Decentralized Control of Cooperative Robotic Vehicles: Theory and Application

John T. Feddema, *Member, IEEE*, Chris Lewis, *Member, IEEE*, and David A. Schoenwald, *Member, IEEE*

Abstract—This paper describes how decentralized control theory can be used to analyze the control of multiple cooperative robotic vehicles. Models of cooperation are discussed and related to the input/output reachability, structural observability, and controllability of the entire system. Whereas decentralized control research in the past has concentrated on using decentralized controllers to partition complex physically interconnected systems, this work uses decentralized methods to connect otherwise independent nontouching robotic vehicles so that they behave in a stable, coordinated fashion. A vector Liapunov method is used to prove stability of two examples: the controlled motion of multiple vehicles along a line and the controlled motion of multiple vehicles in formation. Also presented are three applications of this theory: controlling a formation, guarding a perimeter, and surrounding a facility.

Index Terms—Autonomous vehicles, cooperative robots, decentralized control, stability.

I. INTRODUCTION

IN RECENT years, there has been considerable interest in the control of multiple cooperative robotic vehicles, the vision being that multiple robotic vehicles can perform tasks faster and more efficiently than a single vehicle. This is best illustrated in a search and rescue mission where multiple robotic vehicles spread out and search for a missing aircraft. During the search, the vehicles share information about their current location and the areas that they have already visited. If one vehicle's sensor detects a strong signal indicating the presence of the missing aircraft, it may tell the other vehicles to concentrate their efforts in a particular area.

Other types of cooperative tasks range from moving large objects [1] to troop hunting behaviors [2]. Conceptually, large groups of mobile vehicles outfitted with sensors should be able to automatically perform military tasks like formation following, localization of chemical sources, demining, target assignments, autonomous driving, perimeter control, surveillance, and search and rescue missions [3]–[6]. Simulation and experiments have shown that by sharing concurrent sensory information, the group can better estimate the shape of a chemical plume and, therefore, localize its source [7]. Similarly, for a search and rescue operation, a moving target is more easily found using an organized team [8], [9].

In the field of distributed mobile robot systems, much research has been performed, and summaries are given in [10] and [11]. The strategies of cooperation encompass theories from such diverse disciplines as artificial intelligence, game theory/economics, theoretical biology, distributed computing/control, animal ethology, and artificial life.

Much of the early research concentrated on animal-like cooperative behavior. Arkin [12] studied an approach to "cooperation without communication" for multiple mobile robots that are to forage and retrieve objects in a hostile environment. This behavior-based approach was extended in [13] to perform formation control of multiple robot teams. Motor schemas such as avoid-static-obstacle, avoid-robot, move-to-goal and maintain-formation were combined by an arbiter to maintain the formation while driving the vehicles to their destination. Each motor schema contained parameters such as an attractive or repulsive gain value, a sphere of influence, and a minimum range that were selected by the designer. "When interrobot communication is required, the robots transmit their current position in world coordinates with updates as rapidly as required for the given formation speed and environmental conditions." [13]

Another behavior-based approach includes Kube and Zhang [14]. Much of their study examined comparisons of behaviors of different social insects such as ants and bees. They considered a box-pushing task and utilized a subsumption approach [15], [16] as well as adaptive logic networks (ALN). Similar studies using analogs to animal behavior can be found in Fukuda *et al.* [17]. Noreils [18] dealt with robots that were not necessarily homogeneous. His architecture consisted of three levels: functional level, control level, and planner level. The planner level was the high-level decision maker. Most behavior-based approaches do not include a formal development of the system controls from a stability point of view. Many of the schemes such as the subsumption approach rely on stable controls at a lower level while providing coordination at a higher level.

More recently, researchers have begun to take a system-controls perspective and analyze the stability of multiple vehicles when driving in formations. Chen and Luh [19] examined decentralized control laws that drove a set of holonomic mobile robots into a circular formation. A conservative stability requirement for the sample period is given in terms of the damping ratio and the undamped natural frequency of the system. Similarly, Yamaguchi studied line formations [20] and general formations [21] of nonholonomic vehicles, as did Yoshida *et al.* [22]. Decentralized control laws using a potential field approach to guide vehicles away from obstacles can be found in [23] and

[24]. In these studies, only continuous-time analyses have been performed, assuming that the relative position between vehicles and obstacles can be measured at all times.

Another way of analyzing stability is to investigate the convergence of a distributed algorithm. Beni and Liang [25] prove the convergence of a linear swarm of asynchronous distributed autonomous agents into a synchronously achievable configuration. The linear swarm is modeled as a set of linear equations that are solved iteratively. Their formulation is best applied to resource allocation problems that can be described by linear equations. Liu *et al.* [26] provide conditions for convergence of an asynchronous swarm in which swarm "cohesiveness" is the stability property under study. Their paper assumes position information is passed between nearest neighbors only, and proximity sensors prevent collisions.

Also of importance is the recent research combining graph theory with decentralized controls. Most cooperative mobile robot vehicles have wireless communication, and simulations have shown that a wireless network of mobile robots can be modeled as an undirected graph [27]. These same graphs can be used to control a formation. Desai *et al.* [28], [29] used directed graph theory to control a team of robots navigating terrain with obstacles while maintaining a desired formation and changing formations when needed. When changing formations, the transition matrix between the current adjacency matrix and all possible control graphs are evaluated. In the next section, the reader will notice that graph theory is also used in this paper to evaluate the controllability and observability of the system.

Other methods for controlling a group of vehicles range from distributed autonomy [30] to intelligent squad control and general purpose cooperative mission planning [31]. In addition, satisfaction propagation is proposed in [32] to contribute to adaptive cooperation of mobile distributed vehicles. The decentralized localization problem is examined by Roumeliotis and Bekey [33] and Bozorg *et al.* [34] via the use of distributed Kalman filters. Uchibe *et al.* [35] use canonical variate analysis (CVA) for this same problem.

In this paper, we address the stable control of multiple vehicles using large-scale decentralized control techniques. The objective is to first analyze whether a large group of robotic vehicles, which is spread over an extensive spatial terrain, is input/output reachable and structurally controllable and observable. This depends on the communication paths available between vehicles and the information transmitted and received. Once we know that a system is structurally controllable and observable, we use provably asymptotically stable control laws to regulate the coordinated motion of the vehicles. The stability of these control laws is proven with a vector Liapunov technique.

The approach taken in this paper differs from previous efforts in that the analysis techniques are scalable to very large dimensions and they ensure stability even under structural perturbations such as communication failures and parameter variations. While this depth of analysis may not be necessary when controlling smaller numbers of vehicles, the formalism introduced here is necessary when tens to hundreds, possibly thousands, of vehicles are involved. With hundreds of vehicles, it is not feasible to experimentally determine the interaction gains and the communication rates between vehicles necessary to stabilize the system. The analysis techniques discussed in the following sections allow the system designer to determine the required sampling periods for communication and control and the theoretical limits on the interaction gains between each vehicle. Both continuous time and discrete time examples are given with stability regions defined for up to 10 000 vehicles. While this paper only addresses linear problems, these analysis techniques have also been applied to more complex nonlinear problems [44], opening a new and exciting area of research in nonlinear control of large-scale swarms of vehicles.

The following section describes the model of cooperation used in the analysis. This is followed by a stability analysis of two cases: the controlled motion of multiple vehicles along a straight line, and the controlled motion of multiple vehicles in a formation. The remaining sections discuss how this theory has been implemented on a test platform for several applications.

II. MODEL OF COOPERATION

In this section, a group of robotic vehicles is modeled as a large dimensional interconnected system. It is a well-known fact that testing controllability and observability is a difficult numerical problem for large dimensions. Because of this, simple binary tests have been developed which test for input and output reachability and structural controllability and observability [36]. These tests are valid not only for the nominal nonlinear system but also for perturbed systems where the exact system parameters are unknown. Once controllability and observability have been assured, vector Liapunov techniques exist for testing asymptotic stability of the overall system. The analysis below shows some of the progress made in understanding how these techniques can be used in the design of large-scale distributed cooperative robotic vehicular systems.

Suppose that the overall system is denoted by

$$\mathbf{S} : \dot{x} = f(t, x, u)$$
$$y = h(t, x) \tag{1}$$

where $x(t) \in \Re^n$ is the state of $\mathbf{S}$ (e.g., x, y position, orientation, and linear and angular velocities of all vehicles) at time $t \in T$, $u(t) \in \Re^m$ are the inputs (e.g., the commanded wheel velocities of all vehicles), and $y(t) \in \Re^\ell$ are the outputs (e.g., Global Positioning System (GPS)-measured x, y position of all vehicles). The function $f : T \times \Re^n \times \Re^m \to \Re^n$ describes the dynamics of $\mathbf{S}$, and the function $h : T \times \Re^n \to \Re^\ell$ describes the observations of $\mathbf{S}$. We can partition the system into N interconnected subsystems given by

$$\mathbf{S} : \dot{x}_i = f_i(t, x_i, u_i) + \tilde{f}_i(t, x, u), \quad i \in \{1, \dots, N\}$$
$$y_i = h_i(t, x_i) + \tilde{h}_i(t, x) \tag{2}$$

where $x_i(t) \in \Re^{n_i}$ is the state of the ith subsystem $\mathbf{S}_i$ at time $t \in \Re$, $u_i(t) \in \Re^{m_i}$ are the inputs to $\mathbf{S}_i$, and $y_i(t) \in \Re^{\ell_i}$ are the outputs of $\mathbf{S}_i$. The function $f_i : T \times \Re^{n_i} \times \Re^{m_i} \to \Re^{n_i}$ describes the dynamics of $\mathbf{S}_i$, and the function $\tilde{f}_i : T \times \Re^n \times \Re^m \to \Re^{n_i}$ represents the dynamic interaction of $\mathbf{S}_i$ with the rest of the system $\mathbf{S}$. The function $h_i : T \times \Re^{n_i} \to \Re^{\ell_i}$ represents observations at $\mathbf{S}_i$ derived only from local-state variables of $\mathbf{S}_i$, and the function $\tilde{h}_i : T \times \Re^n \to \Re^{\ell_i}$ represents observations at

$\mathbf{S}_i$ derived from the rest of $\mathbf{S}$. The N independent subsystems are denoted as

$$\mathbf{S}_i : \dot{x}_i = f_i(t, x_i, u_i), \quad i \in \{1, \ldots, N\}$$
$$y_i = h_i(t, x_i). \tag{3}$$

To determine input and output reachability and structural controllability and observability, we want to determine which inputs, outputs, and state variables affect each other through either a linear or nonlinear relation. To perform this operation, it is convenient to write the state interconnection function as

$$\tilde{f}_i(t, x, u) = \tilde{f}_i\Big(t, \bar{a}_{i1}x_1, \bar{a}_{i2}x_2, \ldots, \bar{a}_{iN}x_N,$$
$$\bar{b}_{i1}u_1, \bar{b}_{i2}u_2, \ldots, \bar{b}_{iN}u_N\Big)$$
$$i \in \{1, \ldots, N\} \tag{4}$$

where the matrices $\bar{a}_{ij} \in B^{n_i \times n_j}$ and $\bar{b}_{ij} \in B^{n_i \times m_j}$ and the elements of the matrices are

$$(\bar{a}_{ij})_{pq} = \begin{cases} 1, & (x_j)_q \text{ occurs in } \left(\tilde{f}_i(t, x, u)\right)_p \\ 0, & (x_j)_q \text{ does not occur in } \left(\tilde{f}_i(t, x, u)\right)_p \end{cases} \tag{5}$$

$$(\bar{b}_{ij})_{pr} = \begin{cases} 1, & (u_j)_r \text{ occurs in } \left(\tilde{f}_i(t, x, u)\right)_p \\ 0, & (u_j)_r \text{ does not occur in } \left(\tilde{f}_i(t, x, u)\right)_p \end{cases} \tag{6}$$

where $q \in \{n_j\}$, $p \in \{n_i\}$, and $r \in \{m_j\}$. Similarly, the observation interconnection function may be written as

$$\tilde{h}_i(t, x) = \tilde{h}_i(t, \bar{c}_{i1}x_1, \bar{c}_{i2}x_2, \ldots, \bar{c}_{iN}x_N), \quad i \in \{1, \ldots, N\} \tag{7}$$

where $\bar{c}_{ij} \in B^{\ell_i \times n_j}$ and the elements of the matrix are

$$(\bar{c}_{ij})_{zq} = \begin{cases} 1, & (x_j)_q \text{ occurs in } \left(\tilde{h}_i(t, x)\right)_z \\ 0, & (x_j)_q \text{ does not occur in } \left(\tilde{h}_i(t, x)\right)_z \end{cases} \tag{8}$$

where $q \in \{n_j\}$ and $z \in \{\ell_i\}$. Using these definitions, the interconnection matrix of $\mathbf{S}$ is a binary $(n+m+\ell) \times (n+m+\ell)$ matrix defined as

$$E = \begin{bmatrix} \bar{A} & \bar{B} & 0 \\ 0 & 0 & 0 \\ \bar{C} & 0 & 0 \end{bmatrix} \tag{9}$$

where the matrices $\bar{A} = (\bar{a}_{ij})$, $\bar{B} = (\bar{b}_{ij})$, and $\bar{C} = (\bar{c}_{ij})$. The three rows and columns of the interconnection matrix represent the coupling between the state, input, and output variables. For large-scale systems, the interconnection matrix E is often represented as a directed graph mapping state, input, and output variables from one subsystem to another subsystem. By searching this directed graph, it is possible to check for input and output reachability of the system [36]. Input reachability tells us if we can reach all the state variables from the input variables, while output reachability tells us if we can reach all the output variables from the state variables.

Mathematically, it is possible to check for input and output reachability using the reachability matrix

$$R = E \vee E^2 \vee \cdots \vee E^s = \begin{bmatrix} F & G & 0 \\ 0 & 0 & 0 \\ H & \theta & 0 \end{bmatrix} \tag{10}$$

where $s = n + m + \ell$, $E^k = E^{k-1} \wedge E$, $\vee$ is the Boolean OR operator ($0 \vee 0 = 0, 0 \vee 1 = 1 \vee 0 = 1 \vee 1 = 1$), and $\wedge$ is the Boolean AND operator ($1 \wedge 1 = 1, 0 \wedge 1 = 1 \wedge 0 = 0 \wedge 0 = 0$). For two $s \times s$ binary matrices $A = (a_{ij})$ and $B = (b_{ij})$, the Boolean operations $C = (c_{ij}) = A \wedge B$ and $D = (d_{ij}) = A \vee B$ are defined by $c_{ij} = \vee_{k=1}^{s}(a_{ik} \wedge b_{kj})$ and $d_{ij} = a_{ij} \vee b_{ij}$.

The system $\mathbf{S}$ is input reachable if and only if the binary matrix G has no zero rows, and it is output reachable if and only if the binary matrix H has no zero rows. The system $\mathbf{S}$ is input-output reachable if and only if the binary matrix θ has neither zero rows nor zero columns. A system is structurally controllable if it is input reachable and the corresponding directed graph has no dilations, essentially meaning that there are enough input variables available to independently control all state variables. More formally, a directed graph $D = (U \cup X, E)$ is said to have a dilation if there exists a subset $X_k \subseteq X$, such that the number of distinct vertices of D from which a vertex in X_k is reachable, is less than the number of vertices of X_k. In this definition, the set of input variables is U, the set of state variables is X, and E is the set of edges connecting the set of vertices $U \cup X$. No dilation exists when the generic rank $\rho([\tilde{A} \quad \tilde{B}]) = n$ where $\tilde{A}$ and $\tilde{B}$ are the same as $\bar{A}$ and $\bar{B}$ except the "1" elements can take on any value. Similarly, a system is structurally observable if it is output reachable and the corresponding directed graph $D = (X \cup Y, E)$ has no dilations (i.e., generic rank $\rho([\tilde{A}^T \quad \tilde{C}^T]) = n$).

Feedback may be added to the system with

$$u_i = k_i(t, y_i) + \tilde{k}_i(t, y), \quad i \in \{1, \ldots, N\} \tag{11}$$

where the feedback interconnection function is given by

$$\tilde{k}_i(t, y) = \tilde{k}_i(t, \bar{k}_{i1}y_1, \bar{k}_{i2}y_2, \ldots, \bar{k}_{iN}y_N) \tag{12}$$

and $\bar{k}_{ij} \in B^{m_i \times \ell_j}$ and the elements of the matrix are

$$(\bar{k}_{ij})_{rz} = \begin{cases} 1, & (y_j)_z \text{ occurs in } \left(\tilde{k}_i(t, y)\right)_r \\ 0, & (y_j)_z \text{ does not occur in } \left(\tilde{k}_i(t, y)\right)_r \end{cases} \tag{13}$$

where $r \in \{m_i\}$ and $z \in \{\ell_j\}$. With the feedback interconnection matrix denoted by $\bar{K} = (\bar{k}_{ij})$, the system interconnection matrix becomes

$$E = \begin{bmatrix} \bar{A} & \bar{B} & 0 \\ 0 & 0 & \bar{K} \\ \bar{C} & 0 & 0 \end{bmatrix} \tag{14}$$

Again, the reachability matrix ($R = E \vee E^2 \vee \cdots \vee E^s$) may be used to determine input/output reachability and structural observability and controllability.

Note that in most prior research on decentralized control, the state interconnection function $\tilde{f}_i(t, x, u)$ is nonzero, while the feedback interconnection function $\tilde{k}_i(t, y_i)$ is zero. In other
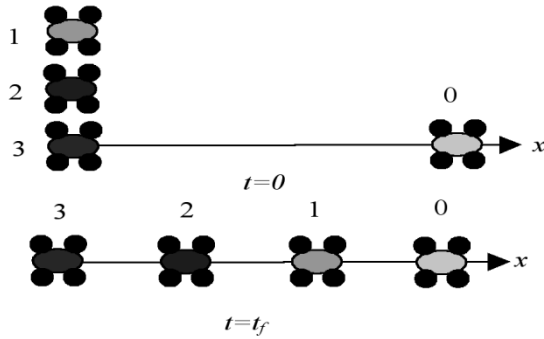
Fig. 1.   One-dimensional control problem. The top line is the initial state. The second line is the desired final state. Vehicles 0 and 3 are boundary conditions. Vehicles 1 and 2 spread out along the line by using only the position of their left and right neighbor.

words, typically it is desirable to stabilize a complex interconnected system using only decentralized controllers. However, in the case of multiple nontouching robotic vehicles, we have many noninterconnected systems, but we want to connect these systems through communication so that they behave in a coordinated fashion. For this case, the state interconnection function $\hat{f}_i(t, x, u)$ is zero and feedback interconnection function $\tilde{k}_i(t, y_i)$ is nonzero.

As an example, let us analyze a simple one-dimensional problem in which a linear chain of interdependent vehicles is to spread out along a line as shown in Fig. 1. The objective is to spread out evenly along the line using only information from the nearest neighbor.

Assume that the vehicle's plant is modeled as a simple integrator and the commanded input is the desired velocity of the vehicle along the line. A feedback loop and a proportional gain K_p are used to control each vehicle's position. Fig. 2 shows a block diagram of the control system. The dynamics of each subsystem are

$$\mathbf{S}_i : \dot{x}_i = -K_p x_i + K_p u_i, \qquad i \in \{1, \ldots, N\}$$
$$y_i = x_i \tag{15}$$

where x_i is the position of the ith vehicle, u_i is the control input, and y_i is the observation. Assume the control of each vehicle is a function of the two nearest vehicles' observed positions and the boundary conditions on the first and last vehicle are 1 and 0, respectively. Then

$$u_1 = \gamma + \gamma y_2$$
$$u_i = \gamma (y_{i-1} + y_{i+1}) \quad i \in \{2, \ldots, N-1\}$$
$$u_N = \gamma y_{N-1} \tag{16}$$

where γ is the interaction gain between vehicles. The interconnection matrix of this system is

$$\bar{E} = \begin{bmatrix} \bar{A} & \bar{B} & 0 \\ 0 & 0 & \bar{K} \\ \bar{C} & 0 & 0 \end{bmatrix} \tag{17}$$
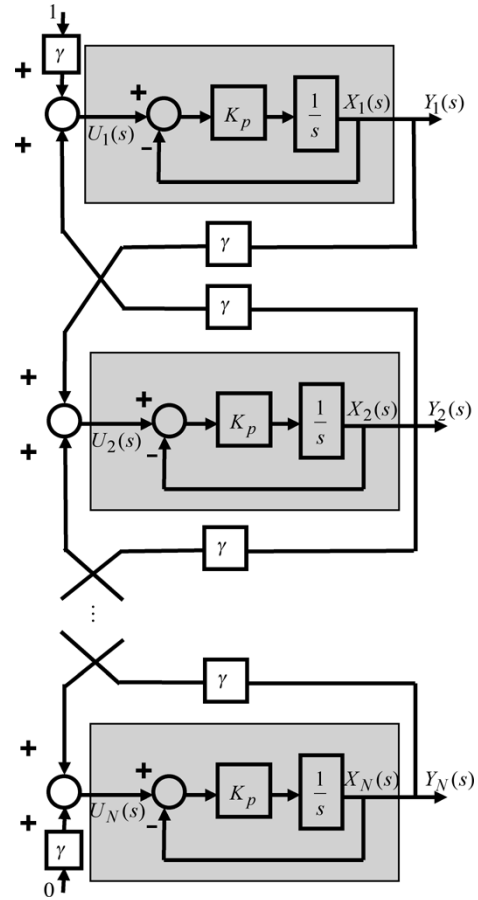

Fig. 2.   Control block diagram of the N vehicle interaction problem.

where $\bar{A} = \bar{B} = \bar{C} = I$

$$\bar{K} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 1 & 0 & 1 & & 0 \\ 0 & 1 & 0 & & 0 \\ \vdots & & & \ddots & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \in B^{N \times N} \tag{18}$$

and I is the identity matrix of dimension $N \times N$. In this problem, the reachability matrix $R = E \vee E^2 \vee \cdots \vee E^s$ is a $3N \times 3N$ matrix of all ones, meaning that any state, input, or output can reach any other state, input, or output. Since the system is input and output reachable and there are no dilations, we know that the system is structurally observable and controllable.

III. STABILITY OF LARGE-SCALE SYSTEMS

Once we know that a system is structurally observable and controllable, the next question to ask is that of connective stability. Will the overall system be globally asymptotically stable under structural perturbations? Analysis of connective stability is based upon the concept of vector Liapunov functions, which associate several scalar functions with a dynamic system in such a way that each function guarantees stability in different portions of the state space. The objective is to prove that there exist Liapunov functions for each of the individual subsystems, and then prove that the vector sum of these Liapunov functions is a Liapunov function for the entire system.

To simplify matters, we will assume that the control function has already been chosen and the closed-loop dynamics of the system can be written as

$$\mathbf{S} : \dot{x}_i = g_i(t, x_i) + \tilde{g}_i(t, x), \quad i \in \{1, \dots, N\}. \quad (19)$$

The interconnection function can be written as

$$\tilde{g}_i(t, x) = \tilde{g}_i(t, \bar{e}_{i1}x_1, \bar{e}_{i2}x_2, \dots, \bar{e}_{iN}x_N) \quad i \in \{1, \dots, N\} \quad (20)$$

where $\bar{e}_{ij} \in B^{n_i \times n_j}$ and the elements of the fundamental interconnection matrix $\bar{E} = (\bar{e}_{ij})$ are

$$(\bar{e}_{ij})_{pq} = \begin{cases} 1, & (x_j)_q \text{ occurs in } (\tilde{g}_i(t, x, u))_p \\ 0, & (x_j)_q \text{ does not occur in } (\tilde{g}_i(t, x, u))_p. \end{cases} \quad (21)$$

where $q \in \{n_j\}$ and $p \in \{n_i\}$.

The structural perturbations of $\mathbf{S}$ are introduced by assuming that the elements of the fundamental interconnection matrix that are one can be replaced by any number between zero and one, i.e.,

$$e_{ij} = \begin{cases} [0, 1], & \bar{e}_{ij} = 1 \\ 0, & \bar{e}_{ij} = 0. \end{cases} \quad (22)$$

Therefore, the elements e_{ij} represent the strength of coupling between the individual subsystems. A system is connectively stable if it is stable in the sense of Liapunov for all possible $E = (e_{ij})$ [36]. In other words, if a system is connectively stable, it is stable even if an interconnection becomes decoupled, i.e., $e_{ij} = 0$, or if interconnection parameters are perturbed, i.e., $0 < e_{ij} < 1$. This is potentially very powerful, as it proves that the system will be stable if an interconnection is lost through communication failure.

For linear systems such as in Fig. 2, the linear system dynamics may be written as

$$\mathbf{S} : \dot{x}_i = A_i x_i + \sum_{j=1}^{N} e_{ij} A_{ij} x_j, \quad i \in \{1, \dots, N\} \quad (23)$$

and the Liapunov function for each individual subsystem is $v_i(x_i) = (x_i^T H_i x_i)^{1/2}$ where H_i is a positive definite matrix. For the system $\mathbf{S}$ to be connectively stable, the following test matrix $W = (w_{ij})$ must be an M matrix (i.e., all leading principal minors must be positive) [36]

$$w_{ij} = \begin{cases} \frac{\lambda_m(G_i)}{2\lambda_M(H_i)} - \bar{e}_{ii} \lambda_M^{1/2} \left(A_{ii}^T A_{ii} \right), & i = j \\ -\bar{e}_{ij} \lambda_M^{1/2} \left(A_{ij}^T A_{ij} \right), & i \neq j \end{cases} \quad (24)$$

where the symmetric positive definite matrix G_i satisfies the Liapunov matrix equation $A_i^T H_i + H_i A_i = -G_i$ and $\lambda_m(\bullet)$ and $\lambda_M(\bullet)$ are the minimum and maximum eigenvalues of the corresponding matrices.

In the example, the test matrix becomes

$$W = \begin{bmatrix} K_p & -K_p\gamma & 0 & \cdots & 0 \\ -K_p\gamma & K_p & -K_p\gamma & & \vdots \\ 0 & -K_p\gamma & K_p & & 0 \\ \vdots & & & \ddots & -K_p\gamma \\ 0 & \cdots & 0 & -K_p\gamma & K_p \end{bmatrix}. \quad (25)$$

For $N = 2$, this test matrix is an M matrix (i.e., the system is connectively stable) if $|\gamma| < 1$. For $N = 3$, the system is
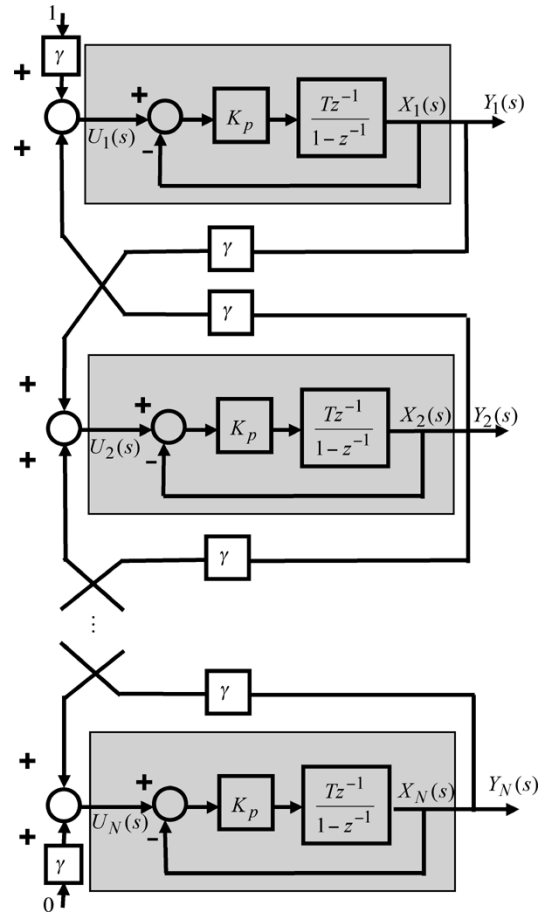

Fig. 3. Discrete time-control block diagram of N vehicle interaction problem.

connectively stable if $|\gamma| < 1/\sqrt{2}$. For $N = 4$, the system is connectively stable if $|\gamma| < 0.618$. Notice how the range of the interaction gain gets smaller for larger sized systems. It can be shown that as the number of interacting vehicles increases, the interaction gain range reaches a limit of $|\gamma| \leq 0.5$ for infinite numbers of vehicles. Since the structural perturbations or parameter uncertainties are included in the term γ, this example shows the robustness of the control to variations in interaction gain decreases as the number of vehicles increase.

This same analysis can also be performed in the discrete domain [37]. Consider a discrete dynamic system described by

$$\mathbf{S} : x_i(k+1) = A_{ii} x_i(k) + \sum_{j=1}^{N} e_{ij} A_{ij} x_j(k), \quad i \in \{1, \dots, N\} \quad (26)$$

and a Liapunov function $v_i(x_i) = (x_i^T H_i x_i)^{1/2}$. The test matrix is

$$w_{ij} = \begin{cases} \xi_i, & i = j \\ -e_{ij}\xi_{ij}, & i \neq j \end{cases} \quad (27)$$

where $\xi_i = 1 - \sqrt{1 - 1/(\lambda_M(H_i^*))}$, $\xi_{ij} = \lambda_M^{1/2}(A_{ij}^T A_{ij})$ and $A_{ii}^T H_i^* A_{ii} - H_i^* = -I$ and the superscript * denotes the Hermitian operator.

Inserting a zeroth-order hold function before the integrator in Fig. 2, we can transform our example problem above into the discrete time domain as shown in Fig. 3. The sampling period is denoted by T. The sampling period is both the communica-
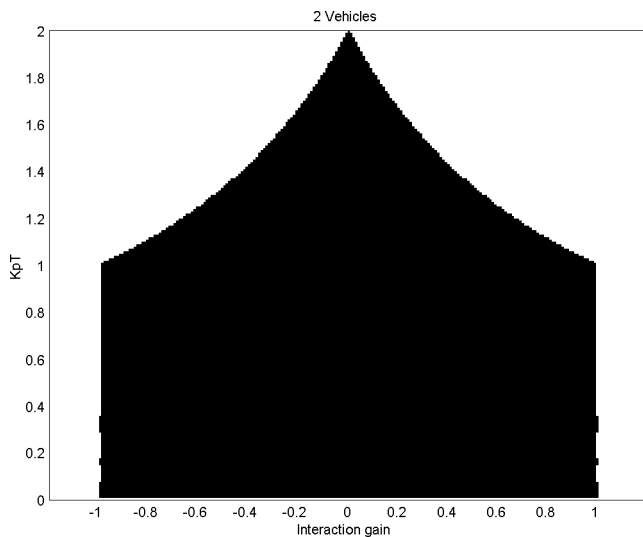
Fig. 4. Stability region for the $N = 2$ vehicle case.

Fig. 5. Stability region for the $N = 10\,000$ vehicle case.

tion and position update sample time. The state equations of the system are

$$\mathbf{S} : x_1(k+1) = (1 - K_pT)\,x_1(k) + \gamma K_pT x_2(k) + \gamma$$
$$x_i(k+1) = (1 - K_pT)\,x_i(k) + \gamma K_pT x_{i-1}(k)$$
$$+ \gamma K_pT x_{i+1}(k), \qquad i \in \{2, \dots, N-1\}$$
$$x_N(k+1) = (1 - K_pT)\,x_N(k) + \gamma K_pT x_{N-1}(k) \quad (28)$$
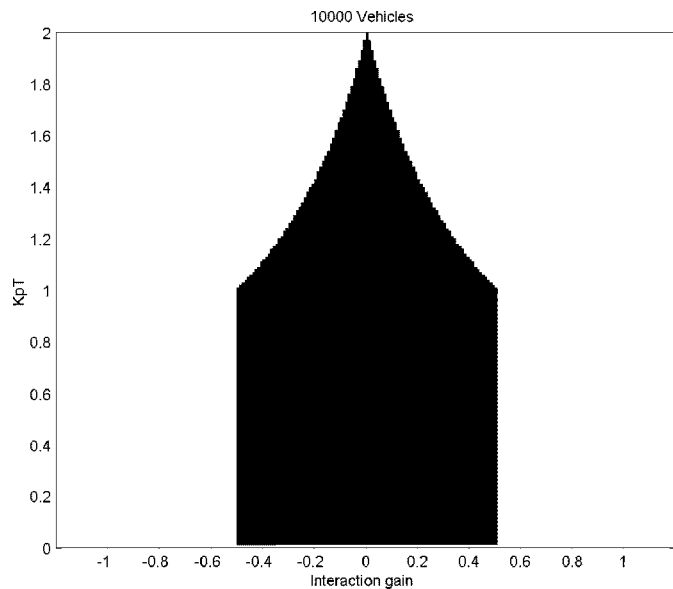
If $0 < K_pT \leq 1$, the resulting test matrix is

$$W = \begin{bmatrix} K_pT & -K_pT\gamma & 0 & \cdots & 0 \\ -K_pT\gamma & K_pT & -K_pT\gamma & & \vdots \\ 0 & -K_pT\gamma & K_pT & & 0 \\ \vdots & & & \ddots & -K_pT\gamma \\ 0 & \cdots & 0 & -K_pT\gamma & K_pT \end{bmatrix} \quad (29)$$

and if $1 < K_p \leq 2$, the test matrix is as shown in (30) at the bottom of the page. For $N = 2$, the test matrix is an M matrix, and the system is connectively stable if

$$|\gamma| < \begin{cases} 1, & 0 < K_pT \leq 1 \\ \frac{2}{K_pT} - 1, & 1 < K_pT \leq 2. \end{cases} \quad (31)$$

Fig. 4 illustrates the stability region for the case of $N = 2$. The dark region represents stable combinations of the interaction gain γ and K_pT (proportional control gain multiplied by the sampling period). The white region represents unstable combinations of γ and K_pT. We refer to the dark region as a stability "house" due to the shape of the stable zone. The size of this stability house varies only with N. As N is increased, the house gets smaller in width but maintains the same height and shape.

The size of the stability house is a measure of the robustness of the closed-loop system to parameter variations in interaction gain γ, sampling period T, and proportional control gain K_p. Fig. 5 shows the stability region for $N = 10\,000$.

For this particular example, another way to check the stability of this linear system is to check that the eigenvalues of the system matrix A are within the unit circle. There is a special formula ([38, p. 59]) for the eigenvalues of A given by

$$\lambda_i(A) = 1 - K_pT + 2K_pT\gamma \cos\left(\frac{i\pi}{N+1}\right), \qquad i = 1, \dots, N. \quad (32)$$

From this formula, we can see that as $N \to \infty$, the cosine term becomes unity. This implies that γ must stay between -0.5 and 0.5 for K_pT less than one in order to maintain stability. For K_pT greater than one, the admissible γ values taper off parabolically (the sloped "roof") until $K_pT = 2$.

It is instructive to look at the step response of one and two vehicles to understand why the interaction gain limits on the stability house converges so quickly to ± 0.5. The step responses of a single vehicle with varying K_pT are shown in Fig. 6(a)–(c). A single vehicle is stable when $0 < K_pT < 2$; however, the step response will overshoot for $1 < K_pT < 2$. The step responses of two interconnect vehicles with the same values of K_pT are shown in Fig. 6(d)–(f). With an interaction gain of 0.5, two vehicles are stable if $0 < K_pT < 1.33$ (note this range is smaller than for a single vehicle). When $K_pT = 1.7$, we can see that the overshoot of each vehicle is amplified by the other until both

$$W = \begin{bmatrix} (2 - K_pT) & -K_pT\gamma & 0 & \cdots & 0 \\ -K_pT\gamma & (2 - K_pT) & -K_pT\gamma & & \vdots \\ 0 & -K_pT\gamma & (2 - K_pT) & & 0 \\ \vdots & & & \ddots & -K_pT\gamma \\ 0 & \cdots & 0 & -K_pT\gamma & (2 - K_pT) \end{bmatrix} \quad (30)$$
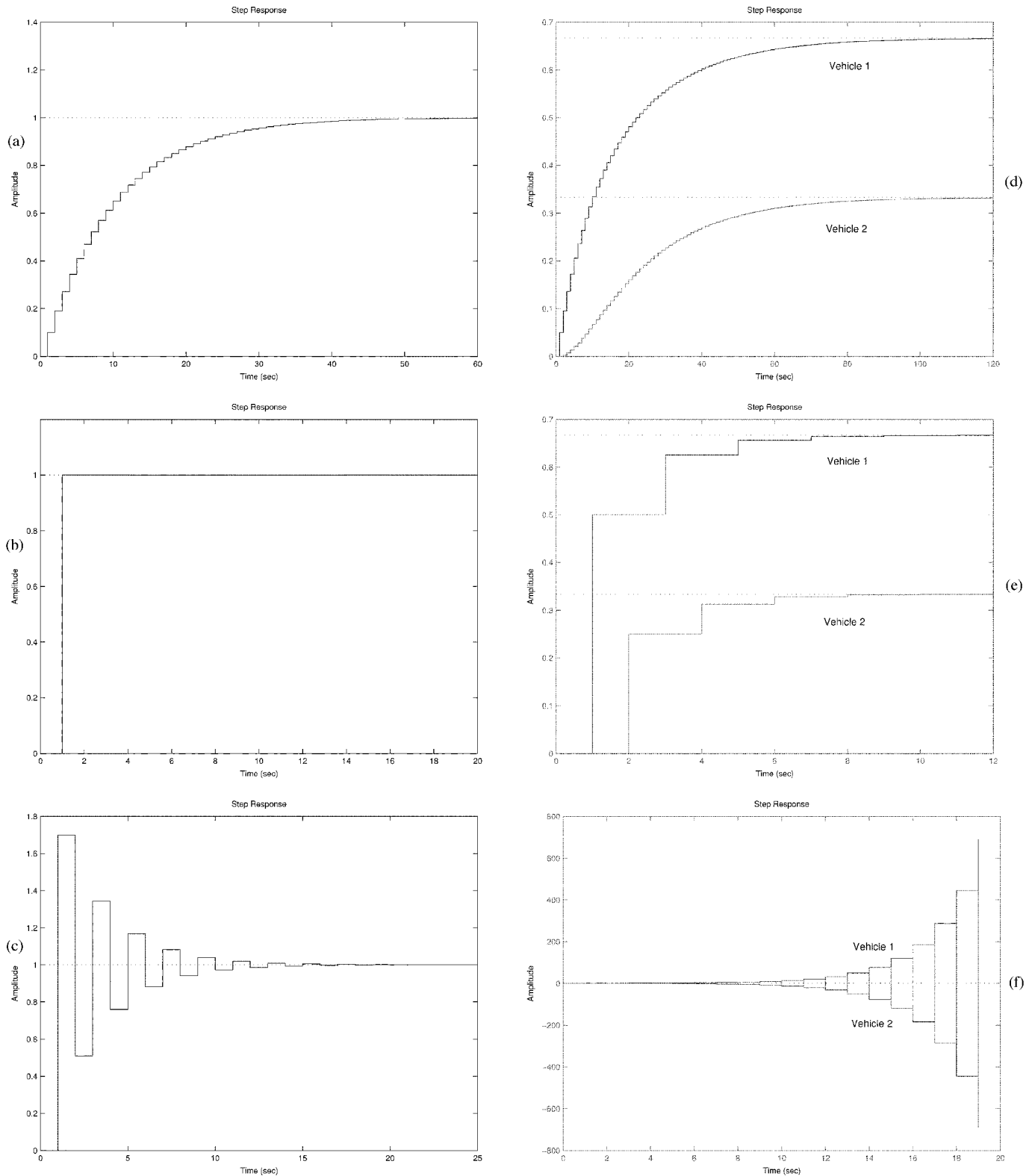
Fig. 6. Step response. (a) Single vehicle with $K_pT = 0.1$. (b) Single vehicle with $K_pT = 1$. (c) Single vehicle with $K_pT = 1.7$. (d) Two vehicles with $K_pT = 0.1$ and $\gamma = 0.5$. (e) Two vehicles with $K_pT = 1$ and $\gamma = 0.5$. (f) Two vehicles with $K_pT = 1.7$ and $\gamma = 0.5$.

go unstable. When more vehicles are involved, any amount of overshoot can cause the whole group to go unstable.

It must be remembered that the above example assumed that the sampling period for both communication and position are the same. It can be shown that if the position sampling period is much less than the communication sampling period T, then the stability region is independent of T and only dependent on the interaction gain γ. In the limit, the position feedback loop may be modeled as a continuous time system and the zeroth-order hold may be moved outside the position feedback loop. As long

as the position feedback loop is stable ($K_p > 0$), then there will be no overshoot in driving the vehicle, and the vehicle will stop at the desired position given by $\gamma(x_{i-1} + x_{i+1})$ at each communication sample period. Intuitively, this result is obvious.

Several conclusions can be drawn from this stability analysis. First, asymptotic stability of vehicle positions depends on vehicle responsiveness K_p, communication sampling period T, and vehicle interaction gain γ. If the vehicle is too fast (large K_p), or the sample period is too long (large T), then the vehicles will go unstable. There is a dependence on interaction gain for stability as well. Second, the interaction gains can be used to bunch the vehicles closer together or spread them out. Third, the stability region shrinks as the number of vehicles N increases, but only to a defined limit.

To further demonstrate the power of this stability analysis, let us next consider the stability of a formation-control problem where the desired position of each vehicle is a function of the position of all the vehicles. To simplify the problem, we will assume that the vehicles' x and y positions can be independently controlled. This assumption is valid if each vehicle's position is controlled at a faster servo rate using the inverse Jacobian control law given in the appendix. Considering only the x position of the vehicles, the dynamics of each subsystem is again assumed to be

$$\mathbf{S}_i : \dot{x}_i = -K_p x_i + K_p u_i, \qquad i \in \{1, \ldots, N\}$$
$$y_i = x_i \tag{33}$$

where x_i is the position of the ith vehicle, u_i is the control input, and y_i is the observation. In the previous example, the control of each vehicle is dependent on the position of the two neighboring vehicles. For formation control, the control of each vehicle is a function of all the vehicle positions. Assume the control of each vehicle is a constant position offset x_{io} plus the sum of the position of each vehicle multiplied by an interaction gain γ.

$$u_i = x_{io} + \gamma \sum_{j=1}^{N} y_i, \qquad i \in \{1, \ldots, N\}. \tag{34}$$

In this example, the feedback interaction matrix $\bar{K}$ is a matrix of all ones and the reachability matrix is also a matrix of all ones. Since the system is input and output reachable and there are no dilations, we know that the system is structurally observable and controllable. The resulting stability test matrix is

$$w_{ij} = \begin{cases} K_p(1-\gamma), & i = j \\ -K_p \gamma, & i \neq j \end{cases} \tag{35}$$

and it is an M matrix (i.e., the system is connectively asymptotically stable) if and only if $\gamma < 1/N$. It is interesting to note that when $\gamma = 1/N$, the vehicles will converge to their offset formation position about the average position of the vehicles given by $x_{io} + (1/N) \sum_{i=1}^{N} x_i$. While this condition is stable, it is not asymptotically stable because the *group* does not converge to the origin. In order to make the vehicles converge in their formation and the entire group to move to the origin, the interaction gain $\gamma < 1/N$. Of course, when driving the vehicles in formation from point A to point B, the origin is moved along the line connecting the two points, and the x and y values are computed with respect to the new origin.

Fig. 7. RATLER vehicles around the laptop base station.

IV. EXPERIMENTAL TEST PLATFORMS

To test the analysis provided in the previous sections, a squad of semiautonomous all-terrain vehicles was developed for remote cooperative sensing applications (see Fig. 7). The system has been used to demonstrate the feasibility of using a cooperative team of robotic sentry vehicles to investigate alarms from intrusion detection sensors and to surround and monitor an enemy facility.

The "Roving All-Terrain Lunar Explorer Rover" (RATLER) vehicles are electric, all-wheel drive vehicles with two composite bodies joined by a passive central pivot. This flexible structure, when combined with an aggressive asymmetric tread on custom carbon-composite wheels, provides agile off-road capabilities. With a PC104 Intel 80486, the RATLER vehicles are fully equipped with a wide range of sensors and peripherals. Software on the vehicles is currently a single-threaded DOS-based application for simplicity. The vehicles have been programmed to operate either through teleoperation or autonomously. The RATLER vehicles rely on radio frequency (RF) signals for communications. Currently, the vehicles are outfitted with differential GPS receivers and two spread-spectrum RF modems. One modem is for intervehicle and base-to-vehicle communication, and the other is for the differential GPS correction. Video cameras communicate with the base station via a separate RF video link.

A laptop computer is used as the base station. A Windows NT application was written to control the vehicles from the base station. A graphical user interface (GUI) displays vehicle status information and allows the operator to monitor the vehicles' positions on a Geographic Information System (GIS) map—either aerial photo or topological data, as well as viewing the live video from a selected vehicle (see Fig. 8). Mission-specific control modes such as teleoperation, formation following, autonomous navigation, and perimeter surveillance can be initiated and monitored using this GUI interface.

There are two modes of communication between the base station and the vehicles: a star network and a token-ring network. In the star network, all radio communication is coordinated by the base station. In the token-ring network, each node (either vehicle or base station) speaks only when it receives the token. In our case, an actual token packet was not needed, since each vehicle has an identification number and communication order is determined from this number. All messages are broadcast in half-duplex mode so that each vehicle knows when the other vehicles

Fig. 8. Base station's GUI displays vehicle status, remote sensor status, video, and GPS position on a GIS map.

or the base station has transmitted a message. If a node does not communicate when expected, a timer on the next node expires, signaling that the next node should transmit. The token-ring network is more fault tolerant than the star network, since there is no single point of failure, as there is with the star network. Also, the token-ring network allows the vehicles to continue to operate in perimeter surveillance mode even if the base station is shut down.

V. FORMATION CONTROL

The goal of formation control is to develop a simple user interface that allows a single operator to guide multiple robot vehicles. The ability to maintain a formation is useful for conducting searches and for moving the squad from place to place. This capability has been implemented using the base station's GUI. The decentralized formation control law described in the previous section is used by each vehicle to keep the vehicles in formation while driving the group to a desired destination. To initiate formation control, the operator graphically places the vehicles into a relative formation as shown in Fig. 9. Initially, each vehicle is sent a relative offset and the initial formation location command. Each vehicle determines its own destination by adding its individual offset to the formation command. Subsequent moves only require broadcasting the new formation location command. In the current implementation, orientation is not considered, so the vehicles always traverse nominally the same distance as the formation moves along. A formation always remains aligned to the compass frame rather than to a lead vehicle's frame.

VI. PERIMETER SURVEILLANCE

The goal of robotic perimeter surveillance is to use a cooperative team of robotic sentry vehicles to investigate alarms from miniature intrusion detection sensors (MIDS) [39]. In our tests, we used four different types of MIDS including magnetometer, seismic, passive infrared, and beam-break (or active) infrared. The MIDS are hidden on a defensive perimeter and broadcast unique identification codes to the vehicles when the sensors are tripped.

The vehicles are outfitted with receivers to detect when the sensors are tripped. The vehicles are also programmed to maintain an internal representation of the location of the MIDS sen-
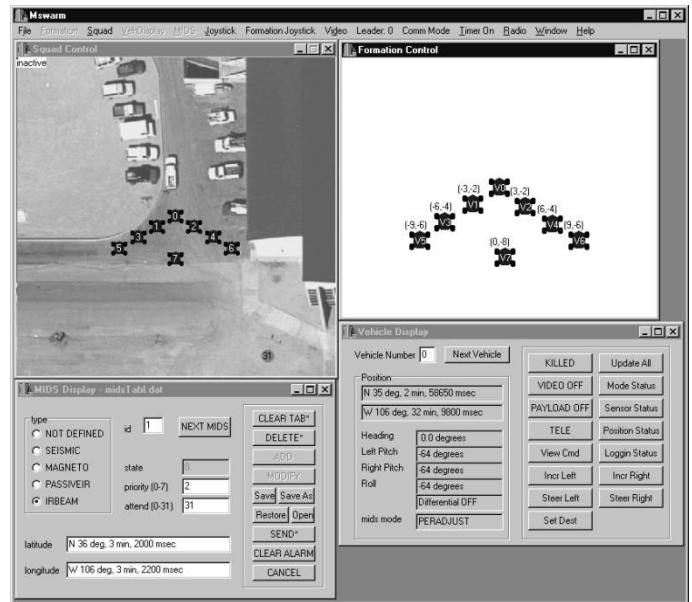

Fig. 9. On the left, the current vehicle locations are displayed. On the right, the user may drag and drop vehicle icons to arrange any desired formation.

Fig. 10. Perimeter being guarded by robot sentries.

sors and the other vehicles. Additional software was also added to the base station to enter and display the MIDS information.

As the sensors are hidden, the operator enters the MIDS attributes at the base station, including:

1) the type of sensor;
2) the GPS location of the sensor;
3) the number of RATLERs to attend the alarm;
4) the priority of the alarm.

The operator draws a perimeter on the GIS map as shown in Fig. 10. The MIDS information and the perimeter region are sent to all the vehicles.

Once the operator places the vehicles in the MIDS sentry mode, the vehicles spread out uniformly along the perimeter maintaining equal distance between their two nearest neighbors using the control law described in the previous sections. An interaction gain of 0.5 is used in the tests. The line that the vehicles are to be controlled on is the curved perimeter in Fig. 10. Differential GPS is used to locate and guide each vehicle. An RF radio on each vehicle is used to broadcast its GPS position to the others. Each vehicle has a communication time slot of 220 ms, which results in a total communication sample period

of 1.1 s for four vehicles and a base station. The differential GPS sample period is 200 ms. As the previous section points out, stable control is guaranteed as long as the differential GPS sample period is faster than the communication sample period, and the vehicle has a faster inner position-control loop based on the GPS position.

When a sensor is alarmed, the vehicles decide, without base station intervention, which of the vehicles can best investigate the intrusion, and how the remaining vehicles should adapt to maintain the perimeter using the same control law.

To maintain the perimeter, the vehicles periodically broadcast (they take turns transmitting every 220 ms) their location and the status of the sensors. In this way, each vehicle can maintain a local representation of where the other vehicles are and which sensors have been tripped. When a vehicle receives an alarm signal, it broadcasts to the other vehicles which alarm has been tripped. If one vehicle receives an alarm and the others do not, the other vehicles will receive the alarm through this broadcast. The base station displays the location of the vehicles and the MIDS sensors on a GIS map. When a MIDS sensor is alarmed, the icon of the MIDS sensor changes color. The user display also indicates which vehicles are moving to investigate the alarm.

The determination of which vehicles attend an alarm is made independent of the base station. When an alarm is received, each vehicle computes its distance to the alarmed sensor as well as the distance of the other vehicles to the same sensor. If the vehicle is closest to the alarmed sensor within the number of vehicles that are to attend the sensor, then it will head toward the alarm. That is, unless a MIDS of higher priority is alarmed, in which case it heads toward the MIDS of higher priority. All of these decisions occur once per second; therefore, a vehicle may be heading toward one alarmed MIDS, when a higher priority MIDS is alarmed, causing it to change directions. When a vehicle is not attending an alarm, it tries to maintain an equidistant position around the perimeter from the other unalarmed vehicles using the control law described in the previous section.

VII. SURROUND TASK

In addition to the formation control and perimeter surveillance tasks, an interactive playbook capability has been developed where the operator can guide individual vehicles or the entire group using drawing tools. In Fig. 11, the operator has used a drawing tool bar to outline the obstacles and indicate goal regions. A simple attractive and repelling potential field algorithm is used to generate the desired paths for the vehicles. The algorithm uses the distance and direction to the nearest goal, obstacle, and neighboring vehicle to determine the gradient used to update the vehicle's position as each vehicle moves from its initial position to the closest goal.

The distance to the closest goal and obstacle is computed as described in [40]. After the closest obstacle, goal, and vehicle positions are computed, the direction of the vehicle is given by

$$\begin{bmatrix} dx \\ dy \end{bmatrix} = K_a \begin{bmatrix} x_a - x_v \\ y_a - y_v \end{bmatrix} - K_r \begin{bmatrix} x_r - x_v \\ y_r - y_v \end{bmatrix} - K_{v2} \begin{bmatrix} x_{cv} - x_v \\ y_{cv} - y_v \end{bmatrix} \tag{36}$$

where (x_v, y_v) is the vehicle's position, (x_a, y_a) is the closest attractive point (goal), (x_r, y_r) is the closest repulsive point

Fig. 11. Base station control window. The initial positions of the vehicles were at the lower left corner of the screen. The vehicles first follow their assigned paths. Once they reach the end of their paths, the vehicles use the obstacles and attractors to navigate to their final positions on the goal attractors. To avoid collision between the vehicles and to uniformly cover the goal attractors, the vehicles are also repulsed by each other. The obstacles are drawn in red, the goals are drawn in green, and the vehicle paths are drawn in black.

(obstacle), (x_{cv}, y_{cv}) is the closest vehicle, and K_a, K_r, and K_{cv} are positive gains. The closest obstacle, goal, and vehicle positions and the potential gradient are updated every 220 ms.

The stability of (36) can also be proven using the same decentralized control techniques discussed in the previous section. To simplify the problem, assume that the x and y position can be controlled independently, and assume that vehicle dynamics of the two closest vehicles are

$$\mathbf{S}_i : \dot{x}_i = -K_p x_i + K_p u_i, \qquad i \in \{1, 2\}. \tag{37}$$

If $K_a = K_r$ and $x_a - x_r = 0$, then the control can be written as

$$u_i = \gamma (x_i - x_j), \qquad i \neq j \tag{38}$$

where $\gamma = K_{v2}$. The resulting stability test matrix is

$$w_{ij} = \begin{cases} K_p(1 - \gamma), & i = j \\ -K_p \gamma, & i \neq j \end{cases} \tag{39}$$

which is an M matrix if $\gamma < 1/2$. When $\gamma = 1/2$, the two vehicles stabilize at $(1/2)[x_1(0) - x_2(0)]$ and $-(1/2)[x_1(0) - x_2(0)]$ where $x_1(0)$ and $x_2(0)$ are the initial x positions of the two vehicles. When $\gamma < 1/2$, both vehicles asymptotically converge to the origin. In this particular application, we do not want the vehicles to converge to the origin, so we chose $\gamma > 1/2$, which pushes the vehicles away from each other until they are a desired distance apart, after which the repulsive term is disabled.

In Fig. 11, six RATLER vehicles were used to surround a facility. The vehicles were initially located in the lower left-hand corner of Fig. 11. This is also where the base station was placed. The facility to be surrounded was located approximately 200 m on the other side of a rough motocross course. It is important to note that the base station's coordinates were obtained directly from a registered aerial photograph and neither surveying nor GPS integration was used. This fact demonstrates the feasibility of a fast-response squad of mobile robots.

When specifying the vehicle paths, the operator can either draw the paths of individuals or groups of vehicles, or draw goal and exclusion regions to be used by all vehicles, or do both. Drawing goal and exclusion regions is a simple way of specifying paths for several vehicles at once. However, there are circumstances when we need to specify the path of individual vehicles, such as when creating a diversion. For the test, the operator drew several different paths (displayed as black lines in Fig. 11) toward the facility. Groups, or in this case, pairs of robots were assigned a single path. The paths were chosen to follow the motocross course so that deep ditches and heavy brush could be avoided. However, these predefined paths ended short of the facility.

Goal and exclusion regions were used to specify the remaining vehicle path to the facility. The operator defined the goals (displayed as green lines) and the exclusion zones (displayed as red lines) on the GIS map. In Fig. 11, a goal line is drawn on the backside of the facility. Two different predefined paths terminate near this surrounding goal. Each of these paths was assigned two robots. Therefore, four robots are expected to participate in the surround task. The remaining two robots were assigned a path that goes near the main entrance of the facility. The nearest goal at the end of this path is inside the main entrance to the facility. These two vehicles were intended to act as a diversion, while the first four vehicles were strategically positioned to watch the rear door.

Once the mission is fully defined, the operator at the base station can view a simulation. In the simulation, the vehicles first follow their predefined paths. Once they reach the end, the potential field algorithm is used to plan the remaining path to the goal. This simulation is important since the potential field approach to path planning can be trapped by local minimum. After the operator previews the plan, the predefined paths, and the exclusion zone and goal polygons are downloaded to the vehicles. On board the vehicle, the same potential field-path planner directs the vehicle to the goal region while avoiding the exclusion regions, neighboring vehicles, and live obstacles. The true position of neighboring vehicles is obtained from the RF network, as each vehicle continually broadcasts its location and status. The vehicles naturally spread out along the goal region because of the repulsive forces between vehicles.

While the test was being performed, the vehicles were mostly able to stay on the motocross course using differentially corrected GPS. The aerial photograph is known to be optically warped, which means that it cannot be calibrated accurately for all regions. However, since the differential transmitter was initialized based on the coordinates taken from the map, the calibration is locally very good. When the vehicles strayed from their course, they ran into obstacles. On-board tilt sensors combined with a simple obstacle recovery algorithm allowed for all but one of the robots to successfully navigate the motocross course. The vehicle that failed to reach the goal was intended to be part of the diversion.

In the test, four vehicles reached the surrounding goal and spread out evenly along this partial perimeter (see Fig. 12). One vehicle entered the front gate. It took about one-half hour to set up, transporting the vehicles and initializing the differential station. It took another one-half hour to execute, including drawing

Fig. 12. Four RATLER vehicles surrounding the backside of the facility.

the paths, goals, and obstacles, downloading the information to the robots, and executing the mission.

VIII. CONCLUSION

In this paper, decentralized control theory is applied to the control of multiple cooperative mobile robotic vehicles. We mathematically described how to determine if a cooperative system is input/output reachable, structurally controllable and observable, and connectively stable. We illustrated the use of these techniques on two simple problems and we showed how these simple examples are applicable to multirobot formation control, perimeter surveillance, and surround problems. The stability analysis was used to determine limits on system parameters such as the interaction gain between vehicles, on the responsiveness of the vehicles, and on the sampling period for communication and position feedback, and to see how these limits vary as a function of the number of vehicles.

APPENDIX

In this appendix, we describe the control method used to drive each RATLER vehicle to a desired position. The RATLER vehicle is modeled as a skid-driven system, since the wheels on each side of the body are controlled with the same inputs. The typical nonholonomic problem (controlling three degrees of freedom with only two control inputs) is transformed into a holonomic problem by only controlling the position of a point p in front of the middle of the vehicle and leaving the orientation unconstrained (see Fig. 13).

When controlling the RATLER vehicle, the inverse Jacobian control law described below is applied in a feedback loop that is updated every 10 ms. This lower level control loop linearizes the vehicle's x, y response to the desired x, y commands from the higher level multivehicle control modeled in the previous sections. Being a skid-driven vehicle with a short wheel base, the RATLER vehicles can turn quickly, and the transient response of turning is negligible compared to the transient response of moving in the x, y directions. At the communication sampling rate of 1.1 s for four vehicles, the lower level control loop makes the vehicle's response in x and y position appear as identical independently driven values.

The control law is also convergent to the goal position as long as the estimate of the angle to the goal is within $\pm 90°$ of the
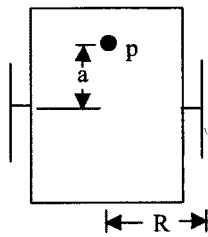
Fig. 13.   Schematic of vehicle.

actual angle. This can be shown by considering a linear perturbation of nonlinear dynamics of the vehicle.

$$\dot{x} = f(x, u)$$
$$\approx f(x_o, u_o) + \frac{\partial f}{\partial x}\bigg|_{x_o, u_o} (x - x_o) + \frac{\partial f}{\partial u}\bigg|_{x_o, u_o} (u - u_o)$$
$$\text{(A1)}$$

where $x \in \Re^3$ is the (x, y) position of the point p on the vehicle and orientation θ, $u \in \Re^2$ are the commanded right and left linear wheel velocities, $f(x, u)$ are the first-order vehicle dynamics, and x_o and u_o are linearized operating points. This can be rewritten as

$$\Delta \dot{x} = \frac{\partial f}{\partial x}\bigg|_{x_o, u_o} \Delta x + \frac{\partial f}{\partial u}\bigg|_{x_o, u_o} \Delta u \qquad \text{(A2)}$$

where

$$\Delta \dot{x} = f(x, u) - f(x_o, u_o)$$
$$\Delta x = x - x_o$$
$$\Delta u = u - u_o.$$

The first-order model of a skid-driven vehicle $\dot{x} = f(x, u)$ is

$$\begin{bmatrix} \dot{x}_p \\ \dot{y}_p \\ \dot{\theta} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \cos\theta - \frac{a}{R}\sin\theta & \cos\theta + \frac{a}{R}\sin\theta \\ \sin\theta + \frac{a}{R}\cos\theta & \sin\theta - \frac{a}{R}\cos\theta \\ \frac{1}{R} & \frac{-1}{R} \end{bmatrix} \begin{bmatrix} u_r \\ u_l \end{bmatrix}$$
$$\text{(A3)}$$

or

$$\dot{x} = B(x)u$$

where R is one-half the wheel base, a is the distance between the vehicle center and point p, and θ is the orientation of the vehicle. If $u_o = 0$, then $\partial f / \partial x|_{x_o, u_o} = 0$ and $\Delta u = u$. Since $\partial f / \partial u|_{x_o, u_o} = B(x_o)$, then

$$\Delta \dot{x} = B(x) \Delta u. \qquad \text{(A4)}$$

We choose the control to be a weighted inverse Jacobian, which is a function of the estimated state $\hat{x}$. Then

$$\Delta u = -[WB(\hat{x})]^{-1} \Delta p \qquad \text{(A5)}$$

where

$$W = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$
$$p = \begin{bmatrix} x_p \\ y_p \end{bmatrix}$$
$$\Delta p = p - p_o$$

and p_o is a linearized operating point. The matrix W is chosen to drive x_p, y_p to the desired reference position, yet leave θ unconstrained. Considering only the position of the vehicle

$$\Delta \dot{p} = -WB(x_o) [WB(\hat{x})]^{-1} \Delta p \qquad \text{(A6)}$$

or

$$\left\lfloor sI + WB(x_o) [WB(\hat{x})]^{-1} \right\rfloor \Delta p = 0. \qquad \text{(A7)}$$

For $\Delta p \to 0$, $WB(x_o)[WB(\hat{x})]^{-1}$ must be positive definite [43]. It can be shown that for the skid-driven dynamics in (A3), this occurs if and only if $-90° < \theta_o - \hat{\theta} < 90°$.

ACKNOWLEDGMENT

REFERENCES

[1] K. Kosuge, T. Oosumi, M. Satou, K. Chiba, and K. Takeo, "Transporation of a single object by two decentralized-controlled nonholonomic mobile robots," in *Proc. Conf. Robotics and Automation*, Leuven, Belgium, May 1998, pp. 2989–2994.

[2] H. Yamaguchi, "A cooperative hunting behavior by mobile robot troops," in *Proc. Conf. Robotics and Automation*, Leuven, Belgium, May 1998, pp. 3204–3209.

[3] F. R. Noreils, "Multi-robot coordination for battlefield strategies," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Raleigh, NC, July 1992, pp. 1777–1784.

[4] D. F. Hougen, M. D. Erickson, P. E. Rybski, S. A. Stoeter, M. Gini, and N. Papanikolopoulos, "Autonomous mobile robots and distributed exploratory missions," in *Distributed Autonomous Robotic Systems 4*, L. E. Parker, G. Bekey, and J. Barhen, Eds.   New York: Springer-Verlag, 2000, pp. 221–230.

[5] B. Brumitt and M. Hebert, "Experiments in autonomous driving with concurrent goals and multiple vehicles," in *Proc. Int. Conf. Robotics and Automation*, Leuven, Belgium, May 1998, pp. 1895–1902.

[6] T. Kaga, J. Starke, P. Molnar, M. Schanz, and T. Fukuda, "Dynamic robot-target assignment—Dependence of recovering from breakdowns on the speed of the selection process," in *Distributed Autonomous Robotic Systems 4*, L. E. Parker, G. Bekey, and J. Barhen, Eds.   New York: Springer-Verlag, 2000, pp. 325–334.

[7] J. E. Hurtado, R. D. Robinett, C. R. Dohrmann, and S. Y. Goldsmith, "Distributed sensing and cooperating control for swarms of robotic vehicles," in *Proc. IASTED Conf. Control and Applications*, Honolulu, HI, Aug. 12–14, 1998, pp. 175–178.

[8] J. S. Jennings, G. Whelan, and W. F. Evans, "Cooperative search and rescue with a team of mobile robots," in *Proc. IEEE Int. Conf. Advanced Robotics*, Monterey, CA, 1997, pp. 193–200.

[9] S. Goldsmith, J. Feddema, and R. Robinett, "Analysis of decentralized variable structure control for collective search by mobile robots," in *Proc. SPIE'98, Sensor Fusion and Decentralized Control in Robotic Systems*, Boston, MA, Nov. 1–6, 1998, pp. 40–47.

[10] Y. U. Cao, A. S. Fukunaga, and A. B. Kahng, "Cooperative mobile robotics: Antecedents and directions," in *Proc. 1995 IEEE/RSJ IROS Conf.*, pp. 226–234.

[11] L. E. Parker, "Current state of the art in distributed autonomous mobile robotics," in *Distributed Autonomous Robotic Systems 4*, L. E. Parker, G. Bekey, and J. Barhen, Eds.   New York: Springer-Verlag, 2000, pp. 3–12.

[12] R. C. Arkin, "Cooperation without communication: Multiagent schema-based robot navigation," *J. Robot. Syst.*, vol. 9, no. 3, pp. 351–364, 1992.

[13] T. Balch and R. C. Arkin, "Behavior-based formation control for multi-robot teams," *IEEE Trans. Robot. Automat.*, vol. 14, pp. 926–939, Dec. 1998.

[14] R. C. Kube and H. Zhang, "Collective robotics: From social insects to robots," *Adaptive Behavior*, vol. 2, no. 2, pp. 189–218, Fall 1993.

[15] R. A. Brooks and A. M. Flynn, "Fast, cheap and out of control: A robot invasion of the solar system," *J. Br. Interplanet. Soc.*, vol. 42, pp. 478–485, 1989.

[16] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE J. Robot. Automat.*, vol. RA-2, pp. 14–23, Mar. 1986.

[17] T. Fukuda, H. Mizoguchi, K. Sekiyama, and F. Arai, "Group behavior control for MARS (micro autonomous robotic system)," in *Proc. Int. Conf. Robotics and Automation*, Detroit, MI, May 1999, pp. 1550–1555.

[18] F. R. Noreils, "Toward a robot architecture integrating cooperation between mobile robots: Application to indoor environment," *Int. J. Robot. Res.*, vol. 12, no. 1, pp. 79–98, Feb. 1993.

[19] Q. Chen and J. Y. S. Luh, "Coordination and control of a group of small mobile robots," in *Proc. IEEE Int. Conf. Robotics and Automation*, vol. 3, 1994, pp. 2315–2320.

[20] H. Yamaguchi and T. Arai, "Distributed and autonomous control method for generating shape of multiple mobile robot group," in *Proc. IEEE Int. Conf. Intelligent Robots and Systems*, vol. 2, 1994, pp. 800–807.

[21] H. Yamaguchi and J. W. Burdick, "Asymptotic stabilization of multiple nonholonomic mobile robots forming group formations," in *Proc. Conf. Robotics and Automation*, Leuven, Belgium, May 1998, pp. 3573–3580.

[22] E. Yoshida, T. Arai, J. Ota, and T. Miki, "Effect of grouping in local communication system of multiple mobile robots," in *Proc. IEEE Int. Conf. Intelligent Robots and Systems*, vol. 2, 1994, pp. 808–815.

[23] P. Molnar and J. Starke, "Communication fault tolerance in distributed robotic systems," in *Distributed Autonomous Robotic Systems 4*, L. E. Parker, G. Bekey, and J. Barhen, Eds. New York: Springer-Verlag, 2000, pp. 99–108.

[24] F. E. Schneider, D. Wildermuth, and H.-L. Wolf, "Motion coordination in formations of multiple robots using a potential field approach," in *Distributed Autonomous Robotic Systems 4*, L. E. Parker, G. Bekey, and J. Barhen, Eds. New York: Springer-Verlag, 2000, pp. 305–314.

[25] G. Beni and P. Liang, "Pattern reconfiguration in swarms—Convergence of a distributed asynchronous and bounded iterative algorithm," *IEEE Trans. Robot. Automat.*, vol. 12, pp. 485–490, June 1996.

[26] Y. Liu, K. Passino, and M. Polycarpou, "Stability analysis of one-dimensional asynchronous swarms," in *Proc. American Control Conf.*, Arlington, VA, June 25–27, 2001, pp. 716–721.

[27] A. Winfield, "Distributed sensing and data collection via broken ad hoc wireless connected networks of mobile robots," in *Distributed Autonomous Robotic Systems 4*, L. E. Parker, G. Bekey, and J. Barhen, Eds. New York: Springer-Verlag, 2000, pp. 273–282.

[28] J. P. Desai, J. Ostrowski, and V. Kumar, "Controlling formations of multiple mobile robots," in *Proc. Conf. Robotics and Automation*, Leuven, Belgium, May 1998, pp. 2864–2869.

[29] J. P. Desai, V. Kumar, and J. P. Ostrowski, "Modeling and control of formations of nonholonomic mobile robots," *IEEE Trans. Robot. Automat.*, vol. 17, pp. 905–908, Dec. 2001.

[30] T. Fukuda *et al.*, "Evaluation on flexibility of swarm intelligent system," in *Proc. Conf. Robotics and Automation*, Leuven, Belgium, May 1998, pp. 3210–3215.

[31] B. Brummitt and A. Stentz, "GRAMMPS: A generalized mission planner for multiple mobile robots in unstructured environments," in *Proc. Conf. Robotics and Automation*, Leuven, Belgium, May 1998, pp. 1564–1571.

[32] O. Simonin, A. Liegeois, and P. Rongier, "An architecture for reactive cooperation of mobile distributed robots," in *Distributed Autonomous Robotic Systems 4*, L. E. Parker, G. Bekey, and J. Barhen, Eds. New York: Springer-Verlag, 2000, pp. 35–44.

[33] S. I. Roumeliotis and G. A. Bekey, "Distributed multi-robot localization," in *Distributed Autonomous Robotic Systems 4*, L. E. Parker, G. Bekey, and J. Barhen, Eds. New York: Springer-Verlag, 2000, pp. 179–188.

[34] M. Bozorg, E. M. Nebot, and H. F. Durrant-Whyte, "A decentralised navigation architecture," in *Proc. Int. Conf. Robotics and Automation*, Leuven, Belgium, May 1998, pp. 3413–3418.

[35] E. Uchibe, M. Asada, and K. Hosoda, "Cooperative behavior acquisition in multi mobile robots environment by reinforcement learning based on state vector estimation," in *Proc. Int. Conf. Robotics and Automation*, Leuven, Belgium, May 1998, pp. 1558–1563.

[36] D. D. Siljak, *Decentralized Control of Complex Systems*. New York: Academic, 1991.

[37] M. E. Sezer and D. D. Siljak, "Robust stability of discrete systems," *Int. J. Contr.*, vol. 48, no. 5, pp. 2055–2063, 1988.

[38] G. D. Smith, *Numerical Solution of Partial Differential Equations: Finite Difference Methods*, 3rd ed. London, U.K.: Oxford Univ. Press, 1985.

[39] C. Lewis, J. T. Feddema, and P. Klarer, "Robotic perimeter detection system," in *Proc. SPIE*, vol. 3577, Boston, MA, Nov. 3–5, 1998, pp. 14–21.

[40] J. Feddema, C. Lewis, and P. Klarer, "Cooperative robotic sentry vehicles," in *Proc. SPIE*, vol. 3839, Boston, MA, Sept. 19–20, 1999, pp. 44–54.

[41] J. Feddema, C. Lewis, and R. LaFarge, "Cooperative sentry vehicles and differential GPS leapfrog," in *Distributed Autonomous Robotic Systems 4*, L. E. Parker, G. Bekey, and J. Barhen, Eds. New York: Springer-Verlag, 2000, pp. 293–302.

[42] J. T. Feddema, D. A. Schoenwald, and F. J. Oppel, "Decentralized control of a collective of autonomous robotic vehicles," in *Proc. American Control Conf.*, Arlington, VA, June 25–27, 2001, pp. 2087–2092.

[43] C. Samson, M. Le Borgne, and B. Espiau, *Robot Control: The Task Function Approach*. Oxford, U. K.: Clarendon, 1991, pp. 218–264.

[44] J. T. Feddema and D. A. Schoenwald, "Stability analysis of decentralized cooperative controls," in *Multi-Robot Systems: From Swarms to Intelligent Automata*, A. C. Shultz and L. E. Parker, Eds. Norwell, MA: Kluwer, 2002.
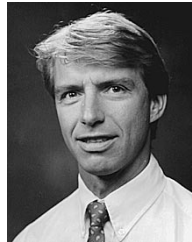
John T. Feddema (M'86) received the B.S. degree from Iowa State University, Ames, in 1984, and the M.S. and Ph.D. degrees from Purdue Unversity, West Lafayette, IN, in 1986 and 1989, respectively, all in electrical engineering.

He is currently a Distinguished Member of Technical Staff in the Intelligent System Sensors and Controls Department, Sandia National Laboratories, Albuquerque, NM. He has performed research on a wide range of robotics projects, including multifingered hands, visual servoing, flexible robot arm control, whole arm obstacle avoidance, sensor fusion, microsurgery and microassembly robotics, and decentralized control of cooperative robotic vehicles. He has over 60 publications in journals, conference proceedings, and book chapters, and he is a co-author of the book, "Flexible Robot Dynamics and Controls" (New York: Kluwer/Plenum, 2002). He holds two patents in sway control of cranes and one patent on a hopping robot.

Dr. Feddema is a member of the IEEE Robotics and Automation Society, and is on the editors' board for the *Journal of Micromechatronics*.

Chris Lewis (S'94–M'95) received the B.S. degree in mechanical engineering from Kansas State University, Manhattan, in 1987, and the M.S. and Ph.D. degrees from the School of Electrical Engineering, Purdue University, West Lafayette, IN, in 1990 and 1994, respectively.

He is currently a Principal Member of Technical Staff with Sandia National Laboratories, Albuquerque, NM, where he conducts research and development for robotic systems.

David A. Schoenwald (S'84–M'86) received the B.S. degree from the University of Iowa, Iowa City, in 1986, the M.S. degree from the University of Illinois, Urbana-Champaign, in 1988, and the Ph.D. degree from Ohio State University, Columbus, in 1992, all in electrical engineering.

Since April 1999, he has been a Principal Member of Technical Staff in the Intelligent Systems and Robotics Center, Sandia National Laboratories, Albuquerque, NM. His current work centers on agent-based simulation and control, economic modeling of critical infrastructures, decentralized control of robotic swarms, and nonlinear control systems. From 1992 to 1999, he was a Development Staff Member, Instrumentation and Controls Division, Oak Ridge National Laboratory, Oak Ridge, TN. He was also an Adjunct Assistant Professor with the University of Tennessee, Electrical Engineering Department. He was an Associate Editor for IEEE *Control Systems Magazine* from 1996 to 2000. He has served as a Chair for several industry conferences, and has served on several conference program committees.

Dr. Schoenwald received the Martin Marietta Technical Achievement Award in 1995 and the Industrial Computing Society Courageous User Award in 1996, both for his work on a direct-drive industrial sewing machine. He has served as an Associate Editor on the IEEE Control Systems Society Conference Editorial Board since 1993, and as an Associate Editor for the IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY since 2000. He was an Associate Editor for IEEE CONTROL SYSTEMS MAGAZINE from 1996 to 2000. He is currently the Registration Chair for the 2002 IEEE Conference on Decision and Control. Previously, he was the Finance Chair for the 1999 American Control Conference, and the Publicity Chair for the 1998 IEEE Conference on Decision and Control. He is a member of Tau Beta Phi and Eta Kappa Nu.