

# Cooperative Material Handling by Human and Robotic Agents: Module Development and System Synthesis \*

J.A. Adams, R. Bajcsy, J. Kosecka, V. Kumar, R. Mandelbaum,  
M. Mintz, R. Paul, C. Wang, Y. Yamamoto, X. Yun.

GRASP Laboratory  
University of Pennsylvania, Philadelphia

## 1 Introduction

In this paper we present a collaborative effort to design and implement a cooperative material handling system by a small team of human and robotic agents in an unstructured indoor environment. Our approach makes fundamental use of the human agents' expertise for aspects of task planning, task monitoring, and error recovery. Our system is neither fully autonomous nor fully teleoperated. It is designed to make effective use of the human's abilities within the present state of the art of autonomous systems. Our robotic agents refer to systems which are each equipped with at least one sensing modality and which possess some capability for self-orientation and/or mobility. Our robotic agents are *not* required to be homogeneous with respect to either capabilities or function.

Our research stresses both paradigms and testbed experimentation. Theory issues include the requisite coordination principles and techniques which are fundamental to a cooperative multiagent system's basic functioning. We have constructed an experimental distributed multiagent-architecture testbed facility. The required modular components of this testbed are currently operational and have been tested individually. Our current research focuses on the agents' integration in a scenario for cooperative material handling.

### 1.1 Related Work

There are several groups addressing the issues of cooperation at many different levels. Some of the approaches study variations of the mobility problem and are motivated by the ethological studies of animal societies. In these scenarios the societies of agents are homogeneous and the tasks such as exploration, wandering, and foraging for food are usually achievable by a single agent [3, 14, 6, 24]. For heterogeneous mobile robots, the roles in the team need to change dynamically, based on the changes in the environment or the

individual robot's capabilities [22]. The cooperation issues are addressed more at the task level with no direct physical interactions between agents. Cooperation examples at the physical level can be found in cellular robots [27] or multi-arm manipulation [8]. As the task's complexity increases, requiring agents with different capabilities, the cooperation issues must be addressed at both high and low levels [20, 7, 10]. The task decomposition problem brings another crucial issue to the control of a society of agents, that is the tradeoff between local and global control. In other words, to what extent should the team members be aware of the team's global intentions as opposed to just acting upon local information sensed through the environment [21]? The amount of global control needed is task-dependent and generally the tasks which require some global resource optimization (e.g. time, space, energy) require a global view [19].

### 1.2 Assumptions

We constrain our investigation to a *static, indoor* environment with somewhat controlled illumination. We have two observation agents A and B, and two manipulatory agents C and D. Agent A has a suite of ultrasonic and infrared sensors, a patterned-light device and a stereo camera pair. Observation agent B is equipped with a stereo camera pair and a turntable with a camera for tracking. Agents C and D each have a manipulator (a PUMA 260 and a Zebra Zero). The human agent is supplied with a three dimensional graphical interface. A geometric workspace model, raw sensory data and processed data from the robotic agents are displayed, assisting with the human's supervisory role. Agent capabilities are described in more detail in section 2. The system is embedded in a Discrete Event Systems control theory framework where low level behaviors operate under DES supervisory control. The *task* is for agents C and D to carry a large object from one place to another through a narrow passage avoiding obstacles. Agents A and B scout the path and *advise* the manipulatory agents as to the free space layout. The human agent's task is to monitor, advise and intervene when necessary.

## 2 Components

In this section we discuss the manipulatory agents' hardware configuration and control architecture, the

\*This research is funded in part by: ARPA Grants N00014-92-J-1647, DAAH04-93-G-0419; ARO Grants DAAL03-89-C-0031PRI, DAAL03-92-G0153; Gateway Grant 9109794; NASA Grants NGT-50729, NGT-70359; NIH Grant 3R01LM0521703S1; NSF Grants BCS92-16691, BCS92-21796, CISE/CDA-88-22719, CDA91-21973, CDA92-11136, CDA92-22732, GER93-55018, IRI92-10030, IRI92-09880, IRI93-03980, IRI93-07126, MSS91-57156-A 02; University Research Foundation Grant 370892; and The Whitaker Foundation

observation agents’ capabilities, as well as the the human agents’ human-machine interface.

## 2.1 Mobile Manipulatory Agents



Figure 1: The two manipulatory agents.

Agents C and D each have a 6 degree-of-freedom (DOF) manipulator with a 6 DOF end-effector force sensor. Figure 1 depicts agent C (left) with a Zebra-ZERO manipulator and agent D with a Puma 260 manipulator. Control is decentralized.

The manipulatory agents’ major subtask is to cooperatively pick up the object prescribed by the human supervisor and transport it to the desired destination. Along the path stable grasping of the object is always required. In order to maneuver in a cluttered environment, the marching configuration may need to be changed from serial to parallel or vice versa. This elementary behavior is described in Section 3.1.3.

## 2.2 Mobile Observation Agents

The two observation agents are equipped with various sensor modalities. The subtasks of the observation agents comprise:

1. Localization of each agent in global coordinates,
2. Iterative verification that proposed trajectories are obstacle-free, and
3. Progress monitoring of the manipulatory agents (agents C and D), allowing for possible intervention by the human supervisor.

The observation agents employ five sensor modalities. In this subsection we describe each modalities’ physical model and basic operation, as well as the type of information we extract.

### 2.2.1 Ultrasound

Observation agent A, depicted in Figure 2, is equipped with a ring of sixteen standard POLAROID™ sensors. We employ a *real-time, polynomial* algorithm for ultrasound feature detection (described in [17]) which aggregates sonic data accumulated from *arbitrary* transducer locations and performs *sequential* clustering. The algorithm is *precise*, computationally *tractable* and *efficient, robust* in the face of measurement noise, and converges in a statistical sense to ground truth. The algorithm’s output comprises the parameters of features in space such as extended planar surfaces or corners. This algorithm may be employed

for exploration and map-building of an unknown environment, and for localization and hence navigation within partially known surroundings.

(a) (b)

Figure 2: (a) Observation agent A with four sensor modalities: ultrasound, stereo pair, patterned light and dead reckoning, and (b) Observation agent B with two sensor modalities: inverse perspective projection and dead reckoning.

### 2.2.2 Patterned-light

Observation agent A, depicted in Figure 2a, is equipped with a patterned-light device consisting of a light source projecting three light planes in front of the robot at an angle to the ground. A camera offset vertically from the light source uses elementary projective geometry to detect an object which intersects any of the light planes.

Gaps in the camera image’s stripes are interpreted as object segments. Over time, these segments may be grouped to form a shape corresponding to the “shadow” of the object. Registration and integration of several shadows extracted from multiple views yields the object’s “footprint”. The approach and the active exploration strategy used to plan the multiple views are described in [17]. This modality is employed to extract information regarding objects’ extents, orientations and shapes. Such information is useful for exploration and map-building, for the disambiguation of landmarks during navigation, and for localization.

### 2.2.3 Stereo

Observation agent A, depicted in Figure 2a, is equipped with a stereo camera pair. The images from the two cameras are compared, and the corresponding features’ azimuthal disparity is used to infer range.

We employ a string-comparison based stereo algorithm (described in [17]) which produces clustered, segmented data. Ideally, each segment corresponds to an object in the scene. Information regarding objects’ extents and azimuthal locations is obtained. This information is useful for exploration and map-building, as well as landmark disambiguation during navigation. Data regarding azimuthal position of landmarks may be used in conjunction with other modalities for agent localization.

### 2.2.4 Inverse Perspective Projection (IPP)

Agents B, depicted in Figure 2, is equipped with a stereo pair of cameras tilted with respect to the horizontal plane. Obstacles are detected through the difference between a pair of stereo images after applying the proper inverse perspective mapping proposed in [16]. Differences in perspective between left and right views are used to determine an obstacle’s presence and approximate location. The computed free-space map from the common field of view of both cameras is used for obstacle avoidance maneuvers [13].

The IPP modality monitors the free space ahead of the mobile base, providing the necessary parameters for obstacle avoidance in cluttered environments using artificial potential fields. The information can also be used for map-building.

### 2.3 Human Supervisory Control

In order to take advantage of our system’s autonomous aspect, and yet ensure successful completion of all tasks possible with teleoperation, we have developed our human-machine interface based upon the mediation hierarchy [2]. This hierarchy permits the human supervisor to interact at any level of our robotic system. Robotic systems are not robust in handling unmodeled events. Reactive behaviors may, or may not, be able to guide the robot back into a modeled state i.e., error recovery may not be achieved. Reasoning systems may fail. Once a system has failed, it is difficult to restart the task from the failed state. Rather, the rule base is revised, programs altered, and the task retried from the beginning.

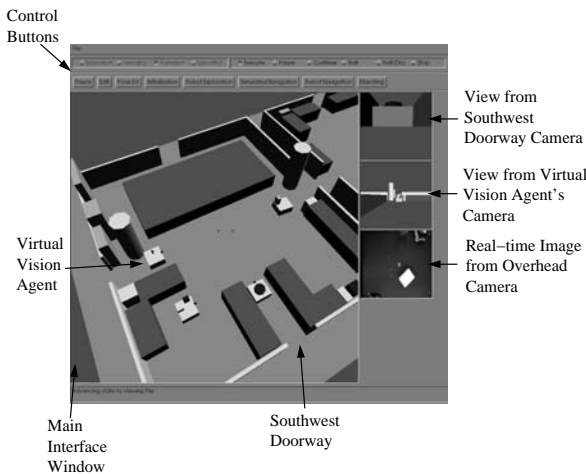


Figure 3: The MASC system interface.

Our interface, MASC — Multiple Agent Supervisory Control — permits the agents to work autonomously until the human supervisor detects a problem or is requested to assist the robots. We combine the advantages of autonomous systems with the human’s ability to control a system through a human-machine interface. MASC provides the human supervisor with interaction tools to all the robotic system’s processing levels. These interactions can correct corrupted data or process decisions which would typi-

cally cause an autonomous system to enter an incorrect state. We desire to create a more comprehensive semi-autonomous system based on this interaction which will successfully complete the assigned task’s execution.

The individual robotic agents and processes may be controlled by the human supervisor through MASC. The human’s primary role is to “supervise” the agents’ actions during task execution [25]. Through MASC, the human supervises the system, observes sensory data and images. Each agent is composed of multiple control and processing levels. For the successful semi-autonomous execution of feasible tasks, MASC must permit the human supervisor to interact with these levels. We have organized the supervisor’s interactions with the many system levels into the mediation hierarchy.

We have provided display control buttons (see Figure 3) to allow the human supervisor to specify system information. The human supervisor may request any agent’s sensory data while in any system state. The agents transmit dead-reckoning information, sensory measurements, raw image data, and processed data. This information is employed by MASC to create various system displays. Image data may be displayed in windows to the right of the main interface window or may be overlaid onto the virtual environment model (see Figure 3). Processed data, such as the free space map originating from the visually guided obstacle avoidance process, may be displayed in a window or overlaid onto the virtual model.

MASC combines autonomous and telerobotic control. The human monitors the task execution and then assists the agents when they enter unstable states from which task execution would fail. While an agent proceeds autonomously it may petition the human’s assistance. The human supervisor must acknowledge the request and then furnish the requisite information.

## 3 Behaviors and Task Description Language

The basic control strategies associated with the available system components comprise a set of *elementary processes* associated with each sensor’s basic perceptual capabilities and each actuator’s basic motion modes. Each motion mode corresponds to a particular control law describing the manner in which the commands are generated. Similarly, sensors have associated data acquisition and extraction procedures. Each elementary process is modeled in terms of finite state machine (FSM). Each state corresponds to the application of a particular control or perceptual strategy. Each transition models an external event (depending on the task) or the successful completion or failure of the strategy. Behaviors and tasks are then expressed as elementary-process networks formed using a set of composition operators. Processes  $R$  and  $S$  can be composed either **sequentially**  $R ; S$  (when process  $R$  terminates  $S$  is initiated), in **conditional** fashion  $R_{\langle v \rangle} : S(v)$  ( $R$  must terminate successfully, computing value  $v$ , before  $S$  can be initialized), **concurrently** ( $R \parallel S$ ) (the two processes are initiated in parallel and, in the case where they share an event, the

communication link between them is established), or in **disabling** fashion  $R \# S$  (similar to concurrent, but if one of the processes fails then the other processes also fails). Other composition operators for modeling iterative processes are **synchronous recurrent** and **asynchronous recurrent**. The intuition behind these operators is identical to Lyons [15]; a more detailed semantic description in FSM terms can be found in [13]. Given a sentence in the task description language, a primary objective of this approach is the synthesis of a FSM controller (DES supervisor) which would guarantee the task execution. By imposing additional liveness and safety constraints and formulating the problem in terms of DES Supervisory Control theory [23], we can synthesize the least restrictive supervisor which satisfies the constraints while accomplishing the task<sup>1</sup>. The DES supervisor then monitors the task execution, invoking correct strategies and monitoring system responses. This DES supervisor operates in parallel with the human supervisor which may override the DES supervisor’s decisions. A more in-depth description of this approach can be found in [13].

For more complicated tasks occurring in multi-agent systems, task decomposition subtask assignment phases are necessary. This stage is currently completed by the human supervisor, although we are currently incorporating a higher-level symbolic planner to assist the human. We demonstrate the idea of representing tasks as process networks in a few examples.

### 3.1 Elementary Processes

#### 3.1.1 Local Maneuvering

The central question for the control of a mobile base is how to move it from one location to another in a structured or unstructured environment. This problem involves issues of path planning, motion planning and localization given available sensory information and/or a priori knowledge. Within this work we explore both path planning and control issues, while assuming that the global goal/objective is determined beforehand. We have implemented two strategies for local maneuvering. One utilizes artificial potential fields to steer the mobile base in a closed loop fashion, while the other constructs an  $R$ -geodesic path which the base follows in an open loop manner. Both strategies take into account the mobile bases’ nonholonomic constraints.

**Potential Fields** This method provides us with an incremental on-line-generated holonomic path, which is modified using simple projection strategy for non-holonomic robots [9]. We adopt an artificial potential field method originally introduced by [12], where the environment with a single goal is represented by an attractive potential field with the minimum at the goal location  $U_a$ , and obstacles are represented by a repulsive hyperbolic potential function  $U_r$ . The desired velocity  $\dot{X}_d$  and desired turning rate  $\dot{\theta}_d$  are computed in the following manner:

$$\begin{aligned} \dot{X}_d &= -\nabla_X (U_a(X) + U_r(X)) \\ \dot{\theta}_d &= \arctan(\dot{x}_d, \dot{y}_d) - \theta \end{aligned} \quad (1)$$

<sup>1</sup>The supervisor in this section is a FSM operating in a closed loop within the system. This is to be distinguished from the human supervisor mentioned in previous sections.

The resulting linear velocity setting is then computed by the projection of the desired velocity  $\dot{X}_d$  to the current heading. By adopting this control strategy for point to point motion, we associate the following motion modes with the mobile base.

The **GoTo** mode implements directly the above derived control law (1). The **GoToMarch** control law generates commands for two mobile bases while marching in parallel formation (next to each other), while keeping constant the distance from the midpoint. It is again derived from the control law (1), while the combined size of the bases is adjusted (i.e., boundaries around obstacles are adjusted), proportionally to the distance between the two platforms. **GoToHeading** mode implements a pure rotation to the desired heading  $\theta_d$ . The mobile base “motion modes” use proportional feedback laws, servoing on goals while avoiding obstacles. The goal and obstacles can be supplied either by perceptual processes, by the DES supervisor process, or by the human supervisor.

**$R$ -geodesic Paths** It is shown in [11] that a  $R$ -geodesic path is the shortest maneuver that brings the nonholonomic vehicle from one configuration to another on a two dimensional plane. The basic elements of such a path consist of an arc, followed by a straight line and then an arc. The arcs are of radius  $R$  which corresponds to the minimum turning radius of the vehicle. We have developed an algorithm to determine the dimensions of these three elements for given starting and ending configurations (see [28]) and use it as a planner for local vehicle maneuvering.

#### 3.1.2 Simple active sensing strategies

Each subtask described in the introduction to section 2.2 necessitates its own *active sensing strategy*. For **localization of agent A**, we employ the localization method described in [18]. For **trajectory verification**, agent A attempts a traversal of the proposed path using the IPP modality for obstacle detection. Once an obstacle-free path has been identified, **progress monitoring** of the sensor-impaired agents is achieved by agent B following manipulatory agents C and D; the human supervisor examines the image stream emanating from one of the cameras.

#### 3.1.3 Coordination of the mobile manipulators

When multiple mobile manipulators are commanded to transport a large object along a desired trajectory, there are two different approaches to achieve the coordinated task. One approach treats both the mobile manipulators and the grasped object as a closed kinematic chain. A controller for the whole system is designed. In this approach all measurements and controls are assumed to take place in the same bandwidth and must address the different dynamic characteristics of a manipulator and a mobile platform — the manipulator is used to achieve a fine, fast positioning while the mobile platform’s response is fairly slow and is only suitable for gross motions. In addition, as the system becomes complex, *e.g.*, the more joints or more mobile manipulators involved, the computational expense for this scheme increases dramatically.

In order for the system to be flexible and computationally manageable, a more decentralized approach is desirable. A decentralized approach implies that

the mobile manipulators should be able to execute the tasks based on a limited amount of information exchange between the agents. This may be achieved in a variety of ways. For instance, a reliable force sensor at the end-point will be helpful to infer the partner agent’s “intention”. Also the desired trajectory may be carefully designed so that it makes the coordination between the agents easier from various perspectives, *e.g.*, the nonholonomic constraints, the manipulator’s workspace, or the controller complexity.

We performed a simulation to demonstrate one of the subtask executions mentioned in section 2.1, *i.e.*, changing the two agents’ configuration from a parallel to a serial formation<sup>2</sup> where the two mobile agents are more loosely coupled. A difficulty with this case is that, due to the nonholonomic constraints, the controller must be switched from one platform to the other in order to align the two platforms in parallel; this causes a small drift of the end-point towards the end of the trajectory. However this small error at the end-point can be compensated by a stiffness type of control on one of the mobile agents [1].

## 4 Future Work

The previous sections have described the various components of our cooperative multiple agent system. Currently we are in the process of integrating these components into a complete system. Upon completion of system integration we will test the system’s performance. The overall system goal is to transport a large object from one location to another while avoiding obstacles and passing through a doorway. This goal will be broken up into the following subtasks manually:

- Observation agent A checks the prescribed path for obstacles, and ensures the pathway is wide enough to accommodate the two manipulatory agents carrying the object.
- Observation agent B follows the two manipulatory agents, allowing the human supervisor to monitor their progress via camera images.

Within this context, the following parameters allow for many variations on the basic scenario:

**Initial Localization:** In the simplest case, all agents’ initial locations and orientations within the global coordinate frame (CF) are known. *Partial sensor-based initial localization* is necessary if the agents’ *relative* positions are known in a local CF, but the local CF needs to be aligned with the global CF. If agents’ locations and orientations are unknown, the observation agents must localize all agents before path-planning and execution begins.

**Continuing Localization:** Dead-reckoning systems suffer from error accumulation; for long-term task-execution, they should be supplemented with some other form of localization. *Beacon-based* systems involve modification of the environment, whereas *landmark-based* systems require the detection of *naturally occurring* features [5, 26].

<sup>2</sup>The details of the coordination scheme employed for agent C can be found in [29]

**Path generation:** The human agent may specify the path waypoints explicitly, or teleoperate an observation agent along a desired path. Alternatively, a path-planner may be employed [4, 28].

**Knowledge of the environment:** There are no unforeseen obstacles in a *completely-known* environment. In a *partially-known* environment, the landmark locations in the global CF are known. A *completely-unknown* environment requires exploration and map-building by the observation agents before any path planning or traversal may begin.

**Obstacles:** In an *obstacle-free* environment, no deviation from the precomputed plan is necessary. If obstacles occur relatively infrequently, deviation from the path may be necessary, though no reconfiguration of the manipulatory agents. In environments with a *high* density of obstacles, the manipulatory agents may have to change to leader-follower configuration (without dropping the object).

**Need for reconfiguration:** Even in an obstacle-free environment, the doorway through which the manipulatory agents must pass may be too narrow to permit a side-by-side configuration, and a reconfiguration during execution is necessary. In contrast to the case involving *unforeseen* obstacles, such a reconfiguration may be planned and scheduled.

The number of combinations afforded by the above scenario parameters is prohibitively large. Initially we will test the following three illustrative and increasingly complex situations:

	Initial Localization Necessary	Continuing Localization Method	Path Generation
1	none	dead reckoning	human
2	partial	dead reckoning	human
3	partial	dead reckoning	human

	Knowledge of Environment	Obstacle Density	Need to Reconfigure?
1	complete	zero	yes: planned
2	partial	low	yes: planned
3	partial	high	yes: unforeseen

## 5 Conclusions

Based on the experience gained in designing, testing, and integrating the modules in the experimental system described above, we cite the following conclusions:

- The human agent is a necessary component in the successful operation of a system for multiagent cooperative material handling in an unstructured indoor environment. Our approach makes fundamental use of human agents’ expertise for aspects of task planning, task monitoring, and error recovery.
- The partitioning of the robotic agents into two classes — mobile observers, and mobile manipulators — provides very useful degrees of freedom in the experimental design. This flexibility allows us

to build systems which have more extensive abilities to observe manipulatory agents working either in close quarters or near obstacles.

- Our application of potential functions has shown the value of this approach in controlling the motion of multiple vehicles in obstacle-laden environments. The motions exhibit good stability and smoothness characteristics.
- One of the salient aspects of our DES supervisory control system is its ability to schedule and monitor subtasks within a given task requirement.
- At present there is no single sensing modality which is versatile enough to provide sufficient data about the environment for the execution of all tasks within our scenario. Sensory integration between multiple modalities is therefore essential.

## References

- [1] J. Adams, R. Bajcsy, J. Kosecka, V. Kumar, R. Mandelbaum, M. Mintz, R. Paul, C.-C. Wang, Y. Yamamoto, and X. Yun. Cooperative Material Handling by Human and Robotic Agents: Module Development and System Synthesis. *Technical Report, Dept. of Computer and Information Science, University of Pennsylvania, MS-CIS-95-01, GRASP LAB 385*.
- [2] Julie A. Adams. *Human Management of a Hierarchical Control System for Multiple Mobile Robots*. PhD thesis, University of Pennsylvania, In Progress, 1995.
- [3] R. Arkin, T. Balch, and E. Nitz. Communication of behavioral state in multi-agent retrieval tasks. In *IEEE Intl. Conference on Robotics and Automation*, pages 588–593, 1993.
- [4] Craig Becker. *Internet Path Planner Server Protocol*. Stanford University, January 1994.
- [5] M. Betke and K. Gurvits. Mobile robot localization using landmarks. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 2, pages 135–142, May 1994.
- [6] R. A. Brooks, P. Maes, M. Mataric, and G. Moore. Lunar base construction robots. In *IROS-90, IEEE International Workshop on Intelligent Robots and Systems*, 1990.
- [7] P. Caloud, W. Choi, and J. Latombe. Indoor automation with many mobile robots. In *IROS-90*, pages 67–72, 1990.
- [8] P. Chiacchio, S. Chiaverini, L. Sciavicco, and B. Siciliano. Task space dynamic analysis of multiarm system configurations. *The International Journal of Robotics Research*, 10(6):708–715, December 1991.
- [9] A. De Luca and G. Oriolo. Local incremental planning for nonholonomic mobile robots. In *Proceedings of 1994 International Conference on Robotics and Automation*, pages 104–110, San Diego, CA, May 1994.
- [10] K. S. Decker and V. R. Lesser. The analysis of quantitative coordination relationships. *Coins* 91-83, Department of Computer Science, University of Massachusetts, Amherst, 1992.
- [11] L. E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, (79):497–516, 1957.
- [12] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research*, 5(1):90–98, 1986.
- [13] J. Košecká. Supervisory control of autonomous mobile agents. Technical report, GRASP Laboratory, Departments of Computer Science, University of Pennsylvania, 1995.
- [14] G. Lucarini, M. Varoli, R. Cerruti, and G. Sandini. Cellular robotics: Simulation and hw implementation. In *IEEE Robotics and Automation, Intl. Conference*, 1993.
- [15] D. M. Lyons. A formal model of computation for sensory-based robotics. *IEEE Transactions of Robotics and Automation*, 5(3):280 – 293, 1989.
- [16] H.A. Mallot, H.H. Bülthoff, J.J. Little, and S. Bohrer. Inverse perspective mapping simplifies optical flow computation and obstacle detection. *Biological Cybernetics*, 64:177–185, 1991.
- [17] R. Mandelbaum. *Sensor Processing for Mobile Robot Localization, Exploration and Navigation*. PhD thesis, University of Pennsylvania. In preparation.
- [18] R. Mandelbaum and M. Mintz. Feature-based localization using fixed ultrasonic transducers. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, August 1995.
- [19] M. Mataric. Minimizing complexity in controlling a mobile robot population. In *IEEE Intl. Conference on Robotics and Automation, Nice*, pages 830 – 835, 1992.
- [20] F. R. Noreils. An architecture for cooperative and autonomous mobile robots. *Proceedings of the IEEE International Conference on Robotics and Automation*, 3:2703 – 2710, May 1992.
- [21] L. E. Parker. Designing control laws for cooperative agents. In *IEEE Intl. Conference on Robotics and Automation*, pages 582–587, 1993.
- [22] L. E. Parker. Learning in cooperative robot teams. In *Workshop on Dynamically Interacting Robots, IJCAI*, 1993.
- [23] P. J. Ramadge and W. M. Wonham. The control of discrete event systems. *Proceedings of the IEEE*, 77(1):81–97, January 1989.
- [24] C. W. Reynolds. Flocks, herds and schools: A distributed behavioral model. In *SIGGRAPH*, pages 25 – 33, 1987.
- [25] Thomas Sheridan. *Telerobotics, Automation, and Human Supervisory Control*. MIT Press, Cambridge, Mass., 1992.
- [26] W. B. Thompson, T. C. Henderson, T. L. Colvin, L. B. Dick, and C. M. Valiquette. Vision-based localization. In *Proceedings of the 1993 Image Understanding Workshop*, pages 491–498, April 1993.
- [27] T. Ueyama and T. Fukuda. Self-organization of cellular robots using random walk. In *IEEE Intl. Conference on Robotics and Automation*, pages 595–600, 1993.
- [28] Chau-Chang Wang. Local path planner. University of Pennsylvania. Personal Communication, Sept. 1993.
- [29] Y. Yamamoto. *Control and Coordination of Locomotion and Manipulation of a Wheeled Mobile Manipulator*. PhD thesis, University of Pennsylvania, Grasp Laboratory, 1994.