Human-Robot Interactions in Active Sensor Networks

Alexei Makarenko [*], Tobias Kaupp, Ben Grocholsky, and Hugh Durrant-Whyte
Australian Centre for Field Robotics
The University of Sydney, Australia
http://www.acfr.usyd.edu.au/

Abstract

This paper considers the modes of interaction between one or several human operators and an active sensor network – a fully decentralized network of sensors some or all of which have actuators and are in that sense active. The primary goal of this study is to investigate the conditions under which the human involvement will not jeopardize scalability of the overall system. Two aspects of human-robot interaction are considered: the ways in which the global view of the system may be conveyed to the operators, and how the operators may influence the behavior of the system during the course of its operation. The results of analysis favor peer-to-peer information-based interactions between the operators and the network whereby the humans act as extended sensors and communication nodes of the network itself. Experiments on an indoor active sensor network are described.

1 Introduction

This paper considers several aspects of humans interactions with large decentralized robotic systems engaged in information gathering activity. The term *Active Sensor Network* (ASN) used in this work refers to a fully decentralized network of sensors some or all of which have actuators and are, in that sense, active.

Sensor networks which are based on decentralized architecture are scalable, robust, and modular [1]. For an ASN, both the data fusion and the control parts must be decentralized and scalable. For an ASN with a human operator in the loop, the human-robot interface must also be designed with scalability in mind.

The primary goal of this study is to investigate the conditions under which the human involvement will not jeopardize scalability of the overall system.

The nature of human-robot interaction in any multi-robot system is two-fold: a) how to present the user with a global view of the system which is likely to be decentralized; and b) how to influence the actions of many information gathering robots.

The rest of the paper is organized as follows. Next section reviews related work. Section 3 investigates the role human operators play in the data fusion and control aspects of an ASN. Section 4 describes experiments conducted on a practical indoor ASN. The last section presents conclusions and outlines future work directions.

[*]Corresponding author: alexei@acfr.usyd.edu.au

2 Related Work

This work lies at the intersection of the three related fields of sensor networks, multi-robot systems, and human-robot interface as shown graphically in Figure 1. The review of related work follows the regions in that figure.

The area of passive (uncontrolled) sensor networks grew out of the field of distributed sensing, estimation, and data fusion. Recent work focuses on large dynamic sensor networks in which *attribute based naming* [2] must be used instead of traditional address naming. Simulations of information-theoretic sensor selection algorithms in networks of up to a 100 nodes are reported in [17]. This work builds on a decentralized data fusion (DDF) architecture [16], demonstrated in a network of up to 8 nodes.

Active sensor networks (I): A hierarchical architecture applied to real time tracking is described in [7]. The system of up to 32 nodes performed data fusion and made control decision, but it is not fully decentralized and human-robot issues are not addressed. An architecture for DDF and control with an emphasis on scalability is described in [9]. A team of three indoor robots cooperatively build an map of a building in [15]. Both data fusion and control part of the algorithm are centralized. Decentralized control techniques are combined with DDF in [5]. In all of these works, the issues of human-robot interface are not addressed.

This research area also includes a large body of work on reactive architectures, most of which are fully decentralized and some were applied to sensing applications [10, 11]. They fall outside of the scope of this work because they do not perform network-level data fusion and typically do not interact with humans.

Human-robot interaction in multi-robot systems (II): The research in this field to date has been limited to communication between one or several humans and a small number of robots.

The field of *adjustable autonomy* [8] or *collaborative control* [3] is an active research area which aims to span the gap between teleoperation and full autonomy. Current solutions call for bidirectional communication in the form of a human-robot dialog. With respect to scalability this approach suffers from increased communication traffic and the human operator is limited in the ability to provide assistance simultaneously to several robots.

Larger number of robots require more assistance. One solution is to increase the number of operators, another is to increase the level of autonomy exercised by the robots. The goal of this work is to examine architectures which allow growth to arbitrary size which precludes, at current stage, the use of techniques related to adjustable autonomy. Full autonomous operation is therefore assumed.

Human-robot interactions in sensor networks (III): Rencken et al. [13] describes a DDF system of four cam-

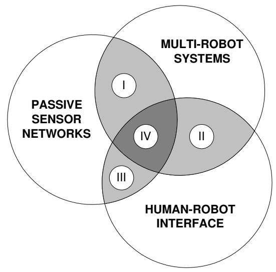


Figure 1: Convergence of the traditional field of multi-robot systems with the fields of passive sensor networks and human-robot interactions. At the intersection lie (I) active sensor networks (ASN), (II) multi-robot human interface, (III) human interface to sensor networks, and (IV) human interface to ASN

eras tracking moving targets. A global view of the network state is provided for the human operator, furthermore, an adaptive task allocation algorithm queues targets to be tracked and classified by the network or the operator. Decentralized control is not part of this work.

3 Active Sensor Networks

The active sensor networks considered in this paper are based on a decentralized architecture. A decentralized system is characterized by three constraints [1]:

1. There is no single central information fusion or coordination center; no node should be central to the successful operation of the network.

2. There is no common communication facility; nodes cannot broadcast results and communication must be kept on a strictly node-to-node basis. Although technically, a common communication facility violates constraints of a decentralized system, a broadcast medium is often a good model of real communication networks.

3. Sensor nodes do not have any global knowledge of sensor network topology; nodes should only know about connections in their own neighborhood.

The constraints imposed provide a number of important characteristics for decentralized systems:

- Eliminating the central coordination center and any common communication facility ensures that the system is *scalable* as there are no limits imposed by centralized computational bottlenecks or lack of communication bandwidth. Scalability here means the network can grow arbitrarily large in the number of connected nodes.

- Ensuring that no node is central and that no global knowledge of the network topology is required means that the system can be made *survivable* to the on-line loss (or addition) of sensing nodes and to dynamic changes in the network structure.

- As all processes must take place locally at each sensor site and no global knowledge of the network is required *a priori*, nodes can be constructed and programmed in a *modular* fashion.

3.1 Multi-Agent Framework

To make discussion of various aspects of communication within an ASN more concrete, a formal team framework will be used [6, 12]. The problem is stated as two tuples.

$$\langle S, \mathbf{A}_\alpha, \mathbf{\Sigma}_\alpha, P, \mathbf{\Omega}_\alpha, \mathbf{O}_\alpha, \mathbf{B}_\alpha, R, T \rangle, \langle \pi_\Sigma, \pi_A \rangle \tag{1}$$

where for each agent i in team α

S	world states (terrain, features, agents, etc.)
$\mathbf{A}_i$	agent's domain-level actions
$\mathbf{\Sigma}_i$	agent's communication actions
P_i	world model, i.e. transitional probability $P : S \times \mathbf{A}_\alpha \times S \to [0, 1]$ $P(s, \mathbf{a}, s') = Pr(S^{t+1} = s' \mid S^t = s, \mathbf{A}^t = \mathbf{a})$
$\mathbf{\Omega}_i$	agent's observations
$\mathbf{O}_i$	agent's observation function $O_i(s, \mathbf{a}, \omega) = Pr(\Omega_i^t = \omega \mid S^t = s, \mathbf{A}^{t-1} = \mathbf{a})$
$\mathbf{B}_i$	agent's belief states, updated through observations and communications
R	a common reward function $R : S \times \mathbf{A}_\alpha \to \Re$
$T > 0$	time horizon

In addition two control policies must be specified

$\pi_{i\Sigma} : B_i \to \Sigma_i$ communication policy

$\pi_{iA} : B_i \to A_i$ action policy

The structured representation of the team problem is general enough to include the ASN problem and allows to examine systematically the possible ways to influence the decisions of individual agents – the essence of human-robot interaction.

3.2 Human Role in Decentralized Data Fusion

Data fusion in general refers to the process of combining observations from multiple and dissimilar sensors into a global consistent view of the world. Decentralized data fusion must, in addition, satisfy the three constraints listed in the Section 3.

One of the challenges in designing an ASN is in how to present the human operator with a global picture of the world in a decentralized system in such a way that the GUI does not become the communication or processing bottleneck.

In broad terms, a network can be queried for two types of information: information about the environment and information about the components of the network itself. It is important to note that the former does not increase in size with the growth of the network and the latter does. The two types of information are considered separately next.

Environment Information

The primary goal of any ASN is to collect information about the environment, fuse it into a consistent belief and take actions based on this belief. The GUI's primary goal is to present this belief to the human operator.

In the notation of team framework, DDF algorithm is primarily a communication policy $\pi_\Sigma^{\mathrm{DDF}}$ which is to send all new information about the environment along the channels [16]. This policy insures that a consistent belief $\mathbf{B}_\alpha$ about the state of the world is maintained across all members of the team.

One of the properties of the DDF architecture is that each node in the network contains the same belief state (ignoring the information propagation delays). Because of this, a GUI in a DDF network needs simply to query a *single* node in order to obtain the knowledge of the entire network.

Since the communication and computational load in the DDF algorithm is independent of the number of network components, the same can be said about the GUI interface which obtains its information from a DDF node.

Network Component Information

The amount of information about the nodes themselves, such as their physical location in space, is proportional to the number of nodes in the network. Therefore a GUI which collects and relies on the state information from the network components cannot scale to arbitrary size.

The information about network components is always useful and often invaluable for debugging and status monitoring purposes. An example of collecting and displaying such information is described in Section 4.2. However, it is important to realize that relying on this information makes the GUI a computational or communication bottleneck.

Following the elements of Equation 1, other examples of component-specific information include current or future actions (plans), communication messages, component models, observations, and decision policies.

3.3 Human Role in Decentralized Control

Some or all network components in an ASN may have actuators. The control objective is typically to maximize the total information gain realized by the network, subject to certain state or control constraints. Modelling of platforms, sensors and environment as a set of continuous states, together with the use of information as payoff, allows the information acquisition problem to be formulated as a standard optimal control problem [4].

Several classification systems for human-robot interactions have been suggested. For example, Scholtz [14] identifies *mechanic*, *supervisory*, and *peer-to-peer* levels of human-robot interaction.

Alternatively, the potential control methods can be identified by examining Equation 1. An operator can tell a robot or a group of robots

- what to do (send an action a),
- what operating mode to switch to (select action policy π_A),
- what outcomes to favor (modify reward function R), or
- what additional information about the world to use (an observation ω or a belief b)

Technically, all of these options can implemented in scalable fashion provided that attribute based naming [2] is used and proper communication infrastructure is available. A tree network created and maintained by the DDF algorithm is suitable for transmitting such command messages as well.

The four options can be grouped into two broad categories: state-based and information-based which are examined in more details below.

State Based Control

Under *direct* type of control an action a or a sequence of actions $\{a\}$ is sent through the network addressed to the components which possess the attributes specified. These commands are to be executed directly and assume no autonomy. This approach is a direct extension of the teleoperation or *mechanic* mode of operation. It may be invaluable in emergency situations, but cannot be considered a primary control tool in large decentralized sensor networks.

Under the *mode switch* type of control, the components of the network operate autonomously most of the time, but are periodically commanded to switch from one pre-programmed action policy to another. This control method falls under the *supervisory* mode.

Under the *payoff change* type of control, a change in the utility function R is sent to all or a subset of the network components addressed by appropriate attributes. This control mode is also at the *supervisory* level.

Information Based Control

Under the *information based* control mode, the information about the environment which has become available from outside of the network is entered into the network by a human operator. It is then fused into the system and will affect actions of one or more of the network components. This is a *peer-to-peer* mode in which the operators and the network components act on the same level of authority.

The information can be entered into the network in one of two forms: as a raw observation ω which will be used to update the affected states or, more directly, as a posterior of a particular state or states. In the first case, the operator acts as a sensor submitting information to a node. In the second, he or she plays the role of another node sending state update through the DDF link.

Influencing the action of the network components by providing it with additional information is a natural extension of the underlying ASN architecture. The information entered by the operator only needs to be transmitted to a single node and the network itself propagates it through the rest of the system, potentially leading to actions by several components.

4 Experiments

This section describes experimental demonstration of a an indoor ASN consisting of several sensor platforms and two GUI nodes. Implementation details are given first, followed by the description of the GUI design in Section 4.2 and decentralized control implementation in Section 4.3.

4.1 Network Implementation

In the design of the software implementation of the current architecture, four fundamental types of decentralized objects were identified: a platform, a node, a sensor, and a controller. These object definitions are by no means unique but they proved to be beneficial to the object oriented qualities of the software and these terms are used throughout the rest of the paper.

A *platform* is a physical object placed in the physical world. It is the *only* physical object and, therefore, in order to have a location in the world all other objects must be attached to a particular platform. A *sensor* is an abstract object which gathers information from the

(a) (b) (c) (d)

Figure 2: Elements of a practical decentralized ASN : (a) mobile platform (Pioneer II) with a mounted sensor (SICK laser range finder), (b) a stationary sensor (SICK laser range finder), (c) a mobile GUI node on an iPAQ PDA, and (d) a stationary GUI node on a Linux workstation

environment. A *node* is an abstract object which processes information and communicates with other nodes in order to reach a consistent belief about the world. A *controller* is an abstract object which maximizes a certain reward function based on the current state of the world and the known model of the world.

A sensor can be connected to a platform which means that it is physically attached to it with a specific offset. A node can be connected to a platform which means that its computations are executed on the processor located on that platform. A sensor can be connected to a node which means that it sends its observations (in information form) to that node. When a node is connected to another node it means that they establish and maintain a *DDF link* leading to synchronization of their belief about the state of the world [16]. Finally, a controller can connect to a platform, a node, and a sensor. This means that, based on the current pose of the platform, the sensor and actuator models, and the current state of the world supplied by the node, the controller will issue commands to the actuator which will maximize a certain reward function [5].

Any number of nodes, sensors, and actuators may be attached to a single platform but from the point of view of a node, a sensor, or a controller platform assignment is unique. Likewise, an arbitrary number of sensors may contribute their observations to a particular node, but any particular sensor sends its observations only to a single node. The current implementation is limited to a maximum of four DDF links per node. This number is sufficient for creating an arbitrary tree network and prevents formations of bottlenecks at highly-connected nodes.

Figure 2 shows several physical components which comprise the indoor ASN. Up to three mobile Pioneer II platforms and one or two stationary sensor nodes were used in the experiments. The sensors used so far have been exclusively SICK LMS291 laser range finders with a scan angle of 180 degrees and a range of 8 m. A vision sensor unit is under development.

The environment is an unmodified office space, approximately 20 m on each side. The long term objective of the project is to cooperatively identify and track prominent features in the environment, both stationary and moving. At the current stage, a number of light reflectors are placed around the space. These can be easily identified as point features by the laser range finders.

The objective of the network is to find and localize the point features. An *information surfing* controller is implemented on the Pioneer robots [5]. It is a zero look-ahead control law which moves the platform in the direction of the steepest descent in information space.

All software components run as separate applications and communicate with each other using a common communication library. Communication is supported over different mechanisms: on the same processor and networked, both wireless and connected by a bus.

Two types of network access points have been implemented: a stationary GUI on a desktop computer and a mobile one running on a PDA.

A dynamic configuration protocol is used to assemble the network at start-up time. Random identification numbers are assigned after testing against duplicates. Tree network topology is used due to its simplicity and availability of theoretical results. Object configuration is performed either at start-up or at any time during the operation through the GUI.

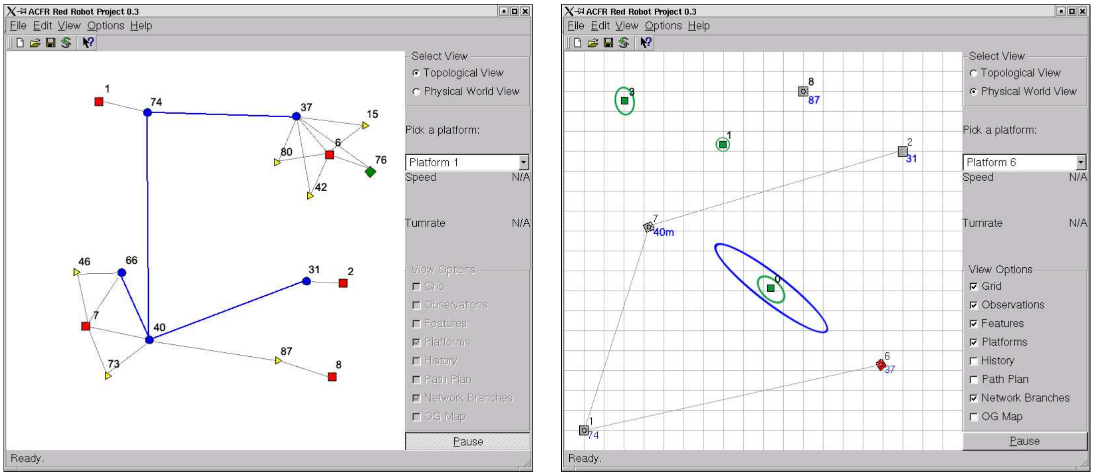
4.2 Graphical User Interface in DDF

To provide a human controller with a view into the functioning of an active sensor network is the first purpose of a GUI. Figure 3 displays two different views of the state of the network: the network topology is shown on the left and the physical layout of the network is shown on the right.

The topological view shows four DDF nodes with randomly assigned IDs connected into a tree network. Nodes N31 and N74 are attached to platforms without any sensors or controllers. They both represent GUI nodes linked to the platforms P1 and P2 representing a stationary operator workstation and a mobile one implemented on a PDA. Because the GUI nodes are ordinary DDF nodes, they are exposed to the same information flow as the rest of the nodes and can be attached at any point in the network.

In one respect a GUI node is different from the rest of the network. It may be capable of accumulating nonlocal information about the network itself, such as the global topology shown in Figure 3(a). This information is used purely for visualization and debugging purposes and, therefore, does not undermine the decentralized nature of the DDF approach. If the communication bandwidth limit is reached due to the size of the network, the network components can be configured to stop sending their state updates to the GUI. The information about the state of the environment is not affected.

Platform P6 holds three sensors (S15, S42, S80), one



(a) (b)

Figure 3: Two views of the same active sensor network: (a) topological and (b) physical. The topological view shows several ASN objects: platforms (□), nodes (○), sensors (△), and controllers (◇) with corresponding connections. The physical view shows mobile platforms connected by DDF links. Point features known to the network are also shown with corresponding covariance ellipses.

controller (C76), and one node (N37) – a typical configuration for a mobile robot shown in Figure 2(a). Two sensors (S46, S73) and two nodes (N40, N66) are attached to platform P7 without a controller because this platform is stationary and does not have any actuators. A stationary laser node is shown in Figure 2(b). Platform P8 has a sensor (S87), but has neither a node nor a controller, possibly due to insufficient computing power to run a DDF node. Instead of processing its observations locally, the information from the sensor is sent to node N40 running on platform P7. Platform P8 is a stationary sensor unit only capable of sending information.

Compare the topological view to the second view showing a snapshot of the network configuration in physical space. The human operators (P1, P2) are drawn as squares. All platforms are connected via their built-in DDF nodes except P8, which does not hold a node and thus is not involved in the DDF network.

Additionally, three known features and their position uncertainty are shown in this view with relatively small ellipses. The large ellipse represents the uncertainty of the latest observations made by a sensor on platform P6. The difference in the size of the ellipses is due to the information accumulation inherent in the data fusion process.

Combined, the two views of the network satisfy the objectives of providing the user with a global system view while maintaining the scalability property.

4.3 Human DDF Node

The second purpose of the GUI is to provide an opportunity for a human operator to influence the decision making inside an ASN. The experimental part of the work has concentrated on the information-based control methods as described in Section 3.3.

Figure 4 demonstrates manual input of feature information though the GUI. Figure 4(a) shows a scenario of two mobile robots (P6, P7) connected to a stationary GUI workstation P1 via their DDF nodes. Two features are currently known to the sensor network and displayed with their uncertainty ellipses. The human operator identifies the likely location of a feature currently unknown to the network and out of the current sensor range.

The uncertainty circle drawn by the operator is centered at the most probable location of the new feature, and the radius of the circle corresponds to three standard deviations of its position uncertainty. The properties of this circle are sent to the GUI node as an observation message and is indistinguishable from messages submitted by regular sensors: the raw observation enters the DDF network and the state vector is augmented.

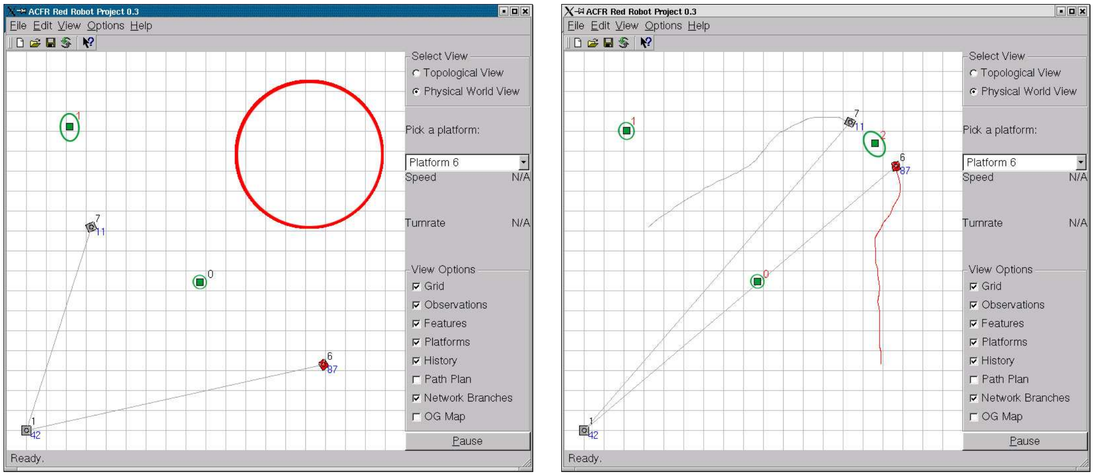
Figure 4(b) displays the response of the platforms to the human-entered feature information. Guided by an "information surfing" controller, the platforms converge onto it. Note that the location uncertainty of the new feature has decreased after due to the actual observations made by the robots.

The other option of entering information into the network is the posterior method. In this method the human operator graphically alters the state of a feature. A node update message is then sent to the linked nodes which will update their state vectors as well. The human operator can directly move the means of features by dragging them on the screen or modify the uncertainty ellipses by shrinking or enlarging them.

5 Conclusions

Main points:

- Human involvement is an integral part of the operation of an ASN and making it scalable is crucial to the overall scalability of the system.
- If the data fusion architecture is decentralized then connecting an operator GUI to it is a straightforward procedure.
- The decentralized control techniques demonstrated in this paper are a natural extension of the DDF architecture. This work shows that, just like in the area of data fusion, the techniques based in



(a) (b)

Figure 4: Human DDF node in action: (a) the operator enters a feature unknown to the sensor network and (b) the active nodes respond to its presence by converging onto it

information space and not state space, are more decentralizable and scalable. That means that, in terms of scalability, providing the network with additional information is preferable to issuing commands to the network components.

- While the architecture allows to connect an arbitrary number of human operators to the network with read and write access, the human factors related to maintaining the integrity and consistency of the network data may present a certain challenge.

Future work:

- Validate information-based robot control on a larger sensor network.
- Examine the extend to which the adjustable autonomy methods can be made scalable.
- Apply current techniques to other domains: outdoor vehicles, flight platforms, underwater, etc.

References

[1] H.F. Durrant-Whyte and M. Stevens. Data fusion in decentralised sensing networks. In *4th Int. Conf. on Information Fusion*, Montreal, Canada, 2001.

[2] D. Estrin, R. Govindan, J. S. Heidemann, and S. Kumar. Next century challenges: Scalable coordination in sensor networks. In *Mobile Computing and Networking*, pages 263–270, Seattle, WA, USA, 1999.

[3] T. Fong and C. Thorpe. Robot as partner: Vehicle teleoperation with collaborative control. In *NRL Workshop on Multi-Robot Systems*, pages 195–202, Washington, DC, USA, 2002.

[4] B. Grocholsky. *Information-Theoretic Control of Multiple Sensor Platforms.* Phd, The University of Sydney, 2002.

[5] B. Grocholsky, A. Makarenko, T. Kaupp, and H. Durrant-Whyte. Scalable control of decentralised sensor platforms. In *The 2nd Int. Workshop on Information Processing in Sensor Networks (IPSN'03)*, Palo Alto, CA, USA, 2003.

[6] Y.-C. Ho. Team decision theory and information structures. *Proceedings of the IEEE*, 68(6):644–654, 1980.

[7] B. Horling, R. Vincent, R. Mailler, J. Shen, R. Becker, K. Rawlins, and V. Lesser. Distributed sensor network for real time tracking. In *the 5th Int. Conf. on Autonomous Agents*, pages 417–424, Montreal, Canada, 2001.

[8] D. Kortenkamp, D. Schreckenghost, and C. Martin. User interaction with multi-robot systems. In *NRL Workshop on Multi-Robot Systems*, pages 213–220, Washington, DC, USA, 2002.

[9] A. G. O. Mutambara. *Decentralized estimation and control for multisensor systems.* CRC Press, Boca Raton, 1998.

[10] L. E. Parker. Cooperative motion control for multi-target observation. In *IROS '97*, pages 1591–1598, 1997.

[11] P. Pirjanian and M. J. Mataric. Multi-robot target acquisition using multiple objective behavior coordination. In *ICRA'00*, San Francisco, USA, 2000.

[12] D. Pynadath and M. Tambe. The communicative multiagent team decision problem: Analyzing teamwork theories and models. *Journal of AI Research*, 2002.

[13] W.D. Rencken and H. Durrant-Whyte. A quantitative model for adaptive task allocation in human-computer interfaces. *IEEE Trans. on Systems, Man, and Cybernetics*, 23(4):1072–1090, 1993.

[14] J. C. Scholtz. Human-robot interaction: Creating synergistic cyber forces. In *NRL Workshop on Multi-Robot Systems*, pages 177–184, Washington, DC, USA, 2002.

[15] R.G. Simmons, D. Apfelbaum, W. Burgard, D. Fox, M. Moors, S. Thrun, and H. Younes. Coordination for multi-robot exploration and mapping. In *AAAI Nat. Conf. on Artificial Intelligence*, pages 852–858, Austin, TX, USA, 2000.

[16] S. Sukkarieh, E. Nettleton, J.-H. Kim, M. Ridley, A. Goktogan, and H. Durrant-Whyte. The ANSER project – multi UAV data fusion. *Int. J. of Robotics Research*, to be published, 2003.

[17] F. Zhao, J. Shin, and J. Reich. Information-driven dynamic sensor collaboration for tracking applications. *IEEE Signal Processing Magazine*, 2002.