Autonomous Robots 4, 53–72 (1997)

Task Modelling in Collective Robotics

C. RONALD KUBE AND HONG ZHANG
Department of Computing Science, University of Alberta, Edmonton, Alberta, Canada T6G 2H1
kube@cs.ualberta.ca
zhang@cs.ualberta.ca

Abstract. Does coherent collective behaviour require an *explicit* mechanism of cooperation? In this paper, we demonstrate that a certain class of cooperative tasks, namely coordinated box manipulation, are possible without explicit communication or cooperation mechanisms. The approach relies on subtask decomposition and sensor preprocessing. A framework is proposed for modelling multi-robot tasks which are described as a series of steps with each step possibly consisting of substeps. Finite state automata theory is used to model steps with state transitions specified as binary sensing predicates called *perceptual cues*. A perceptual cue (Q), whose computation is disjoint from the operation of the automata, is processed by a 3-level finite state machine called a *Q-machine*. The model is based on entomological evidence that suggests local stimulus cues are used to regulate a linear series of building acts in nest construction. The approach is designed for a redundant set of homogeneous mobile robots, and described is an extension of a previous system of 5 box-pushing robots to 11 identical transport robots. Results are presented for a system of physical robots capable of moving a heavy object collectively to an arbitrarily specified goal position. The contribution is a simple task-programming paradigm for mobile multi-robot systems. It is argued that Q-machines and their perceptual cues offer a new approach to environment-specific task modelling in collective robotics.

Keywords: collective robotics, multirobot system, mobile robots

1. Introduction

It has been said that the era of the industrial robot—characterized by its use in the manufacturing industry—is about to make way for the next generation "service robot", a device with a high reliance on mobility to achieve its task specific purpose (Asami, 1994; Engelberger, 1989). Along with mobility comes a need for more autonomous operation than their progenitors whose lives were spent locked in cycles of precise movement in cartesian space. Autonomy and mobility in robotics usually leads to the problem of dealing with increasingly uncertain environments, and advances in autonomous robots are accomplished with a better understanding of the role sensors play in controlling actuators. However, greater autonomy often implies a complexity in the robot's control system which has led some researchers to consider designing systems as an aggregate of many simpler robots organized into collections or societies. Often inspired or motivated by the existence of insect societies, with its synergistic approach to achieving tasks through the mass effect of many, multi-robot systems have typically been studied performing cooperative tasks in which a "mechanism of cooperation" is employed usually involving explicit communication (Cao et al., 1995). Example multi-robot task domains normally fall under collective foraging (Arkin, 1992; Arkin et al., 1993; Altenburg and Pavivic, 1993; Balch and Arkin, 1994; Mataric, 1994; Parker, 1994), box-pushing (Caloud et al., 1990; Kube and Zhang, 1992; Noreils, 1992a; Donald et al., 1994; Mataric et al., 1995; Bay, 1995) and formation marching (Wang, 1991; Noreils, 1992c; Ozaki et al., 1993; Ota and Arai, 1993; Parker, 1993). In this paper, we demonstrate using a box-pushing task that certain classes of cooperative tasks involving coordinated manipulation are possible without cooperation mechanisms or explicit communication. The result is achieved with a subtask decomposition and sensor preprocessing methodology. Proposed is a framework in which sequential tasks—those consisting of a series of steps—are modelled using a hierarchy of finite state machines and implemented using stimulus-response behaviours.

The recent use of reactive control, often termed behaviour-based (Arkin et al., 1993), in mobile robot research has made ethological studies of particular relevance in the design of a collective robotic system. The study of social insects, nature's example of a loosely coupled decentralized control system, is an important starting point in understanding how to control complex systems. Social insects are capable of global task achieving behaviour based on simple rules of interaction and local perception (Wilson, 1971). A better understanding of how behaviours are sequenced, in natural biological systems, may help motivate and elucidate the mechanisms useful in designing synthetic multi-agent systems.

Sequential tasks composed of a series of steps might be performed by a multi-robot system if the architecture allows for a synchronization of efforts from individual asynchronous robots, a problem commonly found in a loosely coupled decentralized control system (Drouin et al., 1991). The problem can be stated as one of controlling state transitions in sets of asynchronous processes in a decentralized manner. A centralized approach would use a communications channel as a way of synchronizing state transitions. Such an approach relies on a convenient point in time for synchronization to occur and is not suitable for processes, like autonomous robots, with widely varying runtimes. However, multi-robot activity can still be coordinated without resorting to explicit communication by providing a common source of information which is locally sensed through the task's environment. This view is supported both by ethological (Downing and Jeanne, 1988) and multi-robot (Balch and Arkin, 1994; Kube and Zhang, 1993) studies.

Nest construction (Downing and Jeanne, 1988) and prey transport (Sudd, 1965) by some social insects are prime examples of tasks performed by a repetitive sequence of behaviours. Sensing plays a key role in triggering the transition between different task construction or transport behaviours. It is reasonable, therefore, to speculate that such a mechanism may also be used as a means of synchronizing several asynchronous robots in the execution of a common task.

The remainder of this paper is organized as follows. In Section 2, we review how tasks are specified by examining the types of collective tasks currently studied for multi-robot systems. In Section 3, we highlight some of nature's examples of tasks involving ordered sequences of behaviours invoked by locally sensed stimuli and how this approach might be applied in multi-robot systems. In Section 4, we present our task-programming framework and how a hierarchy of finite state automata can be controlled using binary sensing predicates derived from local sensor data. In Section 5 we present the results of a multi-robot transport task used to demonstrate the model's feasibility. Finally, in Section 6, we summarize our approach and outline our plan for future work.

2. Collective Tasks and their Ad Hoc Models

Research in autonomous robots has recently taken a new approach, namely, the multi-robot approach, in which systems are designed that distribute, to varying degrees, actuation and sensing, to perform tasks with or without some form of cooperation. Examples requiring distributed actuation include: floor washing, lawn mowing, vacuuming, crop harvesting or wall cleaning, all possible by one robot, without cooperation, given enough time. At the other end are the cooperative tasks requiring more than one robot; examples include heavy object transport, fluid containment, or fire control. Distributed sensing tasks require the system to perceptually cover an area spatially and might include such tasks as mine sweeping, mapping, searching or environmental monitoring. Both cooperative and non-cooperative forms of these tasks exist, and can be differentiated by imposed constraints such as whether a fixed formation needs to be maintained.

Designing systems for these tasks requires architectural decisions to be made. Architectures make use of either a heterogeneous (Parker, 1994; Noreils, 1992a; Caloud et al., 1990; Asama et al., 1991) or a homogeneous (Mataric, 1992; Altenburg and Pavivic, 1993; Kube and Zhang, 1992; Agah and Bekey, 1995) set of robots. Explicit communication between robots varies from none, to a fully connected topology where each robot has access to full global knowledge. System size ranges from a few robots to more than 100, with most systems reporting experimental results with physical implementations having less than 10 robots.

Tasks used in the study of multi-robot control include foraging, which involves searching and retrieving items from a given area, box-pushing, which moves an object between two locations, and formation marching, where robots move while maintaining a fixed pattern. These collective tasks have been studied using both physical robots and simulation. The questions under investigation abound. Should the composition be from a homogeneous or heterogeneous set of robots? What is the size of the system and how many robots are needed to accomplish the task? Does communication help improve task execution speed? What is the most effective control structure for a multi-robot system? Answers to these questions may come from the exploration of the above tasks.

2.1. *Collective Foraging*

In collective foraging robots search and retrieve items distributed in their environment. Objects to be retrieved are either randomly distributed or clustered at one location, and are usually small enough to be picked up by one robot and carried back to some central location. The analogous behaviour in nature, and presumably where foraging gets its name, is found among ant colonies. Ant foraging behaviour is typically used for food retrieval, with small objects carried by either a solitary ant or in group transport for larger items (Moffett, 1992). The envisioned practical applications of multi-robot foraging are cleanup tasks, harvesting in agriculture or mine sweeping in military applications.

The foraging task is often used to study some aspect of collective control and how a particular dimension of the problem like communications or architecture affect performance. For example, the effect communication has on task execution speed in foraging has been studied both in simulation (Arkin and Hobbs, 1992; Arkin et al., 1993; Balch and Arkin, 1994) and with physical implementations (Altenburg and Pavivic, 1993; Altenburg, 1994) with some agreement that a form of broadcasted target location improves execution times. As well, the question of which primitive behaviours are suitable for foraging (Mataric, 1994) or the reliability of heterogeneous systems (Parker, 1994) has also been studied.

Modelling the foraging task so that it may be implemented has taken several forms. Variations on the subsumption architecture (Brooks, 1986) have described tasks as either sets of behaviours, activated in a nonsequential manner by suppressing nonactive behaviour sets (Parker, 1994) or as a composite behaviour from a collection of basic behaviours (Mataric, 1994). Prioritized rules which describe foraging and can be triggered by either locally sensed cues, broadcast signals, or by timeouts, provide a reflexive task description (Altenburg, 1994). Another reactive method employed is schema-based control (Arkin, 1989) in which motor schemas are combined in sets which represent one of several states (Balch and Arkin, 1994). For the most part, task modelling is a secondary issue left to the implementation stages of the early research study.

2.2. *Multi-Robot Box-Pushing*

The above examples of foraging are noncooperative collective tasks, in that they could be performed by one robot given enough time. On the other hand, cooperative tasks such as box-pushing and formation marching, require at least 2 robots to complete the task. In box-pushing, both traditional AI and reactive approaches have been employed. Box-pushing requires a cooperative effort from at least two robots to move a box along some trajectory. The equivalent task in nature is often referred to as prey or food transport (Sudd, 1960, 1965; Moffett, 1988) and has been more recently identified as a behaviour worthy of study in its own right (Moffett, 1992). The practical applications of this form of cooperative behaviour are conjectured to be found in a task like material handling (Stilwell and Bay, 1993; Doty and Aken, 1993; Bay, 1995).

Traditional approaches decompose the box-pushing task into subtasks to be allocated to individual robots for execution. One such system uses a centralized task planner to communicate with 3 robots executing a box-pushing task, and coordinates the planning and scheduling activities with a blackboard system (Caloud et al., 1990). An alternate approach uses a decentralized architecture with functional, control and planning levels designed to decompose and allocate the task to a group of robots. An experiment in box-pushing using 2 robots, one to push and one to supervise, is presented in Noreils (1992a). A method dependent on the number of robots is described in a box-pushing protocol for a pair of identical robots with and without explicit communications (Donald et al., 1994). The approach relies on each robot knowing whether it is on the left or right side making it difficult to scale to larger systems. In each of these approaches planning is performed either globally with plans communicated to single robots or some combination of global and local planning with conflict resolution being handled centrally.

Another approach makes use of a reactive system in which planning has been precompiled into the task description itself in the form of reflexive behaviours. The approach has been tested in a box-pushing experiment using a decentralized noncommunicating homogeneous system of 5 mobile robots (Kube and Zhang, 1993).

2.3. Formation Marching

An example of a cooperative task requiring several robots is formation marching. In formation marching a group of robots move while maintaining a desired physical arrangement. Both schools of fish and flocks of birds move in formation. In robots this behaviour is thought to be useful in such tasks as map making, mine sweeping, sample taking or any task in which a physical area must be spatially covered.

Formation marching studies involving physical robots are few, but some have reported on experiments using 2 mobile robots moving in tandem (Noreils, 1992c; Ozaki et al., 1993). Other formation marching studies using simulation have examined motion based on nearest neighbour tracking (Wang, 1991), local path planning (Shibata and Fukuda, 1993), virtual impedance (Ota and Arai, 1993), and combinations of local and global information (Parker, 1993).

Task modelling has taken either a traditional approach, in which the system makes use of centralized planners to decompose and allocate the subtasks to individual robots for local planning (Noreils, 1992c; Asama et al., 1991), or a reactive approach in which both local and global information is used to maintain a prescribed formation (Parker, 1993).

2.4. Ad Hoc Modelling

For the most part little attention has been paid to the problem of task modelling, with most studies concentrating on other aspects of multi-robot control. As a result, most task modelling is performed with ad hoc solutions with results gathered for specific dimensions of the problem under study. The foraging task has been modelled using a set of basic behaviours (Mataric, 1994), prioritized rules (Altenburg, 1994), and finite state acceptors (Balch and Arkin, 1994). Box-pushing has been modelled using reflexive behaviours (Kube and Zhang, 1992), non-linear behaviour sets triggered by motivational behaviours (Parker, 1994), and coordinated protocols implemented as predicate/transition nets (Noreils, 1992a). In formation marching both traditional top down approaches (Noreils, 1992c; Asama et al., 1991), and bottom up reactive models have been used (Parker, 1993). An alternate approach to modelling tasks that does not suffer from ordered behaviour sequencing, and presents several analogies to the proposed framework, has recently been applied to perceptual tasks (Arkin and MacKenzie, 1994).

From the material presented in this section, we see in this emerging field that the problems and approaches are still being defined. System composition, communication, and control structure all play major roles in accomplishing tasks with multi-robot systems. From the examples presented it would seem that systems composed of heterogeneous robots require methods that can decompose a task and plan its allocation based on the individual skills of the robots. This often requires some form of negotiated cooperation with results being explicitly communicated. Whereas homogeneous systems, consisting of robots with identical skill sets, alleviate the need for task decomposition and allocation. Explicit communication between robots can improve execution in tasks which involve distributed actuation, such as in foraging, whereas little improvement is made if implicit communication through the task itself is possible as found in box-pushing or spatial coverage tasks (Balch and Arkin, 1994).

The control structure of a system seems to be intimately linked with the complexity of a task. Increasing task complexity would seem to require a means of describing the task in a way that allowed for the temporal ordering of its steps. In simple tasks such as foraging and box-pushing this was possible with a few carefully chosen behaviours. With the methodologies presented it is not clear how these behaviours could be extended to accomplish tasks requiring several ordered steps. Since task modelling has not been a specific issue of study, little general framework exists. As the field of collective robotics matures the resolution of these issues will help illuminate the knowledge necessary for achieving tasks collectively.

3. Collective Tasks in Nature

Examples abound in nature supporting the conjecture that locally sensed stimulus and reflexive behaviour can produce a predictable global effect. An example is the well defined mushroom shaped termites nest that often stand more than two meters high and one meter at its base (Sudd, 1963). Its construction, through a linear series of building steps, is hypothesized to be the result of a building program and stimulus cues used to switch between construction steps, and forms the basis of Grassé's Stigmergy Theory (Grassé, 1959). Can the many examples of perceptual cues, used to trigger behaviour sequences in biological systems, be used to design a similar mechanism for multi-robot control? And can these same cues also be used to govern transitions between task steps in robotic systems the same way they regulate building acts in nest construction? In this section we examine a number of examples with the above two questions in mind.

3.1. Perceptual Cues in Nature

Can a complex decision making process be reduced to simple sensor preprocessing? Wehner argues that an animal's solution to perceptual tasks "is often restricted to a narrow range of stimuli and situations" found in its environment (Wehner, 1987). Insects are cited as a prime example of this "matched filter" approach to perception, in which sensing receptors are spatially arranged to match some environmentally specific stimulus. This, Wehner conjectures, relieves the insect from any heavy computational task by solving the decision making process at the sensory level. Further, he speculates that although this makes the system less general in its ability to handle a variety of sensing input, the information processing is easier and suitable for the "narrow ecological niches" insects occupy.

Two examples Wehner cites, in support of this hypothesis, are the *visual streaks* found in the eyes of both desert ants (Wehner and Srinivasan, 1984) and crabs (Zeil et al., 1986). Visual streaks refer to the close spacing of photoreceptors near the center of the eye, the area in which most retinal images are formed because of the horizon dominated world which the animals inhabit. In crabs this spatial arrangement allows a constant number of receptors to be stimulated by objects of the same size irrespective of the distance the object is from the eye. Wehner speculates that this mechanism is a simple solution to the animal's problem of determining constant size, when the retinal images appear in a predictable way due to the predominantly flat visual environment (Wehner, 1987). It is also speculated that such a mechanism may be used in part as a visual cue and subsequent response to predators (Zeil et al., 1986).

Another example of behaviour triggering by stimulus cues is found in the social insects. Both bees and ants use dawn and dusk to start and stop their foraging behaviour. In fact, bees have a special sensor system consisting of three ocelli used to detect light level intensity and manipulating these sensors affect when the foraging behaviour begins and ends. The light-level threshold at which foraging is triggered can be varied by blinding one, two or three of the ocelli (Schricker, 1965).

Behavioural sequences may be invoked by one or more stimulus cues. A single stimulus which triggers corpse removal behaviour in ants was found and tested by daubing bits of paper with the acetone extracts from ant corpses, causing them to be removed from the nest and dumped on the same refuse pile as dead ants (Wilson et al., 1958). Downing and Jeanne found that multiple cues are used to trigger nonlinear building behaviour in nest construction by paper wasps (Downing and Jeanne, 1990). MacFarland and Bösser cite the work of Baerends and Kruijt (1973) on egg retrieval behaviour found in herring gulls, in which several cues are used to recognize an egg rolled from its nest, and note this mechanism of adding cues is an application of the *law of heterogeneous summation*—in which diverse and independent stimuli have an additive effect on behaviour (Seitz, 1940)—proposed by Seitz (McFarland and Bosser, 1993).

3.2. Nest Construction: A Collective Sequential Task

Nest construction by social insects is a collective task involving a well defined sequence of construction steps. Construction by paper wasps takes place in two stages. In the first stage, a linear series of building acts, or behaviours, are used to construct a petiole, or stem, which holds the nest to the bottom of a horizontal surface, to which walls are added forming the first nest cell. In the second stage, a nonlinear series of building behaviours follow in which either the stem is reinforced, the first cell lengthened, or an additional cell is built (Downing and Jeanne, 1988).

The linear sequence of building acts are:

1. Substrate preparation;
2. Stem construction;
3. Flat sheet construction;
4. First cell construction.

Downing and Jeanne identified the stimulus cues that influenced the transitions between steps and noted that the cues remained consistent within an individual but varied between individuals. For example, they cite the transition between stem and flat sheet construction to be the length of the stem, and that although this length varied among each wasp, an individual wasp would consistently build stems of the same length.

The decisions in the second nonlinear phase of nest construction are more complex since they involve a choice between any one of several building behaviours. Cues used in this phase were composed of more than one sensing stimulus. For example, when constructing the stem of the nest which holds it to a horizontal surface, the wasp measures both sides of the stem to determine its perpendicularity as well as using its reference to gravity (Downing and Jeanne, 1990).

Group transport is a behaviour found almost exclusively among ants in which prey or food is moved in a seemingly cooperative manner (Moffett, 1988; Franks, 1986; Sudd, 1960, 1965). In a detailed study of the strategies used in moving prey by a group of ants, Sudd determined that both realignment and repositioning behaviours were used whenever the transport item became stuck in deadlock (Sudd, 1960, 1965). Realignment involved changing the direction of applied force without the ant releasing its grasp and repositioning changed grasp locations and resulted in larger changes in force (Sudd, 1965). Application of these strategies was usually sufficient to resume movement. In this example, a fixed response of randomly applied directional changes in force was successful in recovering from a deadlock condition.

3.3. *From Social Insects to Multi-Robots*

From the above it can be seen that ethology has much to offer in motivating control mechanisms for multi-robot control. It would seem that nature has evolved a successful approach to the stimulus plethora on which task specific behaviours make their control decisions. How then do the examples presented relate to the design of multi-robot systems?

Locally sensed stimuli has a decision process problem called sensor aliasing. In other words, how do you control the perceptual problem of unique stimuli equating to the correct decision? It would seem, from the above examples, that nature has evolved at least four guiding principles useful in mitigating sensor aliasing:

- **Environment Specific.** By understanding or controlling the stimulus present in the environment, unique behaviour-specific sensors can be designed for the multi-robot system. This means that the environment characterized by its stimulus output is part of the overall solution, which results in an environment-specific robot system.
- **Task Decomposition.** Stimuli needs only to be unique within a subtask, resulting in context or behaviour dependent meaning. An example found in nest construction is the meaning of light intensity. While building enclosed walls light represents a hole to be patched, while in foraging it governs the start and stop of activity. For multi-robot systems this means sensor cues only need to be mutually exclusive to each subtask controller.
- **Orthogonal Stimulus.** Combining nonconflicting stimulus into decisions that govern the transition between behavioural acts reduces sensor aliasing. Multiple cues such as the use of both gravity and stem perpendicularity in wasp nest construction makes the cue unique.
- **Mass Effect.** Since individual behavioural acts are often found to be antagonistic towards progression of a task in nature's reactive systems, successful task completion must rely on mass effect to accomplish its goal. In homogeneous multi-robot systems this means using redundancy to increase the probability of successful task execution.

It remains to be seen, however, whether these perceptual cues can in fact be used to control transitions between task achieving behaviours in our artificial systems of robots. And it is the exploration of this question that motivates our current approach.

4. Multi-Step Task Decomposition

When the task to be performed involves a series of ordered steps, as in an assembly task, and we wish to design an autonomous system that executes the task, then we need to be able to describe and specify when to proceed with each step. Take for example the task of painting an aircraft carrier. Let us assume the task involves removing old paint, priming the surface, and applying new paint, a three step process. One thousand robots with magnetic wheels scour the surface like hungry locusts through a wheat field. An old-paint sensor keeps the scouring behaviour active until such time as a bare-metal sensor reaches its threshold and activates the paint-primer behaviour. Once the system, now empty of priming paint, determines the surface is dry through its dry-paint sensor, the last step begins and the apply-new-paint behaviour completes the operation.

Critical to such a system is the ability to locally perceive a stimulus which causes transitions between steps in a task. Simplification of these *perceptual cues* may be possible by considering the task's environment and combining behaviour specific sensors both in an *orthogonal* and *additive* manner as explained in the sequel. This approach to perception is based on the concept of "matched filters," in which a specific feature of the stimulus is used and, proposed in ethological studies (Wehner, 1987). Coupled with Finite State Automata (FSA) theory to model the individual steps in a task, and the inherent redundancy of a large homogeneous system of robots, we hypothesize that simple stimulus response mechanisms can be organized to produce a robust task specific multi-robot system. Here we describe the basic framework and in the sequel the approach for its evaluation in a transport task with a physical system of mobile robots.

4.1. *Perceptual Cues*

It has been said of insects that understanding the behavioural variety in any species begins with a complete analysis of its sensory physiology (Wilson, 1971). The same could be said of robots and a recent trend towards increasing the capabilities of robotic systems is the use of multiple sensors (Luo and Kay, 1989). How this is accomplished is an issue of multi-sensor integration and fusion. Our approach is through the use of *perceptual cues*, which combine information from *orthogonal* sensors in a manner which is *additive*. Two sensors are orthogonal to each other if they generate incomparable outputs, like temperature and touch, or if their spatial workspaces are non-overlapping such as two range sensors pointing in opposite directions. By additive we mean sensor features that are concatenated together to form a perceptual cue, a binary sensing predicate in Horn clause form. Our approach is based on ethological evidence of the role perception plays in triggering behaviour and we discussed some of the findings which have influenced our work in the previous section.

Sensors provide a robot with a window into its environment that effectively carves the world into a number of discrete perceptual spaces. The size of these spaces is dependent on a sensor's modality, resolution, feature extraction, data fusion and the number of sensors used. These are the issues of perception, which in a robot amount to sensing through a selection of sensors with widely varying parameters. Choosing the size of this window varies the amount of information, available to the robot, on which to make decisions.

In order to test our task modelling framework we have tried to simplify the stimulus used in specifying perceptual cues. We therefore endeavor to make all cues binary by employing either spatially or modally orthogonal sensing and combine the cues additively by simply concatenating their outputs. This decouples the need for sensor processing by a behaviour and renders the perceptual cue a simple behaviour "trigger" mechanism found so commonly in insects (Moser, 1970). In our first system of 5 box-pushing robots (Kube and Zhang, 1992) spatially orthogonal sensing was accomplished in hardware with the sensor's output equal to "1" when a threshold was reached. In the second generation system thresholding, as well as other nonlinear filters, is possible in software.

Sensing can be made *spatially orthogonal* in two ways. Sensors of the same type can be arranged geometrically so that their field of views (FOV) do not overlap, or by partitioning the sensor's FOV using thresholds as illustrated in Fig. 1.

In our model sensing is partitioned to match actuation. For example, for mobility our robots use two actuators: left and right wheel motors; therefore, the sensing used in navigation is divided into left and right side sensors which simplifies the mapping between sensors and actuators.

Sensing can also be made *modally orthogonal* by employing sensors that are complementary like contact and infrared sensors, or temperature and light intensity. Modally orthogonal sensing helps simplify the feature extraction process that takes place when a perceptual cue is used to trigger a behaviour. The trick of course is to be able to specify which combination of orthogonal sensors extract a feature from the environment, and we are in complete agreement with Wilson in the need for a more formal definition of environments in which a robot is operating (Wilson, 1990).

Once sensing has been reduced to a binary decision the outputs are concatenated together to form a perceptual cue. In this form, cues could be expressed using predicate calculus as Horn clauses with atoms representing the stimulus needed to satisfy the truth value of the clause. A perceptual cue is used to either trigger a specific behaviour into activity within a subtask controller or to cause transition between subtask

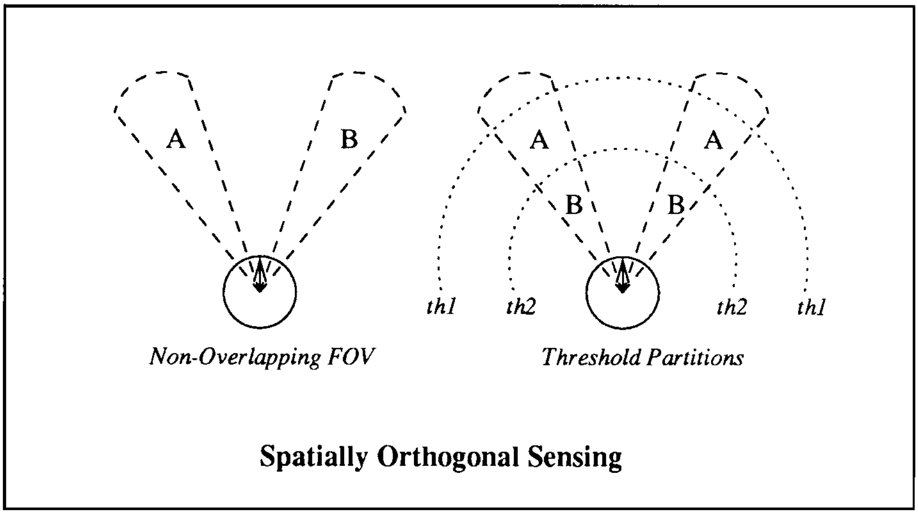


Figure 1. Sensing can be made spatially orthogonal by either arranging the same type of sensors geometrically with non-overlapping fields-of-view or by partitioning the field-of-view with thresholds.

controllers. This approach simply removes the decision making process from the controlling behaviour and "hardwires" it into the perceptual cue. We have used the above approach to create perceptual cues for a transport task and their design is more fully discussed in (Kube and Zhang, 1996).

4.2. Task Modelling

Our model assumes that the task under consideration can be described as an ordered sequence of steps. A finite state machine (FSM) will then be designed to accomplish each step with transitions between steps triggered by *perceptual cues*. Each step may, of course, be composed of sub-steps or subtasks also performed sequentially. In this manner a task may be described in as fine a detail as required by its decompositional analysis. This results in a task description having the hierarchical structure illustrated in Fig. 2.

Finite state automata have been used to model perceptual tasks (Arkin and McKenzie, 1994) and motivational behaviour in animals (Silby and McFarland, 1974; Metz, 1977). Arkin and McKenzie have used FSA to model the space time relationship in a perceptual processing task. This approach allows for perceptual tasks to be sequenced in a reactive control system. In animal behaviour McFarland and Bösser have defined motivational state as a combination of physiological and perceptual state, with behaviour used to change states in motivational space (McFarland and Bösser, 1993). They extended this approach to modelling the system behaviour by assigning state variables to environmental space, behaviour space and task space. Environmental space defines the constraints imposed on the system with regards to movement and topology. Behavioural space refers to the partition of the environment made by the animal's (or robot's) sensory system. Tasks are defined by their initial and final states using state variables that are relevant to the task.

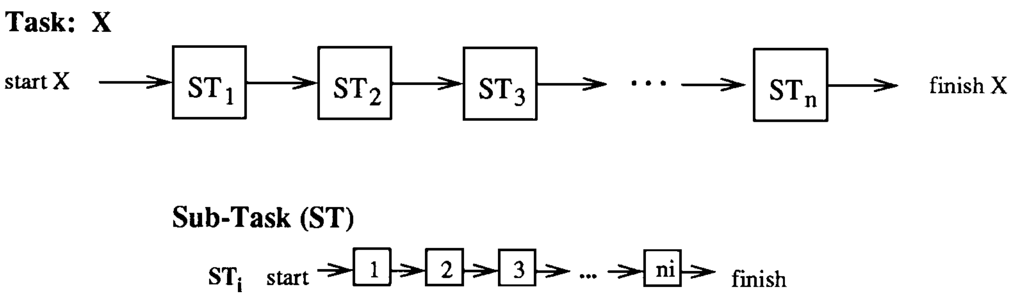


Figure 2. Tasks are described as an ordered sequence of steps, with each step possibly composed of additional subtasks (ST). Steps are modelled as FSM with transitions between steps specified as locally sensed perceptual cues.

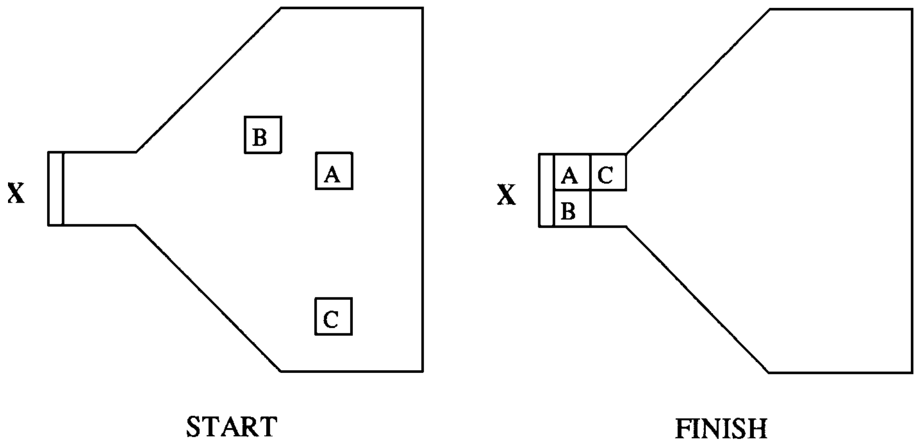


Figure 3. The three step transport task consists of 1) moving box A to location X; 2) moving box B to location X; and 3) moving box C to box A.

For illustration consider the contrived transport task shown in Fig. 3.

Imagine a maintenance system in which items (A, B, C) are moved to an exhaust port (location X). The task is to move the boxes labelled A and B to the narrow end of a room (labelled X) and then to do the same to the box labelled C. This task can be described in three steps that change its environment:

1. Move box A to location X (ST_1);
2. Move box B to location X (ST_2);
3. Move box C to box A (ST_3).

Of course the generality of location X would also allow boxes A and B to be reversed. If their order were important then location X would be specified as two rather than one stimulus in its corresponding perceptual cue. In the above description, Step 1 might be further described as three goal-seeking behaviours:

1. *Find-Box* (ST_{11});
2. *Move-to-Box* (ST_{12});
3. *Push-to-Goal* (ST_{14}).

If the initial analysis of the task shows that this subtask division is suitable, one would then proceed to designing a FSM that implements Step 1.

Next, we need to design a machine that will find a box in the robot's environment while avoiding obstacles. We decide to use a random search strategy and choose three stimulus response behaviours to implement the *Find-Box* FSM: *random-walk*, *avoid*, and *contact* (see Fig. 4). An alternate design might have used a more methodical search and movement generator in

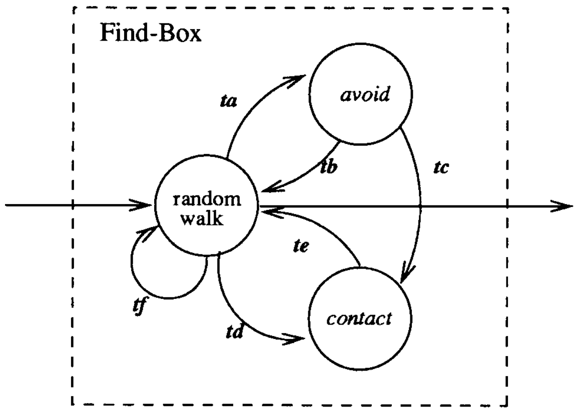


Figure 4. The second level of the task's model at which subtasks are described using goal-seeking behaviours. Here the *Find-Box* behaviour used to implement Part 1 of the first step in the transport task is modelled as a FSM. The perceptual cues used in state transitions, labelled as t_x, are binary outputs from the associated sensor subsystem. Each state's output is obtained from another FSM composed of primitive actuation behaviours.

place of the random-walk, or any one of the many strategies for spatial searching.

This task division is based on a division of task-related and tool-related knowledge proposed by McFarland and Bösser (1993). In their model, used to describe animal behaviour using FSA theory, task-related knowledge refers to the changes in the environment noticeable by external observation and is independent of the procedural mechanism used to accomplish the change. In the transport task under consideration this describes movement of boxes to their final destination. Tool-related knowledge describes how these changes are to be accomplished and equates to our

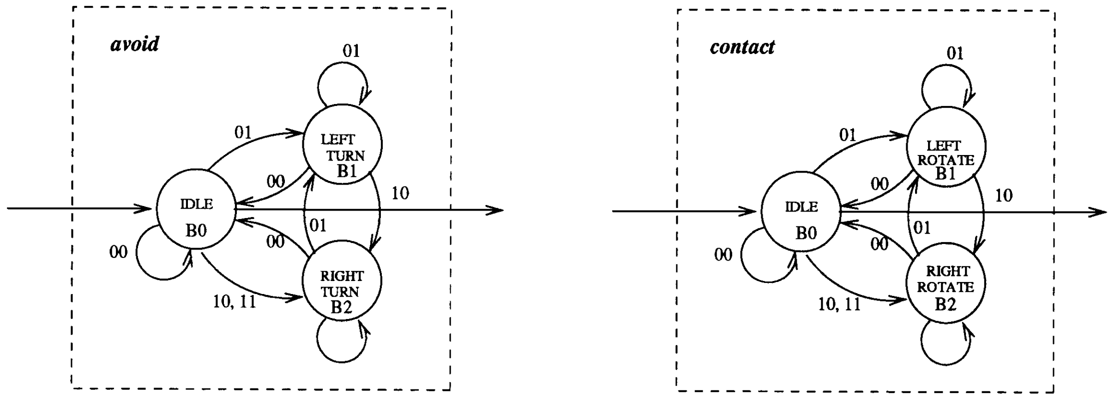


Figure 5. At the third level of the task model are the primitive actuation behaviours. The avoid and contact behaviours modelled as FSMs. These primitive actuation behaviours issue commands to the left and right wheel motors. Transitions are labelled as pairs of {*left*, *right*} binary outputs from the left and right avoid sensors. For example, in the left figure showing the avoid FSM, a transition occurs from state $B0$ idle to $B1$ when the avoid sensor detects an obstacle on its right side. The output from state $B1$ is the left-turn motion command which turns the robot by an amount equal to its sensor's field of view.

FSMs like *Find-Box*. Dividing a task in this manner allows different mechanisms to be employed in each part of the task model. For example, instead of a behaviour-based FSM as in the *Find-Box* subtask we could substitute a conventional AI controller. In our model we simply continue to use FSA theory for all our levels.

Each stimulus response behaviour is also modelled as a FSM. We refer to these stimulus response behaviours as *primitive actuation* analogous to Arkin's motor schemas (Arkin, 1989) or Matarić's basic behaviours (Mataric, 1994), except that they are not based on potential fields or subsumption, but rather more akin to the reflexive behaviours found in Braitenberg's vehicles (Braitenberg, 1984). Our original system of five box-pushing robots implemented these primitive actuation behaviours in combinational logic (Kube and Zhang, 1992). An example of the *avoid* and *contact* primitive actuation behaviours used in the *Find-Box A* FSM is shown in Fig. 5. Each primitive actuation behaviour has a stimulus specific sensor associated with it, and is used at the tool-related level (i.e., *Find-Box*) to cause transitions between each behaviour (i.e., avoid, contact and random walk). Other goal-seeking behaviour, such as *Move-to-Box*, *Push-to-Goal*, are derived in a similar fashion.

By specifying state transitions in terms of events that may be locally sensed (perceptual cues) we hope to obtain a predictable and somewhat synchronous system behaviour from a set of asynchronous robots. The approach is based on the conjecture that local sensing and implicit communication through the task is sufficiently rich in information and therefore does not warrant the need for an explicit mechanism of cooperation. The need for complex sensor processing is reduced by integrating many orthogonal sensors into the task specification. This approach is supported by the evidence seen within entomology, and discussed in the previous section, on specialized perceptual receptors for cue recognition. Tasks are specified in terms of initial and final states and are accomplished through a sequence of behaviours designed to recognize each step in the task. In this manner task complexity can be increased using an analytic approach to task decomposition, creating a control system for each set of initial and final states.

5. A Collective Transport System

Our interest lies in studying tasks for collective robotics that are inherently cooperative in nature, like box-pushing, material transport, construction, or sequential tasks not possible with a single robot. Applications of these multi-robot systems will eventually be found in the task specific areas of service robotics. Our reactive approach to control has been to minimize the processing requirements of the controlling behaviours by making use of a reflexive response to behaviour specific sensing—the so called "action-oriented perception" (Arkin, 1993a). The onus is on creating a sensor which correctly identifies the stimulus needed to trigger the behaviour. Being able to characterize the environment in terms of its stimulus output, as suggested by Wilson (1990), simplifies the sensor design and allows for a more predictive response from the system. We begin by briefly reviewing our previous work and then describe both our method of implementation and the results obtained from tests with our physical system.

5.1. *Our Early Work in Group Transport*

Our initial attempt at obtaining synchronous control from a set of asynchronous robots through local sensing resulted in a system of 5 box-pushing robots controlled by a group *box-pushing* behaviour consisting of the two reactive behaviours *avoid* and *goal* (Kube, 1992; Kube and Zhang, 1992). The system would locate and push the box in a direction that was primarily dependent on the initial configuration. We previously reported that for the *box-pushing* task, both execution time and average success rate, improved as system size increased up to a point which seemed dependent on robot diameter and box-size (Kube and Zhang, 1993). Simulation studies had also shown that such a system of robots could reach deadlock if an equal distribution of forces were to occur, particularly when the system size was small[1].

A similar problem found in prey transport among ants suggested that a possible solution was an ordered set of strategies which ultimately had the effect of introducing increasing amounts of randomness to the applied force, thereby breaking the deadlock condition, much in the same way that Arkin's *noise* motor schemas are used to overcome local minima in potential fields (Arkin, 1989). These strategies, termed stagnation recovery behaviours, consisted of a compound behaviour in which the strategies for varying the applied force were ordered and applied using timeouts. The recovery behaviours were either suppressed or triggered by a perceptual cue depending on the user defined deadlock condition (Kube and Zhang, 1994)[2]. In our reactive controllers stagnation recovery behaviours are a necessary component and their use will be shown in the sequel.

In order to evaluate our task-programming paradigm we have constructed a transport system consisting of 11 box-pushing robots. From a base set of motion primitives, designed for differential steering, primitive actuation behaviours are created from stimulus-response pairs designed for a given environment. The behaviours are used to create subtask controllers for each step in the transport task. Each controller is tested individually and then combined using perceptual cues into the integrated multi-step transport system. This type of control is analogous to phase-based control used in dextrous manipulation tasks (Tremblay and Cutkosky, 1995). Our objective in these experiments is to determine whether our multi-robot programming approach requires an explicit coordination mechanism to achieve the close cooperation needed in a manipulation task such as box-pushing.

5.2. *A Model for Motion*

The mobility hardware is composed of a set of homogeneous two-wheeled robots, each weighing three pounds and measuring approximately seven inches in height and diameter as shown in Fig. 6.

A battery allows for 45 minutes of operation with a 10 minute recharge time. Control electronics are separate modules plugged into the robot, allowing for quick maintenance in the event of failure. Task specific sensors are added to the generic base and can be attached onto a grid of evenly spaced holes. A Motorola 68HC11 microcontroller with 8K of RAM and programmed in Forth is used to map sensor output to one of nine motion primitives.

Continuous motion is accomplished by issuing a series of discrete motion commands, each of which moves the robot a small incremental amount. The commands have the general form: *begin*(*action*), *wait* Δt, *end*(*action*). The motion commands are: *stop*, *forward*, *backward*, *left-turn*, *right-turn*, *left-rotate*, *right-rotate*, *back-left* and *back-right* as shown in Fig. 7.

5.3. *Primitive Actuation Behaviours*

We call reactive behaviours which control a specific set of actuators, primitive actuation (PA) behaviours. PA behaviours simply map their inputs to one of the robot's motion commands. In box-pushing the only action a robot is capable of is movement in a plane. As a result, PA behaviours control direction and velocity. Since each motion primitive controls a left and right wheel motor, PA behaviours that change direction use left and right stimulus pairs. For motions used in box-pushing PA behaviours can be divided into three classes:

1. **Goal Driven.** Behaviours used to attract the robot towards a given stimulus or move in a fixed pattern. For box-pushing the defined behaviours are:
   - RANDOM-WALK—causes the robot to move forward in a "J" pattern.

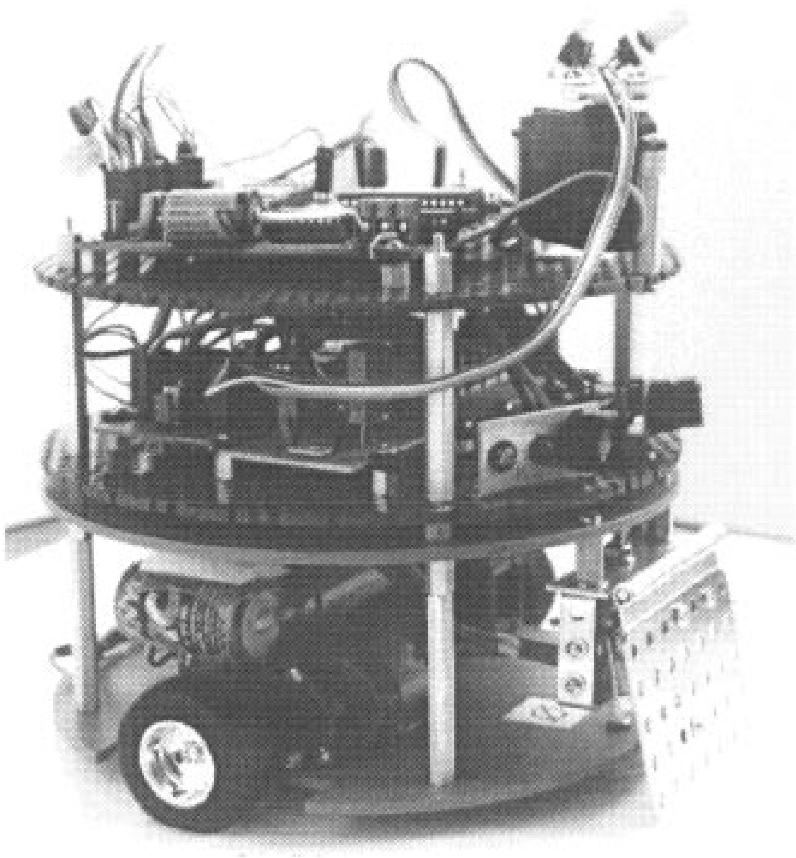


Figure 6. Each of the robots are equipped with 2 forward pointing infrared obstacle sensors, one touch sensor, 2 CdS box-tracking photocells, and a destination sensor, all mounted on a differentially steered base.

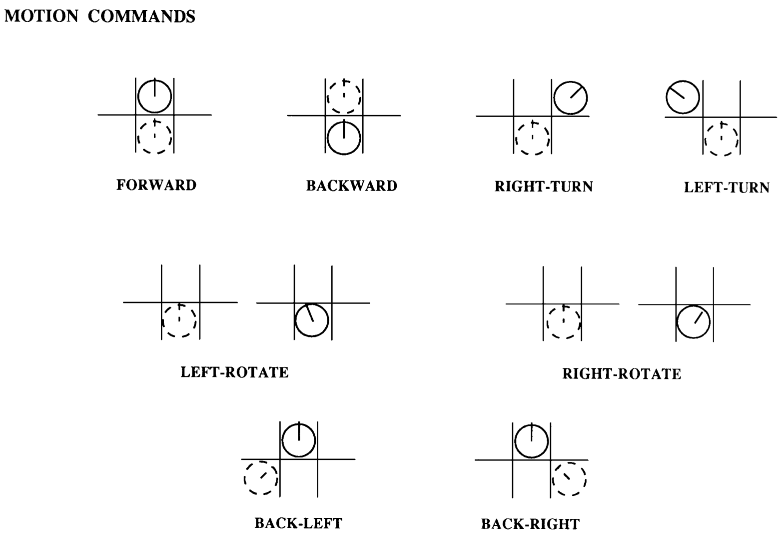


Figure 7. The discrete motions possible by issuing several motion commands. Initial positions are shown as dotted lines.

- *SEEK-BOX*—uses left and right light sensors to move the robot towards the brightly lit box.
- *PUSH-BOX*—increases wheel motor torque used to move the box.

2. **Avoidance Driven.** Behaviours which repel the robot from a given stimulus. In box-pushing the defined behaviours are:

   - *AVOID*—turns the robot away from obstacles sensed in the forward direction with left and right obstacle sensors at a distance of the robot's diameter.
   - *CONTACT*—rotates the robot away from obstacles sensed in the forward direction with left and right obstacle sensors at a distance of half the robot's diameter.

3. **Recovery Driven.** Behaviours used to recover from stagnating conditions. The defined behaviours in box-pushing are:

   - *BACK-OFF*—causes the robot to back away from objects contacted by its touch sensor.
   - *REPOSITION*—moves the robot in a backward arc.

Input stimulus for the PA behaviours is obtained from the robot's sensors consisting of:

- **Obstacle Sensors**—left and right obstacle sensors created using reflected infrared light, from a light emitting diode and phototransistor pair, with different thresholds for each behaviour. Positioned at 20 degrees off center and with a 10 degree field of view, the sensors can detect obstacles within a five foot range.
- **Box Direction Sensors**—used to locate a brightly lit box with two forward facing photocells whose resistance varies as a function of light intensity and are placed on the left and right side of the robot. The field of view has been sufficiently limited so that the sensor only views a narrow band roughly the height of the robot.
- **Touch Sensor**—a forward facing microswitch provides on/off contact sensing.
- **Goal Sensor**—a forward facing rotating phototransistor used to detect the direction of the goal where the box is to be pushed. The light sensor is pointed up at an elevation of 60 degrees, where the goal light is located, to avoid aliasing the box-light.

Once defined the PA behaviours are used to create the subtask controllers. At any given moment the robot's action is taken from the output of one PA behaviour.

5.4. *Sub-Task Controller Design*

The problem of transporting a heavy box from an unknown location, in a given environment, to a known goal destination can be divided into three subtasks:

- **Find the Box.** Search the environment for the box while avoiding collisions with obstacles.
- **Move to the Box.** Once the box is located within a robot's field of view, move towards the box while avoiding obstacles and bring the robot into contact with any side.
- **Push the Box Towards the Goal.** If the box is between the robot and the goal destination then push the box; otherwise reposition the robot to another spot on the box.

The reactive controller designed for each step consists of a method for achieving its task (goal driven PA behaviours), a method for dealing with impediments (avoidance driven PA behaviours), and a method for recovering from stagnation or deadlock conditions (recovery driven PA behaviours). A minimally designed controller must at least contain one PA behaviour from the goal and recovery classes. Controllers used for navigation will also include a PA behaviour from the avoidance class. A subtask controller can be modeled as a FSM with each state represented by a single PA behaviour. For the transport task the three subtask controllers (machines) are **FIND-BOX**, **MOVE-TO-BOX** and **PUSH-TO-GOAL**.

5.5. *Results of Sub-Task Controller Testing*

Each subtask controller was tested individually on the physical robots and then combined to form the final integrated controller used in the transport task.

FIND-BOX is a 4 state machine consisting of *RANDOM-WALK*, *AVOID*, *CONTACT* and *BACK-OFF* PA behaviours. To test the controller a felt-tipped pen is attached to the robot marking its traveled path. A single robot placed on a 2.7 meter square enclosed grid composed of 81 cells and visited 90% of the cells in 3 minutes. The fixed motion pattern generates a pseudo-random motion when the obstacle sensors change the path of the robot. The same controller tested with 11 robots in a 5

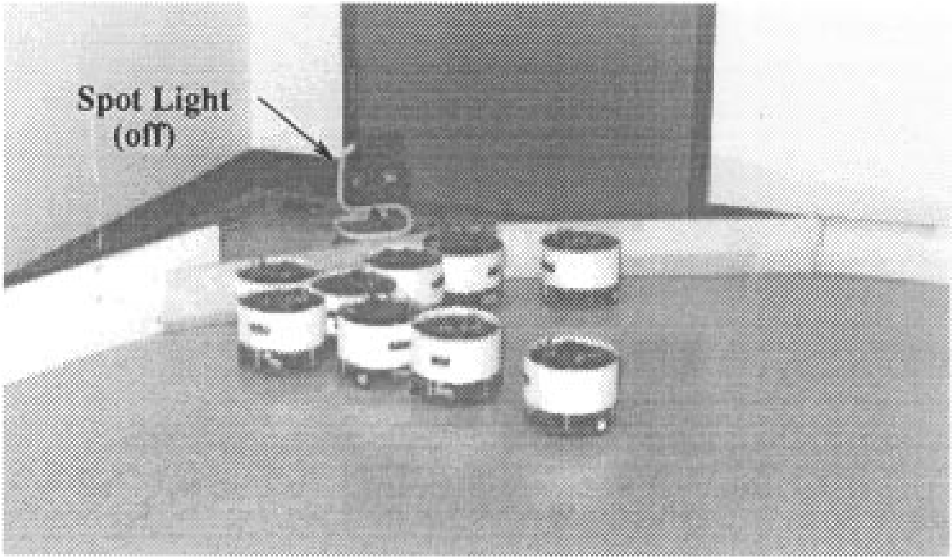


Figure 8. One test of the MOVE-TO-BOX controller involved marching 9 robots back and forth between two floor level lights turned on alternately and placed at opposite corners of the room. A white shell is added to each robot so that a reflective surface is available for the obstacle sensors.

by 6 meter room produces continuous stagnation-free motion in 10 minutes.

MOVE-TO-BOX is a 4 state machine composed of *SEEK-BOX*, *AVOID*, *CONTACT* and *BACK-OFF* PA behaviours. The controller was first tested using a single robot which followed a lit box as it was dragged around the room. Next the **FIND-BOX** machine was added and 10 robots were placed at opposite ends from the box in a 3 by 5 meter room. All robots located the box while obstacle avoidance created an even distribution around its circumference. Finally, floor level lights placed at opposite corners of the same room were used to march 9 robots back and forth across the floor in a simple homing experiment shown in Fig. 8. Interference between interacting robots is minimized by reducing the distance at which obstacles are detected.

PUSH-TO-GOAL is a 2 state machine using the *PUSH-BOX* and *REPOSITION* PA behaviours. The machine enters the *PUSH-BOX* state if the perceptual cue *?SEE-GOAL* is true. *?SEE-GOAL* is true if the box is between the robot and the destination goal as illustrated in Fig. 11. The cue is created using an upward pointing rotating sensor which detects signal peaks within a specified field of view illustrated in Fig. 9. To test the controller by itself two robots, needed to push the box, are positioned on a boxside and facing the goal indicator. The robots successfully push the box towards the goal. Next the robots are placed on a boxside facing away from the goal causing the *REPOSITION* state to move the robots to a new position and orientation. The final tests involve integrating the three subtask controllers described next.

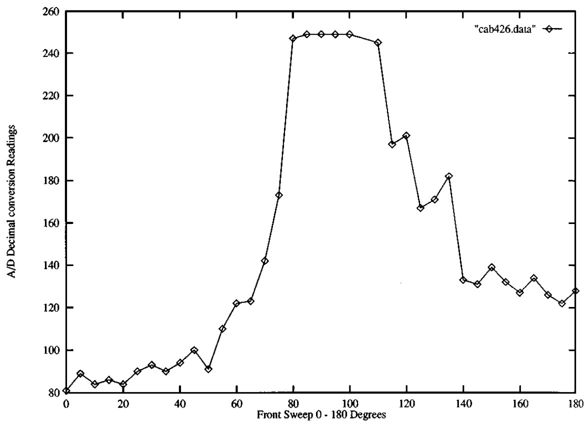


Figure 9. A signal peak occurs in a specified field of view when the box is between the robot and goal destination.

5.6. Results of Controller Integration

In order to integrate the subtask controllers into a single three state machine, with each state representing a single controller, we require a mechanism to control state transitions. Perceptual cues are the computationally disjoint mechanism used. The cues can be expressed as Horn clauses with each atom representing a stimulus needed to satisfy the truth of the clause. The integrated controller is simply a machine that processes perceptual cues (Q) to determine the state (or subtask controller) that controls the robot, hence we call them *Q-machines*. The Q-machine for the transport task is shown in Fig. 10. The perceptual cues used to determine state are:

- **?box** True when the forward box sensors detect a lit box.
- **?box-contact** True when the robot is in contact with a lit box.

The state of the transport machine is specified by:

1. **FIND-BOX** $\Leftarrow \overline{?\mathbf{box}}$
2. **MOVE-TO-BOX** $\Leftarrow ?\mathbf{box} \wedge \overline{?\mathbf{box}-\mathbf{contact}}$
3. **PUSH-TO-GOAL** $\Leftarrow ?\mathbf{box} \wedge ?\mathbf{box}-\mathbf{contact}$

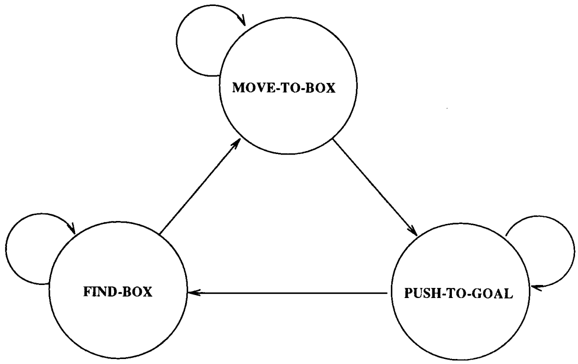


Figure 10. The task-level description of the controller for transporting a box by pushing it from an initially unknown position to a final goal destination.

The integrated controller was tested in a four by five meter room in four experiments totaling 33 trials, each lasting 3 minutes, using from one to six robots and two different sized boxes. The lab environment is depicted in Fig. 12 and shows the initial position and final goal destination. These initial experiments were used

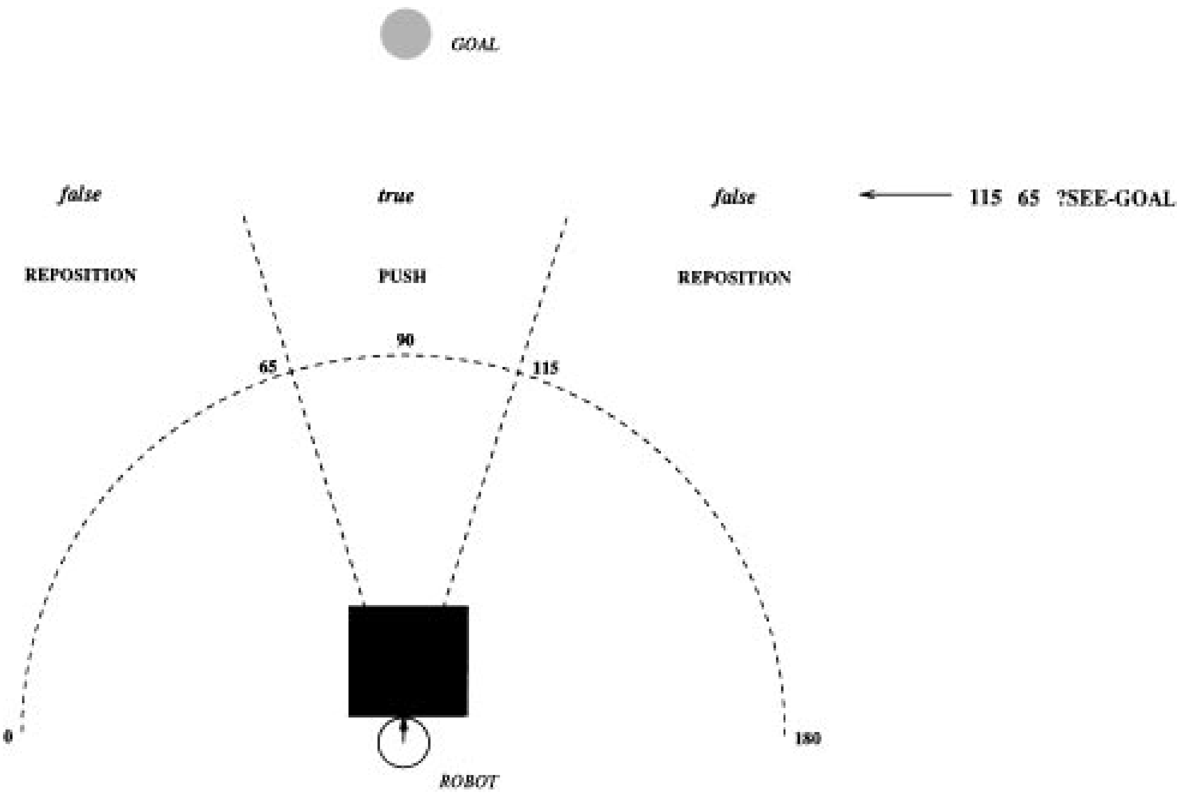


Figure 11. The actions taken by the PUSH-TO-GOAL machine depend on the ?SEE-GOAL perceptual cue. If the goal is within the sensor's field of view the machine is controlled by the PUSH-BOX PA behaviour, else control is passed to REPOSITION which causes the robot to locate another spot on the box.

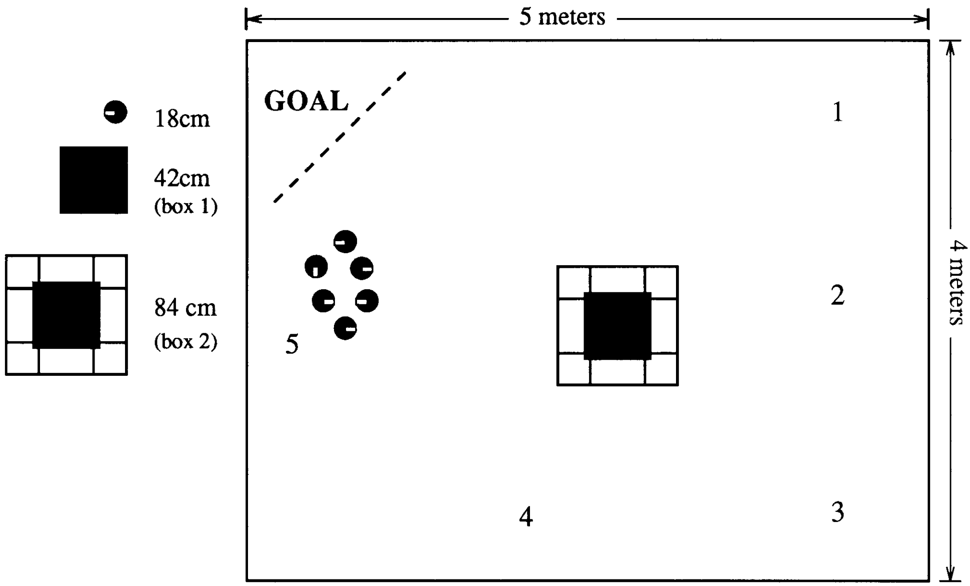


Figure 12. A schematic of the lab environment used to test the integrated transport controller. In each trial the box was placed at initial position 3 meters from the goal line and the robots were placed at one of the five indicated starting positions.

to evaluate and tune the PA behaviour parameters as well compare task performance using two different box sizes. Experiments 1 and 2 used a 42 cm square box and the robot's goal sensor field-of-view (FOV) set to 50 degrees. Experiment 3 increased the goal sensor's FOV to 120 degrees. Finally, Experiment 4 increased the box size to 84 cm per side. The results are summarized in Table 1 with the following additional comments:

Experiments 1 and 2. A successful trial was counted if the robots were able to move a 42 cm box from its initial position to the goal line, a distance of 2 meters, in under 3 minutes. In trials with a single robot the box was not translated, but changes in orientation were possible as the robot exerts a torque about the center of the box by grabbing one of the four corners. In trials with 2 robots synchronization in 3 minutes was difficult, but the box was moved towards the goal without reaching it. Trials with 3–4 robots were successful in over half the runs, with unsuccessful trials achieving some movement towards the goal. Using 5 robots increased the amount of robot interference since in successful trials typically 2–3 robots participated due to the limited size of a boxside.

Table 1. A summary of results of four multi-robot box-pushing experiments testing the integrated controller. In all successful cases the box was moved from its initial position to the final goal in under 3 minutes. Unsuccessful trials moved the box towards the goal but failed to complete the task in 3 minutes.

Size	# Successes	Completion time	Comments
Experiment 1 & 2—Boxsize: 42 cm, Goal FOV: 50 degrees			
1	—	—	Robot unable to translate box, but change orientation is possible.
2	—	—	Box moved 15 cm towards goal; synchronization is difficult.
3	5/9	3 min.	Unsuccessful trials couldn't complete in 3 minutes.
4	2/3	3 min.	Not more than 2–3 robots actively participate at a time.
5	2/6	2:30 min.	Duration of reposition behaviour increased.
Experiment 3—Boxsize: 42 cm, Goal FOV: 120 degrees			
6	2/3	2:30 min.	Larger goal sensor FOV increases persistence.
Experiment 4—Boxsize: 84 cm, Goal FOV: 120 degrees			
6	1/2	30 sec.	Larger box allows 3–4 robots to actively participate at a time.

Experiments 3 and 4. Increasing the goal sensor's FOV to 120 degrees increased the robot's pushing persistence (i.e., the robot would spend more time trying to push on a side) since its perceptual cue was true more often than when the FOV was set to 50 degrees. This helped to synchronize the group pushing behaviour. In the above experiments successful trials usually took 2–3 minutes to complete. In the last experiment, shown in Fig. 13, the box size was doubled which allowed more robots to participate, typically 3–4, and with faster completion times (30 seconds in one case), but only 2 trials were run before tires needed changing[3].

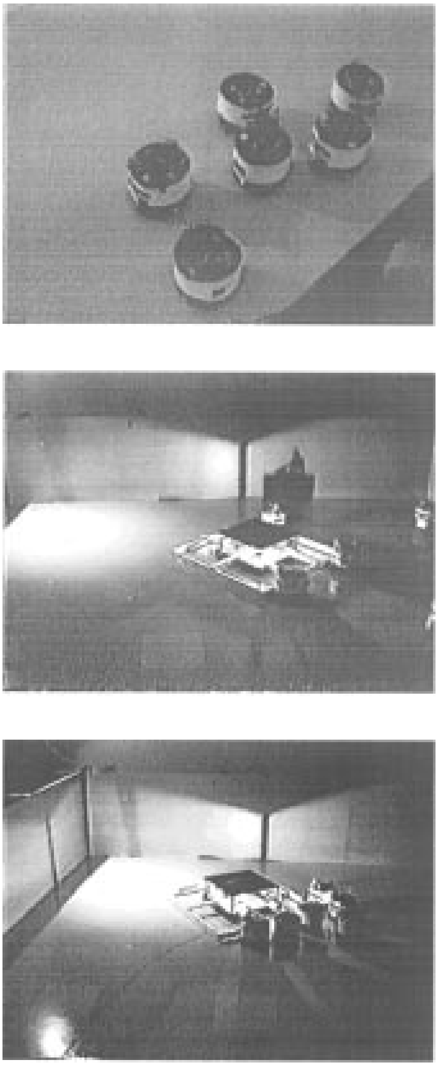


Figure 13. Shown are 6 robots pushing an enlarged box from its initial position 3 meters towards a final goal. The mpeg video from which this sequence was taken is available at http://www.cs.ualberta.ca/∼kube/.

After reviewing the videotaped experiments one is reminded of the antagonistic forces present in ant group transport (Moffett, 1988), yet the end result is invariably transport of the item back to the nest. In our multi-robot box-pushing experiments the path taken to the goal is often not the same nor optimal or continuous, as one would get in a centralized controller, but rather a feasible solution to the problem given the limited abilities of the individual robots. There are even temporary setbacks as the box is moved incorrectly. At times the robots can lose contact with the box, be blocked by other robots, or the box rotates forcing some robots to relocate the box. In the results presented, it can be said that in all cases the robots move the box towards the goal; however, there is ample room for improved performance once the robot parameters are fine tuned. Additional statistical conclusions can also be drawn with more experiments.

6. Summary and Future Work

In this paper, a novel multi-robot task-programming framework is proposed which models tasks with temporal sequences as a hierarchy of finite state machines with state transitions specified using a concatenation of binary environment-specific stimuli, the so called *perceptual cues*. The method does not include an explicit cooperation mechanism, but rather makes use of the mass effect present in a homogeneous system of mobile robots.

The approach is motivated by entomological studies on cues used to regulate linear sequences of behaviours in social insects, which suggest that cues may be constructed with behaviour specific sensing strategies in which decision processing occurs at the sensory level. In many cases, this simple stimulus thresholding triggers task achieving behaviour. In other examples, cues are formed from stimuli which are *orthogonal* in modality, such as the use of tactile and gravity sensing by paper wasps, or spatially as in non-overlapping sensor workspaces. The conjecture, as applied to collective robotics, is that perceptual cues can be used for both controlling primitive actuation behaviours and serving as a mechanism to regulate step transitions between controlling FSMs.

The proposed model's underlying *primitive actuation* (PA) behaviours, used to create subtask controllers that model steps in the collective task, along with a set of perceptual cues have been tested on physical box-pushing robots in a transport task. The results demonstrate that a feasible, versus optimal solution, exists using reactive control in cooperative tasks requiring close coordination without resorting to mechanisms of cooperation such as explicit communication.

Of interest, for future study, is the manner in which a perceptual cue can change over time or be adaptive to changes in the environment and task requirements. Entomological studies of nonlinear sequences of behaviours suggest that cues used to regulate the latter stages of nest construction vary in response to a changing environment (Downing and Jeanne, 1988). With its rich source of inspiring examples, social insects serve as nature's proof-by-example that solutions to complex tasks in autonomous collective robotics may in fact be found underfoot.

Notes

1. Small being a relative term based on the diameter of the robots, the size of the box, and the critical force under which the box can be moved.
2. For example, stagnation in box-pushing occurs if the robot is pushing without moving the box.
3. A robot pit stop?

References

Agah, Arvin and Bekey, George 1995. Learning by trying in an autonomous multi-robot system. In submitted to *IEEE/RSJ International Conference on Intelligent Robots and Systems*.

Altenburg, Karl 1994. Adaptive resource allocation for a multiple mobile robot system using communication. CSOR TR-9404, North Dakota State University.

Altenburg, Karl and Pavivic, Mark 1993. Initial results of the use of inter-robot communication for a multiple, mobile robotic system. In *Proceedings of the Dynamically Interacting Robots IJCAI-93 Workshop*.

Arkin, Ronald C. 1989. Motor schema based mobile robot navigation. *International Journal of Robotics Research*, 8(4):92–112.

Arkin, Ronald C. 1992. Cooperation without communication: Multiagent schema-based robot navigation. *Journal of Robotic Systems*, 9(3):351–364.

Arkin, Ronald C. 1993. Survivable robotic systems: Reactive and homeostatic control. In *Robotics and Remote Systems for Hazardous Environments*, M. Jamshida and P. Eicker (Eds.), Prentice Hall: Englewood Cliffs, New Jersey, vol. 1, no. 7, pp. 135–154.

Arkin, Ronald C. and Hobbs, J. David 1992. Dimensions of communication and social organization in multi-agent robotic systems. In *Second International Conference on Simulation of Adaptive Behavior*, J.A. Meyer, H. Roitblat, and S. Wilson (Eds.), MIT Press, pp. 486–493.

Arkin, Ronald C., Balch, Tucker, and Nitz, Elizabeth 1993. Communication of behavioral state in multi-agent retrieval tasks. In *1993 IEEE International Conference on Robotics and Automation*, vol. 3, pp. 588–594.

Arkin, Ronald C., and McKenzie, Douglas 1994. Temporal coordination of perceptual algorithms for a mobile robot navigation. *IEEE Transactions on Robotics and Automation*, 10(3):276–286.

Asama, Hajime, Habib, Maki K., Endo, Isao, Ozaki, Koichi, Matsumoto, Akihiro, and Ishida, Yoshiki 1991. Functional distribution among multiple mobile robots in an autonomous and decentralized robot system. In *1991 IEEE International Conference on Robotics and Automation*, pp. 1921–1926.

Asami, Shuji 1994. Robots in Japan: Present and future. *IEEE Robotics and Automation Magazine*, pp. 22–26.

Baerends, G.P. and Kruijt, J.P. 1973. Stimulus selection. In *Constraints on Learning: Limitations and Predispositions*, R.A. Hinde and J. Stevenson-Hinde (Eds.), Academic Press.

Balch, Tucker and Arkin, Ronald C. 1994. Communication in reactive multiagent robotic systems. *Autonomous Robots*, 1:1–25.

Bay, John S. 1995. Design of the army-ant cooperative lifting robot. *IEEE Robotics and Automation Magazine*, pp. 36–43.

Braitenberg, Valentino 1984. *Vehicles: Experiments in Synthetic Psychology*. MIT Press.

Brooks, Rodney A. 1986. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, RA-2(1):14–23.

Caloud, Philippe, Choi, Wonyun, Latombe, Jean-Claude, Pape, Claude Le, and Yim, Mark 1990. Indoor automation with many mobile robots. In *1990 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 67–72.

Cao, Y. Uny, Fukunaga, Alex S., Kahng, Andrew B., and Meng, Frank 1995. Cooperative mobile robotics: Antecedents and directions. In *1995 IEEE International Conference on Intelligent Robots and Systems*, 1:226–234.

Donald, Bruce, Jennings, James, and Rus, Daniela 1994. Analyzing teams of cooperating mobile robots. In *1994 IEEE International Conference on Robotics and Automation*, pp. 1896–1903.

Doty, Keith L. and Van Aken, Ronald, E. 1993. Swarm robot materials handling paradigm for a manufacturing workcell. In *1993 IEEE International Conference on Robotics and Automation*, 1:778–782.

Downing, H.A. and Jeanne, R.L. 1988. Nest construction by the paper wasp, Polistes: A test of stigmergy theory. *Animal Behaviour*, 36:1729–1739.

Downing, H.A. and Jeanne, R.L. 1990. The regulation of complex building behaviour in the paper wasp, *polistes fuscatus* (insecta, hymenoptera, vespidae). *Animal Behaviour*, 39:105–124.

Drouin, M., Abou-Kandi, H., and Mariton, M. 1991. *Control of Complex Systems: Methods and Technology*, Plenum Press.

Engelberger, Joseph F. 1989. *Robotics in Service*. Kogan Page Ltd.

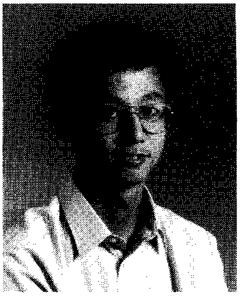
Franks, Nigel R. 1986. Teams in social insects: Group retrieval of prey by army ants. *Behavioral Ecology and Sociobiology*, 18:425–429.

Grassé, P. 1959. La reconstruction du nid et les coordinations interindividuelles chez *bellicositermes natalensis et cubitermes* sp. la théorie de la stigmergie: essai d'interpré. *Insectes Sociaux*, 6:41–81.

Kube, C. Ronald 1992. Collective robotic intelligence: A control theory for robot populations. Master's thesis, University of Alberta.

Kube, C. Ronald and Zhang, Hong 1992. Collective robotic intelligence. In *Second International Conference on Simulation of Adaptive Behavior*, pp. 460–468.
Kube, C. Ronald and Zhang, Hong 1993. Collective robotics: From social insects to robots. *Adaptive Behavior*, 2(2):189–219.
Kube, C. Ronald and Zhang, Hong 1994. Stagnation recovery behaviours for collective robotics. In *1994 IEEE/RSJ/GI International Conference on Intelligent Robots and Systems*, vol. 3, pp. 1883–1890.
Kube, C. Ronald and Zhang, Hong 1996. The use of perceptual cues in multi-robot box-pushing. Submitted to the *Journal Robotica* special issue on interacting robots.
Luo, Ren C. and Kay, Michael G. 1989. Multisensor integration and fusion in intelligent systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(5):901–931.
Mataric, Maja J. 1992. Minimizing complexity in controlling a mobile robot population. In *1992 IEEE International Conference on Robotics and Automation*, pp. 830–835.
Mataric, Maja J. 1994. Interaction and intelligent behavior. Ph.D. Thesis, Massachusetts Institute of Technology.
Mataric, Maja J., Nilsson, Martin, and Kristian T. Simsarian 1995. Cooperative multi-robot box-pushing. In *1995 IEEE International Conference on Intelligent Robots and Systems*, vol. 3, pp. 556–561.
McFarland, David and Bosser, Thomas 1993. *Intelligent Behavior in Animals and Robots*, MIT Press.
Metz, Hans A.J. 1977. State space models for animal behaviour. *Annals of Systems Research*, 6:65–109.
Moffett, M.W. 1988. Cooperative food transport by an asiatic ant. *National Geographic Research*, 4(3):386–394.
Moffett, M.W. 1992. Ant foraging. *National Geographic Research & Exploration*, 8(2):220–231.
Moser, J.C. 1970. Pheromones of social insects. In *Control of Insect Behavior by Natural Products*, D. Wood, R. Silverstein, and M. Nakajima (Eds.), Academic Press, pp. 161–178.
Noreils, Fabrice R. 1992a. An architecture for cooperative and autonomous mobile robots. In *1992 IEEE International Conference on Robotics and Automation*, pp. 2703–2710.
Noreils, Fabrice R. 1992b. Coordinated protocols: An approach to formalize coordination between mobile robots. In *1992 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 717–724.
Noreils, Fabrice R. 1992c. Multi-robot coordination for battlefield strategies. In *1992 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1777–1784.
Ota, Jun and Arai, Tamio 1993. Motion planning of multiple mobile robots using dynamic groups. In *1993 IEEE International Conference on Robotics and Automation*, vol. 2, pp. 28–33.
Ozaki, Koichi, Asama, Hajime, Ishida, Yoshiki, Matsumoto, Akihiro, Yokota, Kazutaka, Kaetsu, Hayato, and Endo, Isao 1993. Synchronized motion by multiple mobile robots using communication. In *Proceedings of the 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1164–1169.
Parker, Lynne E. 1993. Designing control laws for cooperative agent teams. In *1993 IEEE International Conference on Robotics and Automation*, vol. 3, pp. 582–587.
Parker, Lynne E. 1994. ALLIANCE: An architecture for fault tolerant, cooperative control of heterogeneous mobile robots. In *1994 IEEE/RSJ/GI International Conference on Intelligent Robots and Systems*, pp. 776–783.
Schricker, B. 1965. Die orientierung der honigbiene in der dammerung zugleich ein beitrag zur frage der ocellenfunktion bei bienen. *Zeitschrift fur Vergleichende Physiologie*, 49:420–458.
Seitz, A. 1940. Die paarbildung bei einigen cichliden i. z. *Tierpsychol*, 4:40–84.
Shibata, Takanori and Fukuda, Toshio 1993. Coordinative behavior by genetic algorithm and fuzzy in evolutionary multi-agent system. In *1993 IEEE International Conference on Robotics and Automation*, vol. 1, pp. 760–765.
Silby, Richard and McFarland, David 1974. A state-space approach to motivation. In *Motivational Control Systems Analysis*, D.J. McFarland (Ed.), Academic Press, pp. 213–250.
Stilwell, Daniel J. and Bay, John S. 1993. Toward the development of a material transport system using swarms of ant-like robots. In *1993 IEEE International Conference on Robotics and Automation*, vol. 1, pp. 766–771.
Sudd, J.H. 1960. The transport of prey by an ant, *pheidole crassindoa*. *Behaviour*, 16(3–4):295–308.
Sudd, J.H. 1965. The transport of prey by ants. *Behaviour*, 25(3-4):234–271.
Sudd, John 1963. How insects work in groups. *Discovery*, pp. 15–19.
Tremblay, Marc R. and Cutkosky, Mark R. 1995. Using sensor fusion and contextual information to perform event detection during a phase-based manipulation task. In *1995 IEEE International Conference on Intelligent Robots and Systems*, pp. 262–267.
Wang, P.K.C. 1991. Navigation strategies for multiple autonomous mobile robots moving in formation. *Journal of Robotic Systems*, 8(2):177–195.
Wehner, Rudiger 1987. Matched filters—Neural models of the external world. *Journal of Comparative Physiology A*, 161:511–531.
Wehner, R. and Srinivasan, M.V. 1984. The world as the insect sees it. In *Insect Communications*, T. Lewis (Ed.), Academic Press, pp. 29–47.
Wilson, E.O. 1971. *The Insect Societies*. The Belkap Press of Harvard University Press.
Wilson, Stewart W. 1990. The animat path to AI. In *First International Conference on Simulation of Adaptive Behavior*, MIT Press, pp. 15–21.
Wilson, E.O., Durlach, N.I., and Roth, L.M. 1958. Chemical releasers of necrophoric behavior in ants. *Psyche*, 65(4):108–114.
Zeil, J., Nalbach, G., and Nalbach, H.O. 1986. Eyes, eye stalks and the visual world of semi-terrestial crabs. *Journal of Comparative Physiology A*, 159:801–811.

C. Ronald Kube received his BSc and MSc degrees in 1990 and 1992 from the University of Alberta, where he is currently persuing a PhD in Computing Science. He is actively involved in promoting educational robotics, from Elementary school to University levels. His research interests include Collective Robotics, sensor integration, and mobile robot architectures.



Hong Zhang received his B.S. degree from Northeastern University in 1982 and his PhD from Purdue University in 1986 both in Electrical Engineering. He worked as a post-doctorate fellow in the Department of Computer and Information Science, University of Pennsylvania from 1986 to 1988, before he joined the Department of Computing Science, University of Alberta, Edmonton, Alberta, Canada where he is currently an Associate Professor. His research interests include dextrous robot manipulation, tactile sensing, and decentralized control of multiple robots.