robots are controlled through the principled integration of a set of interacting behaviors (e.g., wall following, collision avoidance, landmark recognition, etc.) in order to achieve desired system-level behavior. This chapter will describe, through explanation, discussion of demonstrated simulated and physical mobile robots, and formal design and analysis, the range and capabilities of behavior-based control applied to multi-robot coordination.

We begin by providing a brief overview of single-robot control philosophies and architectures, including behavior-based control, in Section 1. In Section 2 we move from single robots to multi-robot systems (MRS) and discuss the additional challenges this transition entails. In Section 3 we use empirical case studies to discuss and demonstrate three important ways in which robots can interact, and thus coordinate, their behavior. In Section 4 we discuss formal approaches to the design and analysis of MRS that are of fundamental importance if the full potential of MRS is to be achieved. Finally, in Section 5 we briefly discuss the future of coordinated behavior-based MRS and conclude the chapter.

1. Overview of Robot Control Architectures

In this section, we briefly discuss the most popular approaches and techniques for the control of a single robot. In the next section we proceed with the fundamental principle of this chapter, the control of multiple robots, and how it is related to, and different from, the control of a single robot.

1.1 Single Robot Control

We define *robot control* as the process of mapping a robot's sensory information into actions in the real world. We do not consider entities that make no use of sensory information in control decisions as robots, nor do we consider entities that do not perform actions as robots, because neither category is truly interacting in the real world. Any robot must, in one manner or another, use incoming sensory information to make decisions about what actions to execute. There are a number of control philosophies dictating how this mapping from sensory information to actions should occur, each with its advantages and disadvantages. A continuum of approaches to robot control can be described as a spectrum spanning from deliberative to reactive control.

The *deliberative* approach to robot control is usually computationally intensive due to the use of explicit reasoning or planning using symbolic representations and world models (3). For the reasoning processes to be effective, complete and accurate models of the world are required. In domains where such models are difficult to obtain, such as in dynamic and fast-changing environments or situations with significant uncertainty in the robot's sensing and action, it may be impossible for the robot to act in an appropriate or timely manner using deliberative control (3,4).

In contrast to deliberative control, the *reactive* approach to robot control is characterized by a tight coupling of sensing to action, typically involving no intervening reasoning (5,6). Reactive control does not require the acquisition or maintenance of world models, as it does not rely on the types of complex reasoning processes utilized in deliberative control. Rather, simple rule-based methods involving a minimal amount of computation, internal representations, or knowledge of the world are typically used. This makes reactive control especially well suited to dynamic and unstructured worlds where having access to a world model is not a realistic option. Furthermore, the minimal amount of computation involved means reactive systems are able to respond in a timely manner to rapidly changing dynamics.

A middle ground between deliberative and reactive philosophies is found in *hybrid* control, exemplified by three-layered architectures (7,8). In this approach, a single controller includes both reactive and deliberative components. The reactive part of the controller handles low-level control issues requiring fast response time, such as local obstacle avoidance. The deliberative part of the controller handles high-level issues on a longer time-scale, such as global path planning. A necessary third component of hybrid controllers is a middle layer that interfaces the reactive and deliberative components. Three-layered architectures aim to harness the best of reactive controllers in the form of dynamic and time-responsive control and the best of deliberative controllers in the form of globally efficient actions over a long time-scale. However, there are complex issues involved in interfacing these fundamentally differing components and the manner in which their functionality should be partitioned is not yet well understood.

Behavior-based control, described in detail in the next section, offers an alternative to hybrid control. It can also include both deliberative and reactive components, but unlike hybrid control, it is composed of a set of independent modular components that are executed in parallel (1,2).

The presented spectrum of control approaches is continuous and a precise classification of a specific controller on the continuum may be difficult. The distinction between deliberative and reactive control, and hybrid and behavior-based control is often a matter of degree, based on the amount of computation performed and the response time of the system to relevant changes in the world. In a specific domain, the choice of controller is dependent on many factors, including how responsive the robot must be to changes in the world, how accessible a world model is, and what level of efficiency or optimality is required.

1.2 Behavior-Based Control

The control methodology we focus on in this chapter is behavior-based (BB) control. The BB approach to robot control must not be classified as strictly deliberative or reactive, as it can, and in many cases is, both. However, BB control is most closely identified (often incorrectly so) with the reactive side of the control spectrum, because primary importance is placed on maintaining a tight, real-time coupling between sensing and action (7,8).

Fundamentally, a behavior-based controller is composed of a set of modular components, called behaviors, which are executed in parallel. A *behavior* is a control law that clusters a set of constraints in order to achieve and maintain a goal (1,2). Each behavior receives inputs from sensors and/or other behaviors and provides outputs to the robot's actuators or to other behaviors. For example, an obstacle avoidance behavior might send a command to the robot's wheels to turn left or right if the robot sensors detect the robot is moving directly toward an obstacle. There is no centralized world representation or state in a BB system. Instead, individual behaviors and networks of behaviors maintain any models or state information.

Many different behaviors may independently receive input from the same sensors and output action commands to the same actuators. The issue of choosing a particular action given inputs from potentially multiple sensors and behaviors is called *action selection* (12). One of well-known mechanism for action selection is the use of a predefined behavior hierarchy, as in the Subsumption Architecture (10), in which commands from the highest-ranking active behavior are sent to the actuator and all others are ignored. (Note, however, that the Subsumption Architecture has most commonly been used in the context of reactive and not BB systems.) Numerous principled as well as ad hoc methods for addressing the action selection problem have been developed and demonstrated on robotic systems. These include varieties of command fusion (13) and spreading of activation (14), among many others. For a comprehensive survey on action selection mechanisms, see (15).

BB systems are varied, but there are two fundamental tenets all BB systems inherently adhere to: 1) the robot is embodied and 2) the robot is situated. A robot is *embodied* in the sense that it has a physical body and its behavior is limited by physical realities, uncertainties, and consequences of its actions, all of which may be hard to predict or simulate. A robot is *situated* in the sense that it is immersed in the real world and acts directly on the sensory information received from that world, not on abstract or processed representations of the world.

BB control makes no assumptions on the availability of a complete world model; therefore, it is uncommon for a BB controller to perform extensive computation or reasoning relying on such a model. Instead, BB controllers maintain a tight coupling of sensing and action, allowing them to act in a timely manner in response to dynamic and fast-changing worlds. However, BB systems have also demonstrated elegant use of distributed representations enabling robot mapping and task learning (16,17,18).

This section has discussed approaches and philosophies to the control of a single robot, with a focus on the BB approach. In the next section, the scope is expanded to consider the control of a coordinated group of multiple robots.

2. From Single Robot Control to Multi-Robot Control

In this section we discuss the advantages and additional issues involved in the control of multi-robot systems (MRS) as compared to the single-robot systems (SRS) discussed in the previous section. An *MRS* is a system composed of multiple, interacting robots. The study of MRS has received increased attention in recent years. This is not surprising, as continually improving robustness, availability, and cost-effectiveness of robotics technology has made the deployment of MRS consisting of increasingly larger numbers of robots possible. With the growing interest in MRS comes the expectation that, at least in some important respects, multiple robots will be superior to a single robot in achieving a given task. In this section we outline the benefits of a MRS over a SRS and introduce issues involved in MRS control and how they are similar and different to those of SRS control.

This chapter is focused on distributed MRS in which each robot operates independently under local sensing and control. *Distributed MRS* stand in contrast to *centralized MRS*, in which each robot's actions are not completely determined locally, as they may be determined by an outside entity, such as another robot or by any type of external command. In distributed MRS, each robot must make its own control decisions based only on limited, local, and noisy sensor information. We limit our consideration in this chapter to distributed MRS because they are the most appropriate for study with regard to systems that are scalable and capable of performing in uncertain and unstructured real-world environments where uncertainties are inherent in the sensing and action of each robot. Furthermore, this chapter is centered on achieving system-level coordination in a distributed BB MRS. Strictly speaking, the issues in a centralized MRS are more akin to a scheduling or optimal assignment and less of a problem of coordination in a distributed system.

2.1 Advantages and Challenges of Multi-Robot Systems

Potential advantages of MRS over SRS include a reduction in total system cost by utilizing multiple simple and cheap robots as opposed to a single complex and expensive robot. Also, multiple robots can increase system flexibility and robustness by taking advantage of inherent parallelism and redundancy. Furthermore, the inherent complexity of some task environments may require the use of multiple robots, as the necessary capabilities or resource requirements are too substantial to be met by a single robot.

However, the utilization of MRS poses potential disadvantages and additional challenges that must be addressed if MRS are to present a viable and effective alternative to SRS. A poorly designed MRS, with individual robots working toward opposing goals, can be less effective than a carefully designed SRS. A paramount challenge in the design of effective MRS is managing the complexity introduced by multiple, interacting robots. As such, in most cases just taking a suitable SRS solution and scaling it up to multiple robots is not adequate.

2.2 Necessity of Coordination in Multi-Robot Systems

In order to maximize the effectiveness of a MRS, the robots' actions must be spatio-temporally coordinated and directed towards the achievement of a given system-level task or goal. Just having robots interact is not sufficient in itself to produce interesting or practical system-level coordinated behavior. The design of MRS can be quite challenging because unexpected system-level behaviors may emerge due to unanticipated ramifications of the robots' local interactions. In order for the interacting robots to produce coherent task-directed behavior, there must be some overarching coordination mechanism that spatio-temporally organizes the interactions in a manner appropriate for the task.

The design of such coordination mechanisms can be difficult; nonetheless, many elegant handcrafted distributed MRS have been demonstrated, both in simulation and on physical robots (19,20,21). The methods by which these systems have achieved task-directed coordination are diverse and the possibilities are seemingly limited only by the ingenuity of the designer. From a few robots performing a manipulation task (22,23), to tens of robots exploring a large indoor area (24,25), to potentially thousands of ecosystem monitoring nanorobots (26,27), as the number of robots in the system increases, so does the necessity and importance of coordination. The next section examines mechanisms by which system-level coordination can be successfully achieved in a MRS.

3. From Local Interactions to Global Coordination

Given the importance of coordination in a MRS, we now address the issue of how to organize the robots' local interactions in a coherent manner in order to achieve system-level coordination. There are many mechanisms by which the interactions can be organized. We classify them into three broad and often overlapping classes: interaction through the environment, interaction through sensing, and interaction through communication. These classes are not mutually exclusive because MRS can, and often do, simultaneously utilize mechanisms from any or all of these classes to achieve system-level coordinated behavior.

In the following sections we describe each of these interaction classes in detail. Through the discussion of empirical case studies we demonstrate how each type of interaction can be used to achieve system-level coordination in a MRS.

3.1 Interaction Through the Environment

The first mechanism for interaction is through the robots' shared environment. This form of interaction is *indirect* in that it consists of no explicit communication or physical interaction between robots. Instead, the environment itself is used as a medium of indirect communication. This is a powerful approach that can be utilized by very simple robots with no capability for complex reasoning or direct communication.

An example of interaction through the environment is demonstrated in *stigmergy*, a form of interaction employed by a variety of insect societies. Originally introduced in the biological sciences to explain some aspects of social insect nest-building behavior, stigmergy is defined as the process by which the coordination of tasks and the regulation of construction do not depend directly on the workers, but on the constructions themselves (28). This concept was first used to describe the nest-building behavior of termites and ants (29). It was shown that coordination of building activity in a termite colony was not inherent in the termites themselves. Instead, the coordination mechanisms were found to be regulated by the task environment, in that case the growing nest structure. A location on the growing nest stimulates a termite's building behavior, thereby transforming the local nest structure, which in turn stimulates additional building behavior of the same or another termite.

Through the careful design of robot sensing, actuation, and control features, it is possible to utilize the concept of stigmergy in task-directed MRS. This powerful mechanism of coordination is attractive as it typically requires minimal capabilities of the individual robots. The robots do not require direct communication, unique recognition of other robots or even distinguishing other robots from miscellaneous objects in the environment, or the performance of computationally intensive reasoning or planning.

Stigmergy, and more generally interaction through the environment, has been successfully demonstrated as a mechanism to coordinate robot actions in a number of

MRS. It has been demonstrated in an object manipulation domain (30) in which a large box was transported to a goal location through the coordinated pushing actions of a group of robots. There was no globally agreed upon plan as to how or over what trajectory the box should be moved; however, each robot could indirectly sense the pushing actions of other robots through the motions of the box itself. Through simple rules, each robot decided whether to push the box or move to another location based on the motions of the box itself. As a large enough number of robots pushed in compatible directions, the box moved, which in turn encouraged other robots to push in the same direction.

Other examples of the use of stigmergy in MRS include distributed construction in which a given structure was built in a specified construction sequence (31). The individual robots were not capable of explicit communication and executed simple rule-based controllers in which local sensory information was directly mapped to construction actions. The construction actions of one robot altered the environment, and therefore the subsequent sensory information available for it and all other robots. This new sensory information then activated future construction actions. In the following subsection we discuss in detail how the concept of stigmergy was utilized in a MRS object clustering task domain (28).

3.2 Interaction Through the Environment Case Study: Object Clustering

We now describe an empirical case study in an object clustering task domain for which interaction through the environment was used to achieve system-level coordination. The clustering task domain requires a group of objects, originally uniformly positioned in an enclosed environment, to be re-positioned by a group of robots into a single dense cluster of objects. There is no *a priori* target location for the cluster in the environment. Rather, the position of the cluster is to determined dynamically at the time of task execution.

The particular approach to the object clustering task we describe here is from work presented in (28). There, the robots performing the task were extremely simple, capable only of picking up and transporting and dropping a single object at a time. The robots had very limited local sensing and no explicitly communication, memory of past actions, or recognition of other robots. Even with these highly limited capabilities, a homogeneous MRS composed of such robots was shown to be capable of successfully and robustoly performing the object clustering task.

The robots in this task domain were able to coherently achieve system-level coordination in the formation of a single cluster. The mechanism by which they achieved coordination was an example of interaction through the environment. The robots communicated through their individual placement of objects over time, thus modifying the task environment, and thereby indirectly influencing the future object-placement behaviors of other robots and themselves. The location of the final cluster was not determined through explicit communication, negotiation, or planning on the part of the robots. Rather, it was determined through a symmetry break in the initially uniform distribution of objects.

Once a small cluster began to form, it was likely to grow larger. During the early stages of task execution, several clusters were likely to be formed. However, over time, a single large cluster resulted.

The robots in this work were designed in a manner that carefully exploited the physical dynamics of interaction between the robots and their environment. Their hardware and rules were tuned so as to be probabilistically more likely to pick up an object that is not physically proximate to other objects (thus conserving clusters), to not drop objects near boundaries (thus avoiding hard-to-find objects), and to be probabilistically more likely to deposit an object near other objects (thereby building up clusters). Together, there properties resulted in a form of positive feedback in which the larger a cluster of objects became, the more likely it was to grow even larger.

Similar approaches employing stigmergy were also demonstrated in the physical segregation and sorting of a collection of object classes. Additional studies with physical robots have been conducted and, by making various changes in the robots and the task environment, is has been demonstrated that one can influence the location of the final cluster by initializing the initial distribution of objects in a non-uniform manner (28).

Given this specific example of system-level coordination achieved through the use of interaction through the environment, in the following subsection we move on to the next method of organizing the robot's interactions: interaction through sensing.

3.3 Interaction Through Sensing

The second mechanism for interaction among robots is through sensing. As described in (19), interaction through sensing 'refers to local interactions that occur between robots as a result of sensing one another, but without explicit communication.' As with interaction through the environment, interaction through sensing is also *indirect* as there is no explicit communication between robots; however, it requires each robot to be able to distinguish other robots from miscellaneous objects in the environment. In some instances, each robot may be required to uniquely identify all other robots, or classes of other robots. In other instances, it may only be necessary to simply distinguish robots from other objects in the environment.

Interaction through sensing can be used by a robot to model the behavior of other robots or to determine what another robot is doing in order to make decisions and respond appropriately. For example, flocking birds use sensing to monitor the actions of other birds in their vicinity to make local corrections to their own motion. It has been shown that effective flocking results from quite simple local rules followed by each bird responding to the direction and speed of the local neighbors (32).

In the follow subsection we describe a case study in a formation marching domain in which interaction through sensing is used to achieve coordinated group behavior. Other domains in which interaction through sensing has been utilized in MRS include flocking (33), in which each robot adjusts it motions according to the motions of locally observed robots. Through this process, the robots can be made to move as a coherent flock through an obstacle-laden and dynamic environment. Interaction through sensing has also been demonstrated in an adaptive division of labor domain (34). In that domain, each robot dynamically changes the task it is executing based on the observed actions of other robots and the observed availability of tasks in the environment. Through this process, the group of robots coherently divides the labor of the robots appropriately across a set of available tasks.

3.4 Interaction Through Sensing Case Study: Formation Marching

In this section we describe an empirical case study of a formation marching task domain for which interaction through sensing was used to achieve system-level coordination. The formation marching task domain requires a group of robots to achieve and maintain relative positions to one another as the group moves through the environment n a global formation. Each robot in the MRS operates under local sensing and control and is not aware of global information such as all other robot's positions and headings. In some environments, the formation may need to be perturbed in order for the group to move through a constrained passage or around obstacles. In such cases, the formation needs to correctly re-align after the perturbation.

The approach to formation marching described here was presented in (35). The general idea of the approach is that every robot in the MRS positioned itself relative to a designated neighbor robot. This neighbor robot, in turn, positioned itself relative to its own designated neighbor robot. As all robots are only concerned with their relative positions with respect to their neighbor robot, no robot is aware of, or needs to be aware of, the global positions and headings of all robots in the formation. Each robot only needs to be capable of determining the distance and heading to its neighbor. The global geometry of the formation was then determined through the defined chain of neighbors.

A “leader” robot has no neighbors and independently determines the speed and heading of the entire formation. Therefore, as the leader robot moves forward, the robot(s) that had the leader as their neighbor also move forward. This forward motion propagates down the chain of designated neighbors, causing the entire formation to move.

The formation could be dynamically changed by altering the structure of the local neighbor relationships. For example, if the desired formation is a line, each robot may be designated a neighbor robot to its left or right for which it desires to stay next to in order to maintain the line formation. If a cue is given to all robots to change to a diamond formation, each robot may follow a new neighbor at a different relative position and the line formation would then be dynamically changed to a diamond.

In the following subsection we move to the next method of organizing the robot's interactions: interaction through communication.

3.5 Interaction Through Communication

The third mechanism for interaction among robots is through explicit communication. Unlike the first two forms of interaction, described above, which were indirect, in interaction through communication robots may communicate with others directly. Such *robot-directed* communication can be used to request information or action from other robots or to respond to received requests.

Communication in physical robotics is not free or reliable and can be constrained by limited bandwidth and range, and unpredictable interference. When utilizing it, one must consider how and toward what end it is used. In some domains, such as the Internet, communication is reliable and of unlimited range; however, in physical robot systems, communication range and reliability are important factors in system design (2,36).

There are many types of communication. Communication could be direct from one robot to another, direct from one robot to a class of other robots, or broadcast from one robot to all others. Furthermore, the communication protocol can range from simple protocol-less schemes to a complex negotiation-based and communication-intensive schemes. The information encoded in a communication may be state information contained by the communicating robot, a command to one or more other robots, or a request for additional information from other robots, etc.

Communications may be task-related rather than robot-directed, in which case it is made available to all (or a subset) of the robots in the MRS. A common task-related communication scheme is *publish/subscribe messaging*. In publish/subscribe messaging, subscribing robots request to receive certain categories of messages, and publishing robots supply messages to all appropriate subscribers.

In the next subsection, we describe a case study of the effective use of interaction through communication.

3.6 Interaction Through Communication Case Study: Multiple Target Tracking

The case study on interaction through communication in this section is focused on the use of explicit communication in a multi-target tracking task as discussed in (37). In multi-target tracking, the goal is to have a set of robots with limited sensing ranges position and orient themselves such they are able to acquire and track multiple objects moving through their environment. The locations, trajectories, and number of targets are not known *a priori*. These difficulties are compounded in a distributed MRS, where the system must determine which robot(s) should monitor which target(s). Robots redundantly tracking the same target may be wasting resources and letting another target remain untracked. In this domain, explicit communication between the robots has been shown to be capable of effectively achieving system-level coordination.

In the implementation described in (37), each robot had a limited sensing and communication range. Communication was used by each robot to transmit the position and velocities of all targets within its sensing range to all other robots within its communication range. This simple communication scheme involved no handshaking or negotiation.

Each robot was constantly evaluating the importance of its current tracking activities and possible changes in position that could increase the importance of its tracking activities. Communication was used to allow each robot to keep a local map of target movements within the robot's communication range but outside its sensing range. As a result, the group as a whole effectively tracked a maximum number of targets with a minimum number of available robots.

This demonstration of the use of interaction through communication concludes the discussion of MRS coordination mechanisms. As was mentioned above, any given MRS is likely to use any or all of the three mechanisms in varying degrees to achieve system-level coordination. Through an improved understanding of each of these mechanisms of coordination, one is better positioned to design a MRS utilizing the most appropriate combination of mechanisms for achieving a given task. In the next section we provide a discussion on formal methods for the design and analysis of MRS that can provide a principled foundation upon which to base such design decisions.

4. Formal Design and Analysis of Multi-Robot Systems

The design of coordination mechanisms for multi-robot systems (MRS) has proven to be a difficult problem. In the last decade, the design of a variety of such mechanisms over a wide range of task domains has been studied (19,20). Although the literature highlights some elegant solutions, they are generally domain-specific and provide only indirect insights into important questions such as how appropriate a given coordination mechanism is for a particular domain, what performance characteristics one should expect from it, how it is related to other coordination mechanisms, and how one can modify it to improve system performance. These questions must be answered in a principled manner before one can quickly and efficiently produce an effective MRS for a new task domain. To fully utilize the power and potential of MRS and to move the design process closer to a science, principled design tools and methodologies. Such tools and methodologies are needed for establishing a solid foundation upon which to construct increasingly capable, robust, and efficient MRS.

The design of an effective task-directed MRS is often difficult because there is a lack of understanding of the relationship between different design options and resulting task performance. In the common trial-and-error design process, the designer constructs a MRS and then tries it out either in simulation or on physical robots. Either way, the process is resource-intensive. Ideally, the designer should be equipped with an analytical tool for the analysis of a potential MRS design. Such a tool would allow for efficient evaluation of different design options and thus result in more effective and optimized MRS designs.

The BB paradigm for multi-robot control is popular in MRS because it is robust to the dynamic interactions inherent in any MRS. Any MRS represents a highly non-linear system in which the actions of one robot are affected by the actions of all other robots. This makes any control approach that relies on complex reasoning or planning ineffective because it is intractable to accurately predict future states of a non-trivial MRS. For this reason, BB control is frequently used in MRS. The simplicity of the individual robots also confers the advantage of making the external analysis of predicted system performance on a given task feasible.

In the remainder of this section we discuss a variety of approaches to the analysis and synthesis of MRS.

4.1 Analysis of Multi-Robot Systems Using Macroscopic Models

Macroscopic models reason about the system-level MRS behavior without explicit consideration of each individual robot in the system. As such, macroscopic models are generally more scalable and efficient in the calculation of system-level behaviors even as the studied MRS consists of increasingly larger numbers of robots.

A macroscopic mathematical MRS model has been demonstrated in a foraging task domain (38). The model was used to study the effects of interference between robots, the results of which could be used to modify individual robot control or determine the optimal density of robots in order to maximize task performance. A macroscopic analytical model has been applied to the study of the dynamics of collective behavior in a collaborative stick-pulling domain using a series of coupled differential equations (39).

A general macroscopic model for the study of adaptive multi-agent systems was presented in (40) and was applied to the analysis of a multi-robot adaptive task allocation domain that was also addressed experimentally in (34). In this work, the robots constituting the MRS maintain a limited amount of persistent internal state to represent a short history of past events but do not explicitly communicate with other robots.

4.2 Analysis of Multi-Robot Systems Using Microscopic Models

In contrast, microscopic modeling approaches directly consider each robot in the system and may model individual robot interactions with other robots and with the task environment in arbitrary detail, including simulating the exact behavior of each robot. However, most microscopic approaches model the behavior of each robot as a series of stochastic events. Typically, the individual robot controllers are abstracted to some degree and exact robot trajectories or interactions are not directly considered.

A microscopic probabilistic modeling methodology for the study of collective robot behavior in a clustering task domain was presented in (41). The model was validated through a largely quantitative agreement in the prediction of the evolution of cluster sizes with embodied simulation experiments and with real-robot experiments. The effectiveness and accuracy of microscopic and macroscopic modeling techniques compared to real robot experiments and embodied simulations was discussed in (42). Furthermore, a time-discrete, incremental methodology for modeling the dynamics of coordination in a distributed manipulation task domain was presented in (43).

4.3 Principled Synthesis of Multi-Robot System Controllers

One step beyond methodologies for the formal analysis of a given MRS design lie formal methodologies for the synthesis of MRS controllers. *Synthesis* is the process of constructing a MRS controller that meets design requirements such as achieving the desired level of task performance while meeting constraints imposed by limited robot capabilities. Being able to define a task domain and then have a formal method that designs the MRS to accomplish the task while meeting the specified performance criteria is one of the long-term goals of the MRS community.

An important piece of work in the formal design of coordinated MRS was the development of information invariants, which aimed to define the information requirements of a given task and ways in which those requirements could be satisfied in a robot controller (44). Information invariants put the design of SRS and MRS on a formal footing and began to identify how various robot sensors, actuators, and control strategies could be used to satisfy task requirements. Furthermore, the work attempted to show how these features were related and how one or more of these features could be formally described in terms of a set of other features. The concept of information invariants was experimentally studied in a distributed manipulation task domain (45) and was extended through the definition of equivalence classes among task definitions and robot capabilities to assist in the choice of appropriate controller class in a given domain (46).

There has also been significant progress in the design of a formal design methodology based on a MRS formalism that provides a principled framework for formally defining and reasoning about concepts relevant to MRS: the world, task definition, and capabilities of the robots themselves, including action selection, sensing, maintenance of local and persistent internal state, and broadcast communication from one robot to all other robots (47). Based on this formalism, the methodology utilizes an integrated set of MRS synthesis and analysis methods. The methodology includes a suite of systematic MRS synthesis methods, each of which takes as input the formal definitions of the world, task, and robots *sans* controller and outputs a robot controller designed through a logic-induced procedure. Each of the synthesis methods is independent and produces a coordinated MRS through the use of a unique set of coordination mechanisms, including the use of internal state (48), inter-robot communication (34), and/or deterministic and probabilistic action selection. Complimentary to the synthesis methods, this methodology incorporates both macroscopic (47) and microscopic MRS modeling approaches. Together, the synthesis and analysis methods provide more than just pragmatic design tools. A defining feature of this design methodology is the integrated nature of the controller synthesis and analysis methods. The fact that they are integrated allows for the capability to automatically and iteratively synthesize and analyze a large set of possible designs, thereby resulting in more optimal solutions and an improved understanding of the space of possible designs. This principled approach to MRS controller design has been demonstrated in a sequentially constrained multi-robot construction task domain (34,47,48).

A theoretical framework for the design of control algorithms in a multi-robot object clustering task domain has been developed (49). Issues addressed in this formalism include how to design control algorithms that result in a single final cluster, multiple clusters, and how to control the variance in cluster sizes.

Alternative approaches to the synthesis of MRS controllers can be found in evolutionary methods (50) and learning methods (33,51). There also exist a number of MRS design environments, control architectures, and programming languages which assist in the design of coordinated MRS (52,53,54).

5. Conclusions and the Future of Multi-Robot Systems

Behavior-based (BB) control has been a popular paradigm of choice in the control of multi-robot systems (MRS). The BB control methodology represents a robust and effective way to control individual as well as multiple robots. In a MRS, the task environment is inherently dynamic and non-linear as a result of the numerous types of interactions between the individual robots and between the robots and the task environment. This makes complex control strategies relying on accurate world models to perform computationally complex reasoning or planning ineffective. BB control provides a tight coupling between sensing and action and does not rely on the acquisition of such world models. As such it is a very effective control methodology in the dynamic and unstructured environments in which MRS inherently operate.

BB MRS have been empirically demonstrated in a diverse array of task domains -- from foraging, to object clustering, to distributed manipulation, to construction. Each of these task domains requires some overarching mechanism by which to coordinate the interactions of the individual robots such that the resulting system-level behavior is appropriate for the task. We have described and illustrated three different mechanisms to achieve this coordinated behavior: interaction through the environment, interaction through sensing, and interaction through communication. Each provides a coordination scheme capable of organizing the individual robot's behaviors toward system-level goals.

Another advantage of BB MRS is their amenability to formal analysis and synthesis. Due to their rather straight-forward and direct coupling of sensing to action, formal methods of synthesis and analysis become tractable and effective in producing and predicting the system-level behavior of a BB MRS.

The future possibilities and potentials of BB MRS are seemingly unlimited. As technology continues to improve and the nature and implications of different strategies for coordination are better understood, more task domains will become valid candidates for the application of MRS solutions.

Authors' Biographies

Chris Jones is a Robotics Researcher at iRobot Corporation. He received his Ph.D. in Computer Science in May 2005 from the University of Southern California, where he worked on multi-robot coordination in the Interaction Lab. From 1999-2001, he was a

Member of the Technical Staff in the Intelligent Systems and Robotics Center at Sandia National Laboratories. He has worked in the areas of robotics motion planning, mission planning, multi-robot control and coordination, and formal methodologies for multi-robot system controller design.

Maja Matarić is an associate professor in the Computer Science Department and Neuroscience Program at the University of Southern California, founding director of USC's interdisciplinary Center for Robotics and Embedded Systems (CRES) and co-director of the USC Robotics Research Lab. Prof. Mataric´ received her PhD in Computer Science and Artificial Intelligence from MIT in 1994, MS in Computer Science from MIT in 1990, and BS in Computer Science from the University of Kansas in 1987. She is a recipient of the Okawa Foundation Award, the USC Viterbi School of Engineering Service Award, the NSF Career Award, the MIT TR100 Innovation Award, the IEEE Robotics and Automation Society Early Career Award, the USC Viterbi School of Engineering Junior Research Award, and the USC Provost's Center for Interdisciplinary Research Fellowship, and is featured in the Emmy Award-nominated documentary movie about scientists, "Me & Isaac Newton." She is an associate editor of three major journals: International Journal of Autonomous Agents and Multi-Agent Systems, International Journal of Humanoid Robotics, and Adaptive Behavior. She has published over 30 journal articles, 17 book chapters, 4 edited volumes, 94 conference papers, and 23 workshop papers, and has two books in the works with MIT Press. She is active in educational outreach and is collaborating with K-12 teachers to develop hands-on robotics curricula for students at all levels as tools for promoting science and engineering topics and recruiting women and under-represented students. Her Interaction Lab pursues research aimed at endowing robots with the ability to help people through assistive interaction.

6. References

(1) Matarić, M. (1997). Behavior-Based Control: Examples from Navigation, Learning, and Group Behavior. Hexmoor, H., Horswill, I., and Kortenkamp, D., eds., Journal of Experimental and Theoretical Artificial Intelligence, special issue on Software Architectures for Physical Agents 9(2-3):323-336.

(2) Arkin, R. (1998). Behavior-Based Robotics. Cambridge, MA: MIT Press.

(3) Rosenchein, S., Kaelbling, L. (1995). A Situated View of Representation and Control. Special Issue on Computational Research on Interaction and Agency, Elsevier Science, pp. 515-540.

(4) Brooks, R. (1990). Elephants Don't Play Chess. Designing Autonomous Agents 6(1-2). MIT Press, pp. 3-15.

(5) Brooks, R. (1991). Intelligence Without Representation. Artificial Intelligence 47:139-160.

(6) Brooks, R., Connell, J. (1986). Asynchronous Distributed Control Systems for a Mobile Robot. In Proceedings of SPIE's Cambridge Symposium on Optical and Optoelectronic Engineering. Cambridge, MA, pp. 77-84.

(7) Connell, J., (1992) SSS: A Hybrid Architecture Applied to Robot Navigation. In Proceedings IEEE Conference on Robotics and Automation, Nice, France, pp. 2719-2724.

(8) Gat, E., (1998). On Three-Layer Architectures. In Artificial Intelligence and Mobile Robotics, AAAI Press, pp. 195-210.

(9) Agre, P., Chapman, D. (1987). Pengi: An Implementation of a Theory of Activity. In Proceedings of American Association of Artificial Intelligence (AAAI-87), Seattle, WA, pp. 268-272.

(10) Brooks, R. (1986). A Robust Layered Control System for a Mobile Robot. IEEE Journal of Robotics and Automation 2(1):14-23.

(11) Maes, P. (1989). The Dynamics of Action Selection. In Proceedings of the International Joint Conference on Artificial Intelligent (IJCAI-89). Detroit, MI, pp. 991-997.

(12) Pirjanian, P. (2000). Multiple Objective Behavior-based Control. Robotics and Autonomous Systems 31(1-2):53-60.

(13) Payton, D., Keirsey, D., Kimble, D., Krozel, J., Rosenblatt, J. (1992). Do Whatever Works: A Robust Approach to Fault-tolerant Autonomous Control. Applied Intelligence 2(3):225-250.

(14) Maes, P.,Brooks, R. (1990). Learning to Coordinate Behaviors. In Proceedings of the American Association of Artificial Intelligence (AAAI-91), Boston, MA, pp. 796-802.

(15) Pirjanian, P. (1999). Behavior Coordination Mechanisms – State-of-the-Art. University of Southern California Institute of Robotics and Intelligent Systems IRIS Technical Report IRIS-99-375.

(16) Matarić, M. (1992). Integration of Representation Into Goal-Driven Behavior-Based Robots. In IEEE Transactions on Robotics and Automation 8(3):304-312.

(17) Nicolescu, M., Matarić, M. (2001). Experience-based Representation Construction: Learning from Human and Robot Teachers. In Proceedings of IEEE/RSJ International Conference on Robots and Systems (IROS-01). Maui, Hawaii, pp. 740-745.

(18) Nicolescu, M., Matarić, M. (2000). Extending Behavior-Based Systems Capabilities Using An Abstract Behavior Representation. University of Southern California Institute for Robotics and Intelligent Systems IRIS Technical Report IRIS-00-389.

(19) Cao, Y., Fukunaga, A., Kahng, A. (1997). Cooperative Mobile Robotics: Antecedents and Directions. Autonomous Robots 4:7-27.

(20) Dudek, G., Jenkin, M., Milios, E. (2002). A Taxonomy of Multirobot Systems. In Balch, T. and Parker, L., eds. Robot Teams: From Diversity to Polymorphism, Natick, Massachusetts: A.K. Peters, pp. 3-22.

(21) Balch, T., Parker, L., eds. (2002). Robot Teams: From Diversity to Polymorphism. A. K. Peters.

(22) Bohringer, K., Brown, R., Donald, B., Jennings, J. (1997). Distributed Robotic Manipulation: Experiments in Minimalism. In O. Khatib ed. Experimental Robotics IV, Lecture Notes in Control and Information Sciences 223, pp. 11-25. Berlin: Springer Verlag.

(23) Gerkey, B., Matarić, M. (2004). A Formal Analysis and Taxonomy of Task Allocation in Multi-robot Systems. International Journal of Robotics Research 23(9):939-954.

(24) Howard, A., Parker, L., Sukhatme, G. (2004). The SDR Experience: Experiments with a Large-scale Heterogenous Mobile Robot Team. In 9th International Symposium on Experimental Robotics. Singapore. June 18-20.

(25) Konolige, K., Fox, D., Ortiz, C., Agno, A., Eriksen, M., Limketkai, B., Ko, J., Morisset, B., Schulz, D., Stewart, B., Vincent, R. (2004). Centibots: Very Large Scale Distributed Robotic Teams. In 9th International Symposium on Experimental Robotics. Singapore. June 18-20.

(26) Rahimi, M., Pon, R., Kaiser, W., Sukhatme, G., Estrin, D., Srivastava, M. (2004). Adaptive Sampling for Environmental Robotics. In Proceedings of the International Conference on Robotics and Automation, pp. 3537-3544. New Orleans, Louisiana.

(27) Zhang, B., Sukhatme. G., Requicha, A. (2004). Adaptive Sampling for Marine Microorganism Monitoring. To appear in Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Sendai, Japan. October 30-September 2.

(28) Holland, O., Melhuish, C. (1999) Stigmergy, Self-organization, and Sorting in Collective Robotics. Artificial Life 5(2):173-202.

(29) Franks, N., Deneubourg J-L (1997) Self-organising Nest Construction in Ants: Individual Worker Behavior and the Nest's Dynamics. Animal Behaviour 54:779-796.

(30) Kube, C., Zhang, H. (1996). The Use of Perceptual Cues in Multi-robot Box-pushing. In Proceedings IEEE International Conference on Robotics and Automation (ICRA-96), pp. 2085-2090, Minneapolis, Minnesota.

(31) Jones, C., Matarić, M. (2004). Automatic Synthesis of Communication-Based Coordinated Multi-Robot Systems. In Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Sendai, Japan. October 30-September 2.

(32) Reynolds, C. (1987). Flocks, Herds, and Schools: A Distributed Behavior Model. Computer Graphics 21(4):25-34.

(33) Matarić, M. (1995). Designing and Understanding Adaptive Group Behavior. Adaptive Behavior 4(1):51-80.

(34) Jones, C., Matarić, M. (2003). Adaptive Division of Labor in Large-Scale Minimalist Multi-Robot Systems. In Proceedings of the IEEE/RSJ International Conference on Robotics and Intelligent Systems (IROS). Las Vegas, Nevada. pp. 1969-1974.

(35) Fredslund, J., Matarić, M. (2002). A General, Local Algorithm for Robot Formations. IEEE Transactions on Robotics and Automation, Special Issue on Multi-Robot Systems 18(5):837-846.

(36) Gerkey, B., Matarić, M. (2001). Principled Communication for Dynamic Multi-Robot Task Allocation. In Proceedings International Symposium on Experimental Robotics 2000. Waikiki, Hawaii, pp. 341-352.

(37) Parker, L. (1997). Behavior-Based Cooperative Robotics Applied to Multi-Target Observation. In Bolles, R., Bunke, R., Noltemeier, H. eds. Intelligent robots: Sensing, modeling, and planning, R. Bolles, H. Bunke, and H. Noltemeier. World Scientific. pp. 356-373.

(38) Lerman, K., Galstyan, A. (2002). Mathematical Model of Foraging in a Group of Robots: Effects of Interference. Autonomous Robots 13(2):127-141.

(39) Lerman, K., Galstyan, A., Martinoli, A., Ijspeert, A. (2001). A Macroscopic Analytical Model of Collaboration in Distributed Robotic Systems. Artificial Life 7(4):375-393.

(40) Lerman, K., Galstyan, A. (2003). Macroscopic Analysis of Adaptive Task Allocation in Robots. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems. Las Vegas, Nevada. pages 1951-1956.

(41) Martinoli, A., Ijspeert, A., Mondada, F. (1999). Understanding Collective Aggregation Mechanisms: From Probabilistic Modeling to Experiments with Real Robots. Robotics and Autonomous Systems 29:51-63.

(42) Martinoli, A. Easton, K. (2002). Modeling Swarm Robotic Systems. In Siciliano, B. Dario, P. eds. Experimental Robotics VIII. Springer-Verlag, pp. 297-306.

(43) Martinoli, A., Easton, K., Agassounon, W. (2004). Modeling Swarm Robotic Systems: A Case Study in Collaborative Distributed Manipulation. Special issue on Experimental Robotics, International Journal of Robotics Research 23(4):415-436.

(44) Donald, B. (1995). Information Invariants in Robotics. Artificial Intelligence 72(1-2):217-304.

(45) Donald, B., Jennings, J., Rus, D. (1995). Information Invariants for Distributed Manipulation. In Goldberg, K., Halperin, D., Latombe, J.-C., Wilson, R. eds. International Workshop on the Algorithmic Foundations of Robotics. Boston, MA, pp. 431-459.

(46) Parker, L. (1998). Toward the Automated Synthesis of Cooperative Mobile Robot Teams. In Proceedings of SPIE Mobile Robots XIII, volume 3525. Boston, MA, pp. 82-93.

(47) Jones, C., Matarić, M. (2004). Synthesis and Analysis of Non-reactive Controllers for Multi-robot Sequential Task Domains. In Proceedings of the International Symposium on Experimental Robotics. Singapore. June 18-20.

(48) Jones, C., Matarić, M. (2004). The Use of Internal State in Multi-robot Coordination. In Proceedings of the Hawaii International Conference on Computer Sciences (HICCS), Waikiki, Hawaii, pp. 27-32.

(49) Zhang, A., Chung, M., Lee, B., Cho, R., Kazadi, S., Vishwanath, R. (2002). Variance in Converging Puck Cluster Sizes. In Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS). Bologna, Italy, pp. 209-217.

(50) Floreano, D. (1993). Patterns of Interactions in Shared Environments. In Proceedings of the Second European Conference on Artificial Life. Brussels, Belgium, pp. 347-366.

(51) Parker, L. (1998). ALLIANCE: An Architecture for Fault-tolerant Multi-robot Cooperation. Transactions on Robotics and Automation 14(2):220-240.

(52) Matarić, M. (1995). Issues and Approaches in the Design of Collective Autonomous Agents. Robotics and Autonomous Systems 16(2-4):321-331.

(53) Arkin, R., Balch, T. (1997). AuRA: Principles and Practice in Review. Journal of Experimental and Theoretical Artificial Intelligence 9:175-189.

(54) Alur, R., Grosu, R., Hur, Y., Kumar, V., Lee, I. (2000). Modular Specification of Hybrid Systems in CHARON. In Proceedings of the 3rd International Workshop on Hybrid Systems: Computation and Control. Pittsburgh, PA, pp. 6-19.