2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

Decentralized H_∞ Filtering in a Multi-Agent System

Thomas R. Nelson and Randy A. Freeman

Abstract— **This paper develops a decentralized H_∞ filtering algorithm for a mobile sensor network. Each agent maintains an estimate of the target and moves to improve the information from its sensor. Simulation results compare the performance of the multi-agent system to a system using only local estimates when the network is tracking an evasive target.**

I. Introduction

The area of mobile multi-agent systems has received considerable attention recently. These multi-agent systems are able to perform a variety of tasks such as environmental monitoring [9], [14], [17], search and rescue operations [7], [8], target tracking [10], [11], [21], and formation and coverage control [2], [6], [20].

This paper considers the problem of estimation and control for multiple mobile agents with limited sensing, computation, communication and motion capabilities. Previous work in the area of target tracking in sensor networks focused primarily on Kalman filtering for both fixed sensors [1], [12] and mobile sensors [5], [20]. One of the underlying assumptions of the Kalman filter is that both the process and measurement noises behave like random signals. An alternative is to design a filter based on limited knowledge of the noise behavior. The H_∞, or minimax, filter approach minimizes the worst-case error variance over all admissible l_2 energy signals. The H_∞ filter [4], [16] has not received as much attention as the Kalman filter, and few have tried to implement this filter in a decentralized fashion [19].

More recent decentralized Kalman filter techniques using consensus estimators have been proposed [11], [20]. The use of consensus estimators are of great interest because of the similarities in structure between the H_∞ and Kalman filters. The decentralized H_∞ filter algorithm presented in this paper borrows from these techniques. In addition, a decentralized control law is derived. This control law will minimize a cost function in order to improve the overall sensing quality. This paper investigates the performance of this mobile sensor network estimating the position of a target.

Several different motion models will be used for the target tracking problem. In the first, the target will be driven by a random signal. The second will assume the target will have some information about the location of each agent and the sensor model used. In motivating the set-valued estimator, Schweppe [13] proposed an idea in which the target moves intentionally to evade tracking. This paper will present a target motion model which moves to increase uncertainty based on limited knowledge of the sensor network.

The main contributions in this paper include a novel approach to decentralized H_∞ filtering and a decentralized motion control law used to improve measurements and limit filter gains. It also discusses a evasive target model based on information about the sensor model.

The organization of the paper is as follows. Section II formulates the problem of a mobile multi-agent group estimating a dynamic system's state with limited information about the process noise. The example of target tracking will be used throughout the paper. Section III proposes a novel method for decentralized H_∞ filtering. Section IV investigates a decentralized motion control law in an effort to improve estimates and reduce filter gains. Evasive target design is discussed in section V. The results from the decentralized estimation and control are presented in section VI. Section VII discusses the results and outlines future work.

II. Problem Statement

Suppose there are N mobile sensors and a target moving in a plane with positions $p_1, p_2, ..., p_N \in \mathbb{R}^2$ and $x_t \in \mathbb{R}^2$, respectively. The sensors estimate the position of the target whose dynamics are described by

$$\begin{aligned} x_{k+1} &= A_k x_k + B_k w_k \\ z_k &= L_k x_k \end{aligned} \tag{1}$$

where $x_k \in \mathbb{R}^2$ is the state vector, and the z_k term is a linear combination of target states which will be discussed in the next section. The noise driving this system, $w_k \in \mathbb{R}^2$, is assumed to be an l_2 signal. Measurements of the target position are given by:

$$y_{i,k} = x_k + m_{i,k} \tag{2}$$

where $y_{i,k} \in \mathbb{R}^2$ is the measurement reading, and $m_{i,k} \in \mathbb{R}^2$ is the measurement uncertainty for agent i. There are several different models that can be used, but in this paper, the range-bearing model will be used. The level of uncertainty for agent i is dependent on the distance and angle between the target and sensor, $r_{i,k} = |x_{i,k} - p_{i,k}|$ and $\theta_{i,k} = \angle(x_{i,k} - p_{i,k})$. The measurement noise is described by,

$$m_i = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \end{bmatrix} \begin{bmatrix} \sqrt{f_r(r_i)} & 0 \\ 0 & \sqrt{f_b(r_i)} \end{bmatrix} \tag{3}$$

where f_r and f_b represent the range and bearing noise variances described by the convex functions,

This work was supported in part by NSF grant ECS-0601661.

Thomas Nelson and Randy Freeman are with the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL, 60208-3118, USA thomas.nelson@u.northwestern.edu, freeman@eecs.northwestern.edu

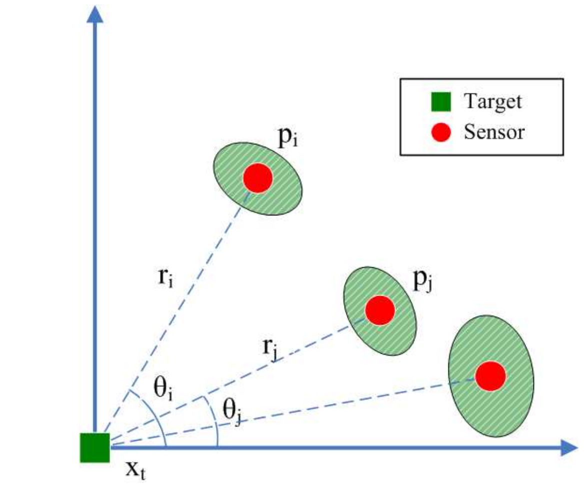

Fig. 1. Measurement uncertainties ellipses for three sensors. The ellipses are placed at sensor locations for ease of viewing. It illustrates the distance dependent nature of this uncertainty.

$$
\begin{aligned}
f_r(r_i) &= a_2(r_i - a_1)^2 + a_0 \\
f_b(r_i) &= \alpha f_r(r_i)
\end{aligned} \tag{4}
$$

where a_0, a_1, a_2, α are model parameters. Each sensor has a "sweet spot" when $r_i = a_1$ and the size of the measurement uncertainty is at its minimum values. Figure 1 illustrates how the size of the measurement uncertainty changes as a function of distance.

The H_∞ filtering scheme and motion controller each use the sum of information from each individual sensor in the calculation of the global variables Σ_{global}, Σ_i^r, and Σ_i^θ. Like [20], this is a motivating factor for the use of a PI dynamic consensus estimator. A discrete time version of the dynamic consensus estimator presented in [20] will be used to estimate these global variables. Each agent has an input $u_i \in \mathbb{R}^{k \times r}$, internal states $v_i, w_i \in \mathbb{R}^{k \times r}$, and an output $y_i = v_i$. The discrete time version of the PI estimator is given by the following equations,

$$
\begin{aligned}
v_i(k+1) &= (1-\gamma h)v_i(k) - hK_p \sum_{j \in N_i(k)} [v_i(k) - v_j(k)] \\
&\quad + hK_i \sum_{j \in N_i(k)} [w_i(k) - w_j(k)] + \gamma h u_i(k) \\
w_i(k+1) &= w_i(k) - hK_i \sum_{j \in N_i(k)} [v_i(k) - v_j(k)]
\end{aligned} \tag{5}
$$

Here $\gamma > 0$ is a design parameter, N_i contains the set of all one-hop neighbors of agent i, h is the sample time, and K_p, K_i are the proportional and integral gains, respectively. This discrete-time PI consensus estimator is stable as long as the sample time, h, is small. The consensus estimator will track the global signal $\frac{1}{N}\sum_{i=1}^N u_i$ with zero steady-state error when the inputs are constant, and there is a constant connected network topology.

III. H_∞ Estimator

The group of mobile sensors estimate the location of a target using H_∞ filters. The target dynamics are represented by (1). The centralized version of this filter can be written as [15],

$$
\begin{aligned}
\hat{x}_k &= A\hat{x}_{k-1} + K_k(y_k - CA\hat{x}_{k-1}) \\
\Delta_k &= A\Sigma_{k-1}A^T + BW_{k-1}B^T \\
K_k &= \Delta_k(I + C^T M_k^{-1} C\Delta_k)^{-1} C^T M_k^{-1} \\
\Sigma_k &= \Delta_k(I - \gamma^{-2} L_k^T Q_k L_k \Delta_k + C^T M_k^{-1} C \Delta_k)^{-1} \\
\hat{z}_k &= L_k \hat{x}_k
\end{aligned} \tag{6}
$$

where K_k is the filter gain, and P_k, M_k and W_k are analogous to the error, measurement and process covariances of the Kalman filter. The noise attenuation level, γ, and the weighting matrix, L_k are user defined values.

The structure of the H_∞ filter is similar to the Kalman filter structure, but z_k, a linear combination of x_k, is being estimated. This gives the H_∞ filter a directional property. It allows the designer added flexibility in finding the optimal performance along a specified direction in the state space [18]. The directional property appears in the modified Riccati equation and its dependence on L_k; there is no analog with the conventional Kalman filter. As γ tends to infinity, the H_∞ filter reduces to the Kalman filter. It should also be noted that the filter will be unstable if the γ value is chosen to be smaller than the threshold γ_0, the minimum attenuation level. If the information from the N agents are represented as, $y^T = [y_1^T, y_2^T, ..., y_N^T]$, $C = [C_1 C_2 ... C_N]^T$ and $M = \text{diag}(M_1, M_2, ..., M_N)$ for the centralized filter then the following simplifications can be made $C^T M_k^{-1} C = \sum_{i=1}^N C_i^T M_{k,i}^{-1} C_i$ and $C^T M_k^{-1} y_k = \sum_{i=1}^N C_i^T M_{k,i}^{-1} y_{k,i}$.

Given that each agent has limited communication, assume it can communicate with at least one neighbor (i.e. the network is connected). Each agent runs PI consensus estimators to approximate the average fused covariance matrix

$$
\tilde{C} \approx \frac{1}{N} \sum_{i=1}^N C_{k,i}^T M_{k,i}^{-1} C_{k,i} \tag{7}
$$

and the average fused measurement

$$
\tilde{y} \approx \frac{1}{N} \sum_{i=1}^N C_{k,i}^T M_{k,i}^{-1} y_{k,i} \tag{8}
$$

where $M_{k,i}$, $C_{k,i}$ and $y_{k,i}$ are dependent on the i^{th} sensor, and N is the total number of sensors in the system. The output of the consensus estimators are fed into the H_∞ filter resulting in the equations:

$$
\begin{aligned}
\hat{x}_k &= A\hat{x}_{k-1} + N\tilde{K}_k(\tilde{y_k} - \tilde{C}A\hat{x}_{k-1}) \\
\Delta_k &= A\Sigma_{k-1}A^T + BW_{k-1}B^T \\
\tilde{K_k} &= \Delta_k(I + N\tilde{C}\Delta_k)^{-1} \\
\Sigma_k &= \Delta_k(I - \gamma^{-2} L_k^T Q_k L_k \Delta_k + N\tilde{C}\Delta_k)^{-1} \\
\hat{z}_k &= L_k \hat{x}_k
\end{aligned} \tag{9}
$$

where the approximate fused covariance matrix and fused measurement replace the actual variables allowing for decentralized filtering.

IV. Motion Control Law

To capitalize on its limited motion, each agent implements a motion controller in order to improve the overall sensing quality of the system. Given the limited communication capacity of each agent, the controller needs to be decentralized.

We could design a central (unimplementable) control law based on a global cost function, and replace the unknown global values with local estimates [20]. The cost function, J, is defined as the trace of Σ_{global}:

$$J = \text{tr}\left[\Sigma_{\text{global}}\right] \tag{10}$$

Σ_{global} is the solution to the Riccati equation from the centralized H_∞ filter:

$$\Sigma_{global,k} = \Delta_k(I + C^T M^{-1} C\Delta_k - \gamma^{-2}\bar{Q}_k\Delta_k)^{-1} \tag{11}$$

We want to take the gradient with respect to each sensor's state to determine the best path.

Suppose all the agents are kinematic, and fully actuated so that $p_k = p_{k-1} + u_i$; our initial control law follows [20]

$$u_i = -\Gamma T_i^T \begin{bmatrix} \frac{\partial J}{\partial r_i} \\ \\ \frac{1}{r_i}\frac{\partial J}{\partial \theta_i} \end{bmatrix} \tag{12}$$

where $\Gamma > 0$ is a gain matrix and T_i is the rotation matrix for agent i. The derivatives are written as

$$\begin{aligned} \frac{\partial J}{\partial r_i} &= \text{tr}\left[\Sigma_i^r\right] \\ \frac{\partial J}{\partial \theta_i} &= \text{tr}\left[\Sigma_i^\theta\right] \end{aligned} \tag{13}$$

with Σ_i^r and Σ_i^θ representing the derivatives of the global covariance with respect to r_i and θ_i, respectively.

Proposition 1. *(Motion Control Law)*
Take the gradients of the cost function J with respect to the sensor's coordinates (z_i represents either r_i or θ_i). The gradient controller can be written in closed form solution as:

$$\Delta_k^z = A\Sigma_{k-1}^z A^T \tag{14}$$

$$\begin{aligned} \Sigma_{i,k}^z = {} & \Delta_k^z\Omega^{-1} + \Delta_k\Omega^{-1}C^T\frac{\partial}{\partial z_i}(M^{-1})C\Delta_k\Omega^{-1} \\ & - \Delta_k\Omega^{-1}(C^T M^{-1}C\Delta_k^z - \gamma^{-2}L^T Q L\Delta_k^z)\Omega^{-1} \end{aligned} \tag{15}$$

with

$$\Omega = I - \gamma^{-2}L^T Q_k L\Delta_k - C^T M_k^{-1} C\Delta_k \tag{16}$$

Based on our sensor-bearing model, it follows

$$\begin{aligned} \frac{\partial}{\partial r_i}V_k^{-1} &= -2a_2(r_{i,k} - a_1)T_{i,k}R_{i,k}^{-2}\begin{bmatrix} 1 & 0 \\ 0 & \alpha \end{bmatrix} T_{i,k}^T \\ \frac{\partial}{\partial \theta_i}V_k^{-1} &= V_k^{-1}(\Psi_{i,k} + \Psi_{i,k}^T)V_k^{-1} \end{aligned} \tag{17}$$

with

$$\Psi_{i,k} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} T_{i,k}R_{i,k}^{-1}T_{i,k}^T \tag{18}$$

A decentralized version of this control can be implemented with some minor changes. Each agent keeps a local copy of the gradients $\Sigma_{i,k}^r$ and $\Sigma_{i,k}^\theta$. The global Δ_k and Ω_k variables are replaced by the equivalent local filter values. The use of consensus estimators allow each agent to replace M^{-1} with its local approximation, $\tilde{C}$.

V. Evasive Target Design

In the application of target tracking, assumptions are made about the signal driving target. For simplicity, the signal is often assumed to be white noise. How does the filter perform if the target is purposely performing evasive maneuvers, as suggested in [13]? The target requires information about the sensor model and possibly the filter used by each agent for it to move in an effective manner. Assuming the target is capable of determining the distance between itself and each agent, in addition to knowledge of the sensor model, it is able to construct a possible uncertainty ellipses $M_{i,k}$ for each agent i. Let $Q_{\texttt{fused}}$ represent the fused target uncertainty ellipse based on all of the sensor measurement ellipses. The calculation of $Q_{\texttt{fused}}$ will be discussed later in this section.

There are several different ways the target can behave. The target can behave in a greedy manner and try to maximize trace of the overall measurement set, i.e.,

$$\max \text{tr}[Q(k)] \tag{19}$$

at every time step, as discussed previously, or it can try to maximize the time before capture, when the trace of the uncertainty set drops below a certain threshold level:

$$\max\left\{T : \text{tr}[Q(k)] \geq \gamma,\ t_0 \leq t \leq t_0 + T\right\} \tag{20}$$

The level of evader intelligence is also an aspect to consider. In this example, the target is only aware of the sensor models used but is unaware of the filtering techniques used. The target motion problem can be formulated as an optimization problem searching for the minimum trace (or volume) ellipsoid.

In this paper, the goal of the target is to maximize the uncertainty at every time step (19), so the cost function is,

$$J = \text{tr}\ Q_{\text{fused}} \tag{21}$$

the trace of a fused measurement covariance matrix. The target does not use the "one-shot" approach. It looks at the smallest over-bounding ellipse of the intersection of a series of measurement covariances (22).

$$Q_{\text{fused}} \supset \bigcap_{i=1}^{N} M_{i,k} \tag{22}$$

This over-bounding is accomplished as follows. Given a symmetric matrix $A > 0$, a vector b of same size and a scalar $\rho > 0$, define vector q and symmetric matrix $Q > 0$ as

$$\begin{aligned} Q &= \left[1 + b^T A^{-1} b - \rho\right] A^{-1} \\ q &= A^{-1} b \end{aligned} \tag{23}$$

$\mathcal{F}$ is defined as a mapping, $(q, Q) = \mathcal{F}(A, b, \rho)$, and $\mathcal{F}$ is smooth over this domain. Because the target has knowledge of the sensor model used and the distance between it and each agent, the target is able to calculate the corresponding uncertainty, $\mathcal{E}(x_t, M_j)$ for each agent. Given multiple uncertainty ellipses $\mathcal{E}(x_t, M_j)$, we define the smallest overbounding ellipse as:

$$(q, Q) = \mathcal{F}(\sum_j \lambda_j M_j^{-1}, \sum_j \lambda_j M_j^{-1} x_t, \sum_j \lambda_j x_t^T M_j^{-1} x_t) \tag{24}$$

where λ_j are nonnegative weights with $\sum_j \lambda_j = 1$.

VI. Simulations

This section will evaluate the performance of decentralized H_∞ filters for both stationary and mobile sensors. Two target models will be used, the random walk and the evasive motion technique discussed in the previous section. The communication links between agents are described below,

$$l_{i,j} = \begin{cases} 1, & |p_i - p_j| \le r \\ 0, & \text{else} \end{cases} \tag{25}$$

where r is the communication radius, p_i and p_j are the positions of sensor i and j and $l_{i,j}$ represents whether sensor i and sensor j can communicate with each other. The agents will use the sensing model from section II, eqs. (3), (4). The sensor model parameters are $a_0 = 0.3528$,$a_1 = 15.625$, $a_2 = 0.0008$ and $\alpha = 5$. The communication radius is set at $r = 20$ to guarantee the network is connected.

Figure 2 compares the performance of the decentralized and local H_∞ filters for one of the ten stationary agents tracking a target with white process noise. Each agent can communicate with at most two neighbors, and the process noise covariance, $W_k = 0.01I_2$, is known. Figure 3 shows the performance of the decentralized and local H_∞ filters for an agent, but the process noise covariance, $W_k = 0.001I_2$, has decreased. After comparing the performance of the decentralized filter between the figures 2 and 3, the performance of it improves as the speed of the target increases. The increase in performance from the decentralized filter is the result of the consensus estimator. At every time step, each estimator combines its local information with the information its neighbors sent last time step introducing lag into the system. As the process noise decreases, the information passed between agents is more relevant at the next time step and can be used to improve the estimates.

By introducing a motion controller with each agent, the agent should move in a direction to improve the performance of the sensor. Figure 4 shows the estimation error for a stationary agent using only local information for estimates and an agent with the motion controller starting from the same position using a decentralized estimator. The estimation

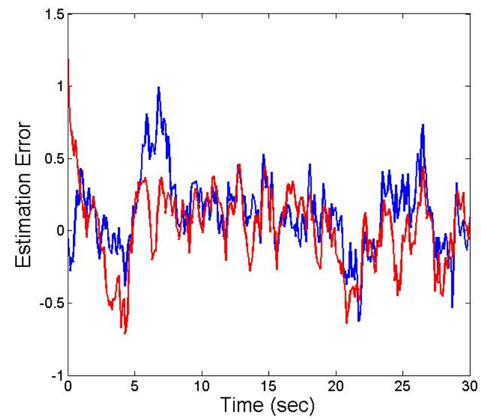

Fig. 2. Estimation error for the local (blue) and decentralized H_∞ (red) filters. The process noise of the target W_k = 0.01I

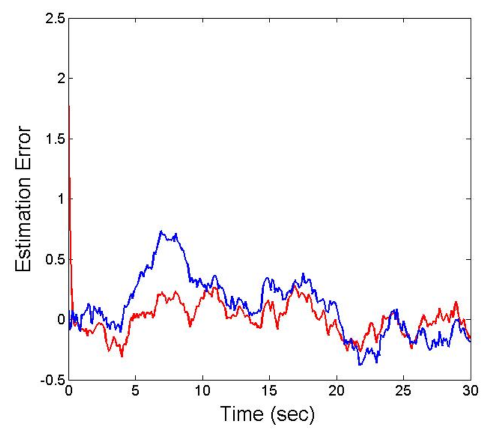

Fig. 3. Estimation error for the local (blue) and decentralized H_∞ (red) filters. The process noise covariance W_k = 0.001I

error for the mobile agent is smoother and approaches zero quicker than the stationary agent. Figure 5 compares the performance of a stationary agent and an agent with motion both using decentralized H_∞ estimation.

Figure 6 illustrates the motion for an evasive target and four sensors. The target uses the equation described in the previous section, eq. (19). The agents follow the control law derived in section IV. The process noise covariance of the target is described as $W_k = 0.01I_2$. All four sensors use the sensor model and parameters discussed in section II. The agents move to align themselves for better sensor readings as the target reacts to the sensor motion.

VII. Summary and Future Work

We present a framework for performing decentralized motion control and H_∞ estimation for a mobile multi-agent system. Because of the similarities in structure between the Kalman and H_∞ filters, we were able to utilize several of the techniques in [20] to perform decentralized H_∞ filtering. We were able to validate this algorithm with simulations. A decentralized gradient control law was also proposed in this paper. Simulations compared the sensing performance

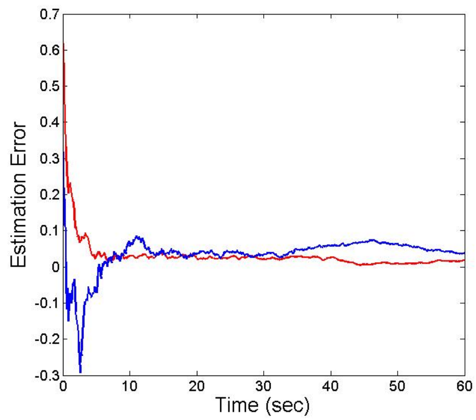

Fig. 4. Estimation error for a stationary (blue) sensor using only local H_∞ filter and a mobile (red) sensor using a decentralized H_∞ filter. The target is stationary in this example.

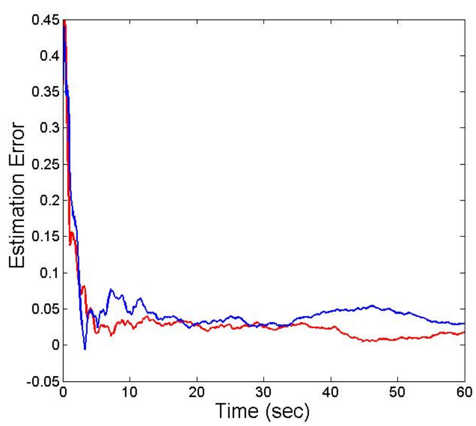

Fig. 5. Estimation error for a stationary (blue) and mobile (red) sensor using decentralized H_∞ filters. The target is stationary in this example.

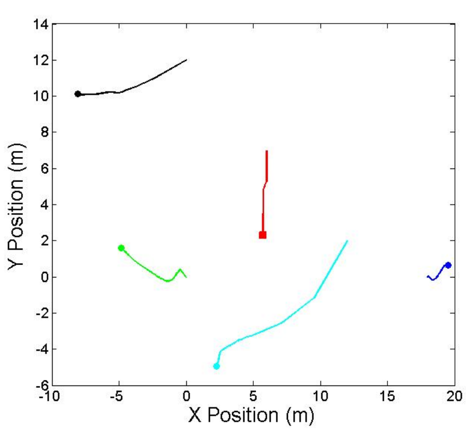

Fig. 6. Motion of the sensor network following an evasive target. The sensors are represented as circles and the target as a square. The tails show how the network and target evolve over time.

between a stationary and mobile sensor for a stationary target. This control law was also tested against a target performing evasive maneuvers.

Future work would include stability analysis for this coupled estimation and control problem, and the effect of time delays in the communication channels and the performance of the consensus filters. The premise of this work is that the H_∞ filter should outperform the Kalman filter when the target maneuvers to intentionally avoid tracking. A more thorough analysis is needed in regards to the performance of these two filters and whether there is a performance advantage with the H_∞ filter.

Acknowledgments

This work was supported in part by NSF grant ECS-0601661.

References

[1] Y. Bar-Shalom and X.R. Li, *Estimation and Tracking: Principles, Techniques, and Software.* Norwood, MA: Artech House, Inc, 1993.
[2] C. Belta and V. Kumar, "Abstraction and control for groups of robots," *IEEE Trans. Robotics*, vol. 20, no. 5, pp. 865 - 875, Oct. 2004.
[3] D. Bernstein, *Matrix Mathematics*. Princeton, NJ: Princeton University Press, 2005.
[4] D. Bertsekas and I. Rhodes, "Recursive state estimation for a set-membership description of uncertainty," *IEEE Trans. Automat. Contr.*, vol. AC- 16, no. 2, pp. 117 - 128, 1971.
[5] T. H. Chung, J. W. Burdick and R. M. Murray, "A Decentralized Motion Coordination Strategy for Dynamic Target Tracking," in *Proc. 2006 IEEE Intern. Conf. Robotics and Automation*, Orlando, FL, May 2006.
[6] J. Cortes, S. Martinez, T. Karatas and F. Bullo, " Coverage control for mobile sensing networks," *IEEE Trans. Robotics and Automation*, vol. 20, no. 2, pp. 243 - 255. 2004.
[7] J.S. Jennings, et. al., "Cooperative search and rescue with a team of mobile robots," *Proc. 8th Intern. Conf. on Advanced Robotics*, Monterrey, CA, 1997.
[8] H. Kitano, et. al., "RoboCup rescue: search and rescue in large-scale disasters as a domain for autonomous agents research," *Proc. of 1999 IEEE Conf. on Systems, Man and Cybern.*, Toyko, Japan, 1999.
[9] K.M. Lynch, I.B. Schwartz, P. Yang, R.A. Freeman, "Decentralized environmental modeling by mobile sensor networks," *IEEE Trans. Robotics*, vol. 24, no. 3, pp. 710-724, 2008.
[10] R. Olfati-Saber. "Distributed Kalman filtering for sensor networks," *Proc. of the 46th IEEE Conference on Decision and Control*, New Orleans, LA, Dec. 2007
[11] R. Olfati-Saber, "Distributed tracking for mobile sensor networks with information-driven mobility," in *Proc. American Control Conference*, New York, NY, 2007.
[12] B.S. Rao, H.F. Durrant-Whyte, "Fully Decentralized Algorithm for Multisensor Kalman Filtering," in *IEEE Proceedings - D*,vol. 138, no. 5, pp.413 - 420, 1991.
[13] F. Schweppe, "Recursive state estimation: unknown but bounded errors and system inputs," *IEEE Trans. Automat. Contr.*, vol. AC-13, no. 1, pp. 22-28, 1968.
[14] S. N. Simic and S. Sastry, "Distributed environmental monitoring using random wireless networks," *2nd Intern. Workshop Inform. Processing in Sensor Networks* PARC Palo Alto, CA, 2003.
[15] U. Shaked and Y. Theodor, "H_∞ - estimation: A tutorial," in *Proc. 31st IEEE Conf. Decision Contr.*, Tuscon, AZ, 1992.
[16] X. Shen and L. Deng, "Game theory approach to discrete H_∞ filter design," *IEEE Trans. Signal Processing*, vol. 45, no. 4, pp. 1092 - 1095, 1997.
[17] S. Susca, S. Martinez and F. Bullo, "Monitoring environmental boundaries with a robotic sensor network," in *Proc. American Control Conference*, pp. 2072 - 2077, 2006.
[18] H. Tsanknakis and M. Athans, "Tracking maneuvering targets using H_∞ filters," in *Proc. American Control Conference*, Baltimore, MD, 1994.
[19] S. Xie, L. Xie and S. Rahardja, "An LMI-based decentralized H_∞ filtering for interconnected systems," *ICASSp 2003*, 2003.
[20] P. Yang, R.A. Freeman and K.M. Lynch, "Multi-agent coordination by decentralized estimation and control," submitted for publication in *IEEE Trans. Automat. Contr.*

[21] F. Zhao, J. Shin and J. Reich, "Information-driven dynamic sensor collaboration for tracking applications," *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 61 - 72, March 2002.