Cooperative State Estimation for Mobile Sensors with Optimal Path Planning

Hossein Hajimirsadeghi

***Abstract*—This paper studies an optimal control problem of maximizing quality of estimation while minimizing control energy for a set of noisy mobile sensors. The cost functional is a weighted sum of the error covariance throughout the maneuver as well as the terminal time, and the control energy throughout the maneuver constrained to sensors motion dynamics and equations of a linear time varying filter for the error covariance. This problem is transformed to a two-point boundary value problem using Pontryagin maximum principle. Boundary conditions with free and fixed terminal sensor states are considered and a mathematical continuation approach is presented to come up with each of the cases. Finally, numerical simulations are provided to investigate the applicability of the proposed approaches to solve the problem for different conditions and also illustrate some interesting aspects of the problem like the effect of cooperation.**

***Index Terms*—Cooperative Control, Kalman Filtering, State Estimation, mobile sensors.**

I. Introduction

MULTI sensory estimation of a process using a set of mobile sensors can be considered as a problem of formation control which is one of the main categories of cooperative control for multi-agent systems. In fact, cooperative estimation can help to increase the quality of estimation by providing more measurements as well as more spatial perceptions, and also to improve robustness of the whole system to failure of some of the individuals. Hence, it is declared that cooperative estimation is more accurate and more fault tolerant. In this respect, moving sensors are more advantageous because of a new degree of freedom they make for the optimal estimation problem. This degree of freedom which is the sensor trajectory can be employed to enhance the accuracy of estimation by making the sensor get close to the process and consequently unveiling some of the uncertainties like measuring noises. However, the required energy for moving the sensors during the maneuver is a crucial factor that must not be ignored. The problem of simultaneous maximization of the estimation and minimization of the control effort for a set of mobile sensors is the problem we are to address in this study. Indeed, this is the optimal control problem which is proposed in [1], and the present work provides a deeper investigation of the problem as well as offering suggestions for new problem conditions and scenarios like fixed final sensor states accompanied by more detailed and complex simulations.

In this paper, we use the concept of central (centralized) Kalman filtering which is addressed in many applications of multi-agent estimation to find the optimality equations. Multi-agent Kalman filtering is proposed in literature in a variety of forms. In [2], the authors introduce combined sensing and control with decentralized Kalman filtering. Distributed Kalman filtering is derived in [3] by transforming the central Kalman filter to a set of micro Kalman filters (μKF) and using consensus filters. In [4], a novel multi-agent Kalman filter is described in a decentralized approach that makes use of estimations of other sensors rather than their measurements. In [5], moving sensors (UAVs) are considered and a low-pass filter is suggested to update the sensors' estimations. Moreover a fault detection mechanism is exploited to identify the best estimations. In [6], a Kalman filtering scheme is employed in a fire detection mission for estimation of alarm locations and a Bayesian approach is manifested to obtain the probability of potential alarms.

In Section II, the optimal control problem we are addressing in this study is described and formulated. Section III is dedicated to two-point boundary value problems with a quick review of numerical methods to solve them. The proposed approaches to solve the optimal control problem of section II are also explained in this section. In section IV, simulation studies for different conditions are summarized, and finally, conclusions are drawn in section V.

II. Problem Description and Formalism

A. Problem Description

The classical Kalman filter gives an optimal linear estimation of the process. However, minimizing the estimation error is not the only concern of designers in many applications of cooperative control. For example in the case of mobile sensors, there are additional degrees of freedom to make optimal trajectories towards the targets for the purpose of estimation accuracy, energy efficiency, meeting the specified waypoints, etc. So, there should be a tradeoff between different goals of the mission to perform the whole project perfectly. While estimation error can be diminished by moving through "sweet spots" of the sensors [7], [8], control effort and spatial constraints of the mission and environment are important factors that must not be ignored. Moreover, the designers might be interested to investigate importance of each of the factors, and this is the time when optimal control strategies can be employed to make the best decisions.

Report Prepared Feb. 9, 2009.
H. Hajimirsadeghi is an M.Sc. student of Control Engineering in the school of Electrical and Computer Engineering, University of Tehran, (Std. No. 810187579; e-mail: h.hajimirsadeghi@ece.ut.ac.ir).

In [1], an optimal control problem is proposed to come up with the challenges described above. This problem is to minimize a cost functional which is a weighted sum of estimation error and control energy subject to equations of sensors motion and error covariance matrix of a linear filter. In the following part we provide the mathematical formulation for this problem according to [1].

B. Formalism

Consider the linear process characterized by linear time varying equations:

$$\dot{x}(t) = A(t)x(t) + B(t)w(t),\ y(t) = C(t)x(t) \tag{1}$$

where $\boldsymbol{x}(t) \in \mathbb{R}^n$ is the state of the process, $w(t) \in \mathbb{R}^r$ is a white Gaussian noise with zero mean and covariance given by $E[w(t)w'(t-\tau)] = \delta(\tau)Q(t)$, where $Q(t)$ is a symmetric positive definite matrix for all t, and $y(t) \in \mathbb{R}^m$ is the output to be measured by the sensors. $A(t)$ is $n \times n$, $B(t)$ is $n \times r$, and $C(t)$ is $m \times n$, where n is the number of state variables for process, r is the dimension of the process noise, and m is the number outputs for the process or the number of measurements by all the sensors. $x(t_0)$ is a Gaussian random variable and independent of the process noise.

Measurements are described by the following equation:

$$z(t) = y(t) + v(t) = C(t)x(t) + v(t) \tag{2}$$

where $v(t) \in \mathbb{R}^m$ is Gaussian white noise with zero mean and covariance given by $\delta(\tau)R(t)$, where $R(t)$ is a symmetric positive definite matrix for all t. In this study, we assume that R is a function of the distance between the sensor and the process. Hence, in the rest of this paper, we use $R(x_s)$ to show that R is dependent on the sensor location (the process location is fixed).

The linear time varying filter is given by

$$\dot{\hat{x}}(t) = F(t)\hat{x} + G(t)z(t) \tag{3}$$

where $F(t)$ is a $n \times n$ matrix and $G(t)$ is a $n \times m$ matrix. By defining $e(t) = x(t) - \hat{x}(t)$, the error dynamics will be

$$\dot{e}(t) = \big(A(t) - F(t) - G(t)C(t)\big)x(t) + F(t)e(t) + B(t)w(t) - G(t)v(t) \tag{4}$$

From (4) it can be concluded that $F(t)$ should be equal to $A(t) - G(t)C(t)$ to have an unbiased estimator. Finally, by defining the error covariance matrix, $\Sigma(t) = E[e(t)e'(t)]$, the following equation can be derived:

$$\dot{\Sigma}(t) = \big(A(t) - G(t)C(t)\big)\Sigma(t) + \Sigma(t)\big(A(t) - G(t)C(t)\big)' + B(t)Q(t)B'(t) + G(t)R(t)G'(t), \Sigma(t_0) = \Sigma_0 \tag{5}$$

Now, we are going through the equations for mobile sensors. Each sensor satisfies the following dynamics:

$$\dot{x}_{s,i} = A_{s,i}x_{s,i} + B_{s,i}u_i,\ x_{s,i}(t_0) = x_{s,i}^0 \tag{6}$$

$i = 1, \dots, N$, where N is the number of sensors, $x_{s,i} \in \mathbb{R}^4$ is the state of sensor i composed of sensor locations, $p_i(t) \in \mathbb{R}^2$ and velocities, $v_i(t) \in \mathbb{R}^2$. $u_i \in \mathbb{R}^2$ is the control input for sensor i. By aggregating the sensors' dynamics in combined vectors and matrices according to the following formulations: $x_s(t) = \big(x_{s,1}(t), x_{s,2}(t), \dots, x_{s,N}(t)\big)$, $u(t) = \big(u_1(t), u_2(t), \dots, u_N(t)\big)$, $A_s = diag\Big(A_{s,1}(t), A_{s,2}(t), \dots, A_{s,N}(t)\Big)$, $B_s = diag\Big(B_{s,1}(t), B_{s,2}(t), \dots, B_{s,N}(t)\Big)$, one can obtain the compact form described in (7)

$$\dot{x}_s = A_s x_s + B_s u_i,\ x_s(t_0) = x_s^0 \tag{7}$$

Now the optimal control problem for cooperative estimation and sensor motion planning which is minimization of weighted sum of the error covariance and control energy acting on the sensors is formulated:

$$J = Tr\big[M(t_f)\Sigma(t_f)\big] + \int_{t_0}^{t_f} \tfrac{\alpha}{2}\|u_s\|^2 + Tr[K\Sigma(t)]\,dt \tag{8}$$

where $\alpha \geq 0$ is the control weighting parameter, Tr denotes the trace of a square matrix, $M(t_f)$ is a $n \times n$ symmetric positive definite weighting matrix for the error covariance at the final time, and K is a $n \times n$ symmetric positive semi-definite weighting matrix for the covariance error during the process. $G(t)$ and $u_s(t)$ are variables that should be found by solving the above-mentioned optimal control problem subject to (5) and (7) which are dynamics of the error covariance and sensors' motion, respectively.

The Pontryagin maximum principle can be used to come up with this problem. We firstly form the Hamiltonian in (9) using $\Lambda(t)$ as a $n \times n$ generalized momentum matrix dual to $\Sigma(t)$ and $\lambda(t) \in \mathbb{R}^n$ as a generalized momentum vector dual to x_s, and then two of the optimality conditions are derived in (10) by solving the differential equations: $\partial H/\partial G(t) = 0$ and $\partial H/\partial u(t) = 0$. Finally, the two remaining optimality conditions are obtained by in (11) and (12). For more detailed calculations one might see [1].

$$H = Tr\big[\dot{\Sigma}(t)\Lambda'\big] + \lambda(t).\dot{x}_s(t) + \tfrac{\alpha}{2}\|u_s\|^2 + Tr[K\Sigma(t)] \tag{9}$$

$$G^*(t) = \Sigma(t)G'(t)R^{-1}\big(x_s(t)\big), u^*(t) = -\tfrac{1}{\alpha}B_s'(t)\lambda^*(t) \tag{10}$$

$$\dot{\Lambda}^*(t) = -\partial H/\partial\Sigma = -\big(A(t) - G^*(t)C(t)\big)'\Lambda^*(t) - \Lambda^*(t)\big(A(t) - G^*(t)C(t)\big) - K \tag{11}$$

$$\dot{\lambda}^*(t) = -\partial H/\partial \mathrm{x_s} = -A_s'\lambda^*(t) + \left(I_{4N} \otimes \frac{\partial f(R)}{\partial(rs(R))}\right)^* \left(\frac{\partial\big(rs(R)\big)'}{\partial x_s}\right)^* \tag{12}$$

The symbol $\otimes$ is the Kronecker product. $rs(A)$, where A is $n \times m$, is a row vector obtained by taking the rows of A and stacking them horizontally to obtain $1 \times nm$ row matrix and $f(R) = Tr[GRG'\Lambda']$.

For free terminal sensor locations and velocities, the boundary conditions are shown in (13) while for fixed terminal sensor locations, the boundary equations change to those in (14), and finally, in the case of fixed sensor locations and velocities the conditions are described in (15).

$$\begin{aligned} \Sigma(t_0) &= \Sigma_0 \\ x_s(t_0) &= x_s^0 \\ \lambda^*(t_f) &= 0 \\ \Lambda^*(t_f) &= M(t_f) \end{aligned} \tag{13}$$

$$\begin{aligned} \Sigma(t_0) &= \Sigma_0 \\ x_s(t_0) &= x_s^0 \\ p_s(t_f) &= p_s^f \\ \lambda^{2^*}(t_f) &= 0 \\ \Lambda^*(t_f) &= M(t_f) \end{aligned} \tag{14}$$

$$\begin{aligned} \Sigma(t_0) &= \Sigma_0 \\ x_s(t_0) &= x_s^0 \\ x_s(t_f) &= x_s^f \\ \Lambda^*(t_f) &= M(t_f) \end{aligned} \tag{15}$$

where p_s is the vector of all sensor locations and λ^{2^*} is the generalized momentum vector associated with the vector of all sensor velocities. Note that the solution to the proposed optimal control problem with free terminal sensor locations and velocities for $\alpha = \infty$ (i.e., sensors are immobile) and $K = 0$ leads to the classical Kalman filter [1].

III. Two Point Boundary Value Problems and Numerical Approaches

A two-point boundary value problem is a problem (TPBVP) in which there are constraints on both initial and terminal values for a set of differential equations. A general TPBVP is shown is shown in (16) and (17).

$$\dot{y}(t) = f(t, y); \quad a \le t \le b \tag{16}$$

$$r\big(y(a), y(b)\big) = 0 \tag{17}$$

where (16) describes the dynamics of the system and (17) defines the boundary conditions on the system. TPBVPs from optimal control problems have usually separated conditions in the form of $r_1\big(y(a)\big) = 0$ and $r_2\big(y(b)\big) = 0$. In this section, a short review of some existing methods for TPBVPs is presented and then our approach to solve the optimal control problem introduced in section II is elaborated for different terminal conditions.

A. *Methods to Solve Two Point Boundary Value Problems*

There exist a number of numerical methods to solve TPBVPs including shooting, collocation and finite difference methods [11], [12], [13]. Among the shooting methods, Simple Shooting Method (SSM), and Multiple Shooting Method (MSM) are the most popular and used in many applications. In SSM, the TPBVP is converted to an initial value problem, and the initial values of the variables are varied to fulfill the desired final conditions. However, SSM is not very efficient in the problems which are sensitive to initial conditions. MSM has been proposed to come up with this problem. In MSM, variables are set at initial time, and then the differential equations are integrated until the distance from a corresponding point on a pre-defined reference path exceeds a tolerance value. After that the integration starts from the corresponding point on the reference path and the previous process repeats until the final time is reached. One of the disadvantages of MSM is the large number of parameters which are to be updated in each iteration which slows down the whole process.

Collocation Method (CM) and Finite Difference Method (FDM) are based on the transformation of the TPBVP to linear or nonlinear algebraic equations and then using suitable methods to solve these equations. In fact, CM and FDM are much more complex to set up than the shooting methods.

In this study, MATLAB®'s bvp4c.m is used to solve TPBVPs [9], [10]. The collocation method of this program leads to a system of nonlinear algebraic equations that is solved by a variant of Newton's method. This program requires an initial guess for time parameterized state variables which is a real challenge for the posed optimal control problem described in section II because of the nonlinearity of the equations and also abundance of the states and equations. The approach to come up with this challenge is the use of mathematical continuation approach which is elaborated for different conditions in the following part.

Another problem with bvp4c is that it approximate the partial derivatives (e.g. in calculation of Jacobian Matrix) with finite differences. The program is more robust and efficient when it is provided by analytical derivatives [14], however, it is inconvenient for the user to make analytical derivatives when the number of state variable increases. To handle this problem, one might use bvp4cAD.m [15], which employs the MAD [16] package of automatic (algorithmic) differentiation (AD) designed for MATLAB® to have accurate derivative values. AD is based on systematic application of the chain rule of differentiation to the floating point evaluation of a function and its derivatives. In this approach, there are no discretization and cancellation errors, hence the results are accurate and without roundoff [14].

B. *The Proposed Approach to Solve the Optimal Control Problem*

In [1], the mathematical continuation approach was proposed to solve the problem for free terminal sensor locations and velocities. As it was declared in section II, the optimal control problem in (8) leads to classical Kalman filter for the case of $\alpha = \infty$ and $K = 0$. So, by defining the cost functional in (18), and setting $\varepsilon = 0$ the solution will be the classical Kalman filter which is easily obtained since it is an initial boundary problem. Consequently, the classical Kalman solution can be used as an initial guess for a higher ε and gradually, by using continuation approach ε increases to 1 and the original problem is solved.

$$\begin{aligned} J_\varepsilon = {} & Tr\big[M(t_f)\Sigma(t_f)\big] \\ & + \frac{1}{\varepsilon}\left(\int_{t_0}^{t_f} \frac{\alpha}{2}\|u_s\|^2 + Tr[K\Sigma(t)]\,dt\right) \end{aligned} \tag{18}$$

For fixed terminal sensor locations or velocities we propose an approach which can be used to come up with scalability of the original problem. Consider a problem in which a moving object with the dynamics described in (6) wants to go from one point to another point while satisfying specified initial and terminal velocities with the minimum control effort. This is a classical problem of minimum energy (fuel) problem which can be easily solved by analytical calculations or numerical methods because there are a few unknown states and co-states in the optimality equations. Also in this problem, according to linear equations in (6), we face a LQR problem which is easily tractable in optimal control literature. Now, consider the cost functional in (19). When $\varepsilon = 0$, the solution is equivalent to solution of the classical minimum energy problem for each of the moving sensors. So, by considering this solution as an initial guess and performing the mathematical continuation approach, the original optimal control problem is solved.

$$J_\varepsilon = \varepsilon\,.Tr\left[M(t_f)\Sigma(t_f)\right] + \int_{t_0}^{t_f} \frac{\alpha}{2}\|u_s\|^2 + \varepsilon\,.Tr[K\Sigma(t)]\,dt \tag{19}$$

IV. Simulation Studies

Manipulating the equations and relations described in section II is not straightforward for simulation of the problems with multiple coupled and correlated processes. In this section, we want to come up with the problems of multiple sensors and processes in which the optimality conditions constitute more than twenty variables. So, for the sake of simplicity, a set of n decoupled and uncorrelated processes are considered, each satisfying a one-dimensional linear time invariant state equation. In fact, this is a common assumption in many of cooperative control missions when the processes are distributed in different areas, and the goal is an accurate sensing and estimation of the state variable for each process (e.g. fire detection mission). The problems which are studied in this section are defined according the formalism provided in the following paragraph.

In the case of decouple and uncorrelated processes we have $A = diag(A_1, \ldots, A_n)$ and $B = diag(B_1, \ldots, B_n)$ a diagonal $n \times n$ matrix (it is assumed that $r = n$). C is chosen to be the $Nn \times n$ row block matrix made up of concatenating N identity matrices of dimension $n \times n$ row-wise (so $m = Nn$), Q is a diagonal matrix equal to $diag(Q_1, \ldots, Q_n)$, and Σ is a diagonal matrix of the form $\Sigma = diag(\Sigma_1, \ldots, \Sigma_n)$. Assuming uncorrelated measurements, the covariance matrix $R(x_s)$ is a $Nn \times Nn$ block diagonal matrix of the form $R(x_s) = diag(R^i(p_i), \ldots, R^i(p_i))$, where each $R^i(p_i)$ $(i = 1, \ldots, N)$ is a $n \times n$ diagonal matrix: $R^i(p_i) = diag(R_1^i(p_i), \ldots, R_n^i(p_i))$ in which p_i is the position of sensor i and $R_j^i(p_i)$ is a function of the distance between sensor i and process j. In the experiments which are simulated in this section $R_j^i(p_i)$ is defined as follows: $R_j^i(p_i) = \beta^i\left\|p_i - s_j\right\|^2 + \gamma^i$.

For the sensors we follow the formalism explained in section II. In all the simulations of this section, the following matrices are employed in sensor motion dynamics:

$$A_{s,i} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, B_{s,i} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, i = 1, \ldots, N$$

Finally, it is assumed that $M(t_f)$ and K are $n \times n$ diagonal matrices of the form $M(t_f) = diag(M_1, \ldots, M_n)$ and $K = diag(K_1, \ldots, K_n)$.

In the rest of this section, we firstly simulate a simple example to have better understanding of the subject and investigate some interesting aspects of the problem like effect of mobility and cooperation. Then three more complex problems with different terminal conditions are studied.

A. Detailed Analysis of a Simple Example

The problem we are to study in this part is composed of a one dimensional process and a single sensor with free terminal states optimality conditions. The characteristics of the process and the sensor according to the formalism described in section II is provided in Table I. Simulation results obtained by the mathematical continuation approach introduced in section III are provided in Fig. 1, Fig. 2. In this experiment ε is set to 0 where we get the classical Kalman filter and gradually, it is increased to 1 with increments of $\delta = 0.05$ in which the solution for the original optimal control problem is achieved.

TABLE I
Parameters for Simulation of Simple Example with 1 Sensor

n	N	a_1	B_1	Q_1	M_1	α
1	1	-0.5	1	1	1	0.04
$\boldsymbol{p_1}(0)$	$\boldsymbol{v_1}(0)$	$\boldsymbol{s_1}(0)$	$\Sigma_1(0)$	β^1	γ^1	t_f
(10, 10)	(0,-2)	(0,0)	200	5	3	10

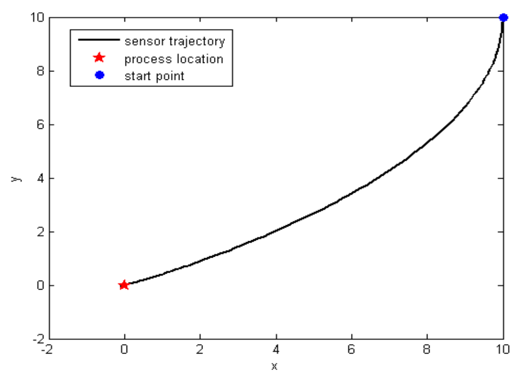

Fig. 1. Sensor motion planning for an example of 1 sensor and 1 process.

Fig. 1 shows the trajectory of motion for the sensor, and one can see that the sensor reaches the process location in the specified time. Although it seems intuitive for a sensor to track the process location in a system of a single process to enhance estimation accuracy, however, it is not the outcome of all cases, i.e., when tracking the target is so energy-consuming, the sensor might accept estimation error for the sake of saving energy. Fig. 2a illustrates the control signal in the process of mathematical continuation approach for two cases, one with $\varepsilon = .05$ and one with $\varepsilon = 1$. As it is shown in the figure, control cost increases as ε (which is disproportionate to control weighting parameter) grows, and consequently, the chance for approaching the target and improving the estimation increases. Fig. 3 compares estimation accuracy between the immobile ($\varepsilon = 0$ or classic Kalman filtering) and the mobile sensor. To have a fair comparison the estimation is performed for 50 times and the results are averaged. In addition, the error bar for the interval of $\pm 3\sigma$ is depicted in this figure which determines the interval that the estimation is placed in it with probability of about 99% (for Gaussian noise). It can be observed that the mobile sensor outperforms the immobile one in accuracy of estimation as the time goes on.

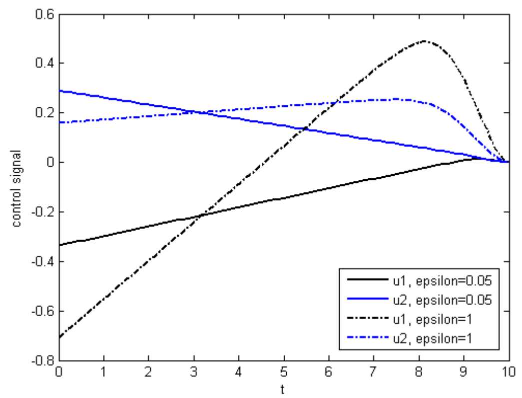

Fig. 2. Control Inputs during the mathematical continuation process.

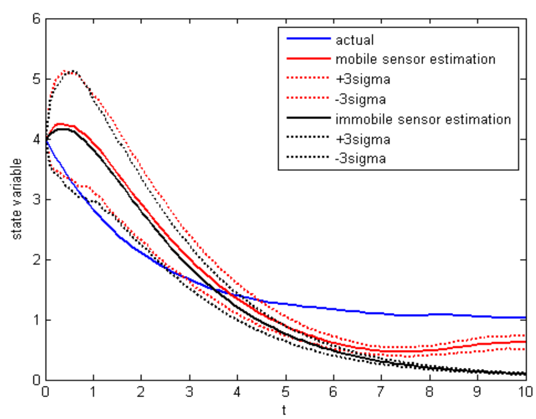

Fig. 3. Comparison between mobile and immobile sensor estimation

Investigating the Effect of Cooperation

To demonstrate the effect of cooperation in estimation, we repeat the experiment described above with two mobile sensors. In fact, another sensor completely the same as the first one is positioned in the $x - y$ plane with equal initial distance from process location. Simulation results for this experiment are provided in Fig. 4 and Fig. 5. It can be observed from Fig. 5 that cooperative estimation has better performance than single sensor estimation.

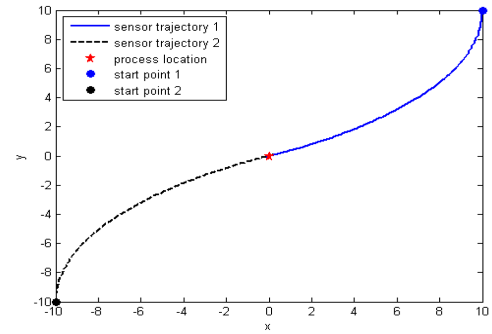

Fig. 4. Sensor motion planning for two identical sensors

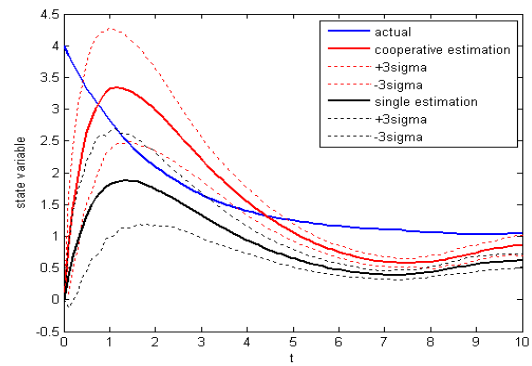

Fig. 5. Comparison between multiple sensors estimation and single sensor estimation

B. *Free Terminal States*

In this part a more complex problem with two sensors and three targets is studied considering free terminal states. The parameters for simulating this problem are shown in Table II. Mathematical continuation approach for free terminal states with $\delta = 0.05$ was employed to solve this problem and the results are depicted in Fig. 6.

TABLE II
PARAMETERS FOR SIMULATION OF SIMPLE EXAMPLE WITH 1 SENSOR

sensors	#	$\boldsymbol{p}(0)$	$\boldsymbol{v}(0)$	β	γ	Optimal Control Problem	
	1	(10, 10)	(0,-1)	5	3	α	0.04
	2	(-12, 15)	(2,1)	5	3	t_f	10
process	#	$\boldsymbol{s}(0)$	Q	K	M		
	1	(0,0)	1	0	1		
	2	(10,0)	1	0	1		
	3	(0,10)	1	0	1		

C. *Fixed Terminal States*

To evaluate the performance of our proposed algorithm to solve the problems with fixed terminal states (sensor locations or velocities), an example of two sensors and three processes is considered. The statistics for this problem is provided in Table IV. Two experiments are performed on this problem with two different terminal conditions. For the first experiment both the terminal locations and velocities are fixed while in the latter terminal sensors velocities are free. The results of simulation for this problem using the mathematical continuation approach are summarized in Fig. 7 and Fig. 8. It can be observed that in both of the experiments, the sensors try to approach the processes to have more accurate estimation. Comparing these two figures, it is notable that the control energy for the case with fixed final velocities is more than the second case because of the extra constraints imposed on the sensors. Likewise, the sensors with free terminal velocities are freer to go near the targets and consequently have better estimation.

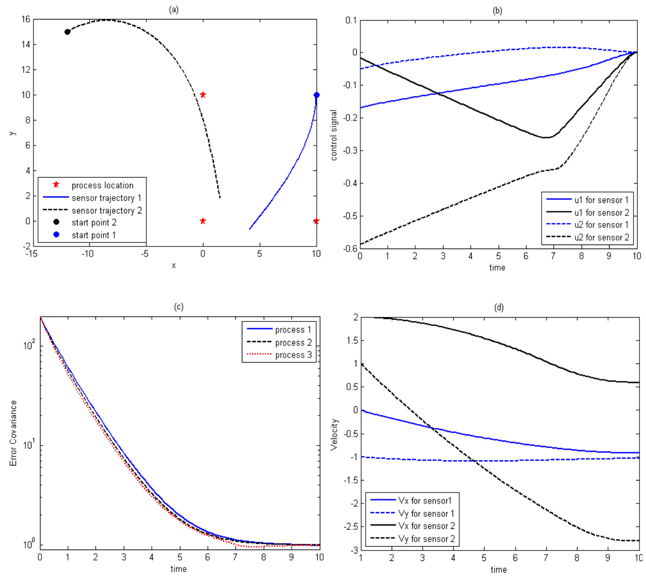

Fig. 6. Cooperative control and estimation with free terminal state variables. (a) motion planning, (b) control inputs, (c) error covariance, and (d) trace of velocities.

TABLE III
PARAMETERS FOR SIMULATION OF SIMPLE EXAMPLE WITH 1 SENSOR

sensors	#	$\boldsymbol{p}(0)$	$\boldsymbol{v}(0)$	$\boldsymbol{p}(t_f)$	$\boldsymbol{v}(t_f)$	β	γ	Optimal Control Problem	
	1	(10, 10)	(0,-1)	-10,-15)	(-2,-1)	5	3	α	0.04
	2	(-12, 15)	(2,1)	(10,-8)	(1,-1)	5	3	t_f	10
process	#	$\boldsymbol{s}(0)$	Q	$\Sigma(0)$	K	M			
	1	(0,0)	1	200	0.01	0			
	2	(10,0)	1	200	0.01	0			
	3	(0,10)	1	200	0.01	0			

I. CONCLUSION

An optimal control problem for simultaneous enhancement of estimation and control effort of a set of mobile sensors with linear motion dynamics was developed in this study. A method based on mathematical continuation approach was proposed for the case of fixed terminal sensor states (location or velocity), and the applicability of that was shown through a number of simulation studies. In addition, the previously studied case of free terminal sensor states was elaborated and analyzed with more details and discussions on more complex test cases.

For future work, a similar optimal control problem can be described for optimal cooperative estimation and planning in a decentralized approach where the constraining equations are individual sensor motion dynamics and the equations of error covariance for a local set of sensors in a distributed or decentralized form. Also, the problem can be retreated for nonlinear sensor motion dynamics or constrained state variables which are the common case of real problems (e.g., Dubins car model with maximum turning rate). Finally, the optimality conditions and settings provided in the presented work are not consistent when the estimated variable is the location of the process because of the fact that the covariance matrix R which is defined to be deterministic becomes a stochastic estimation. Hence, another suggestion is to rederive the relations for the case when R is a function estimated variables.

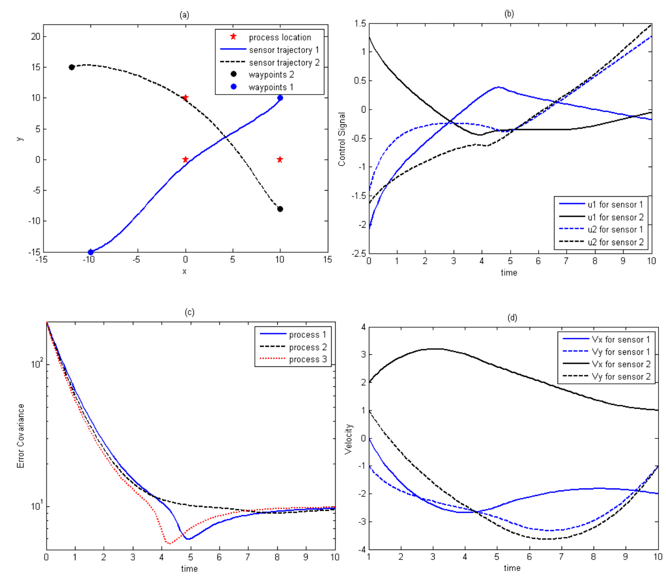

Fig. 6. Cooperative control and estimation with fixed terminal sensor locations and velocities. (a) motion planning, (b) control inputs, (c) error covariance, and (d) trace of velocities.

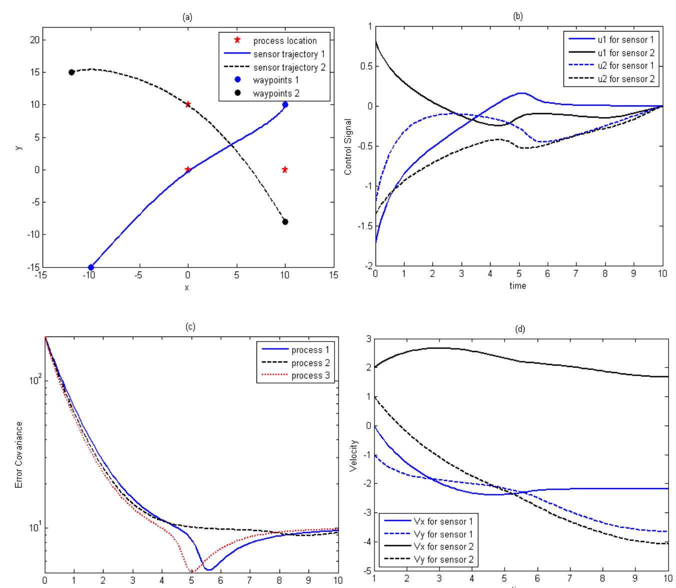

Fig. 7. Cooperative control and estimation with fixed terminal sensor locations and free terminal velocities. (a) motion planning, (b) control inputs, (c) error covariance, and (d) trace of velocities.

REFERENCES

[1] I. I. Hussein, "Kalman Filtering with optimal sensor motion planning," in *Proc. American Control Conference*, Washington, Jun. 2008, pp. 3548–3553.

[2] J. Speyer, "Computation and transmission requirements for a decentralized linear-quadratic-gaussian control problem," *IEEE Transactions on Automatic Control*, vol. 24, no. 2, pp. 266–269, 1979.

[3] R. Olfati-Saber, "Distributed kalman filter with embedded consensus filters," in *Proc. IEEE Conference on Decision and Control*, Seville, Spain, 2005.

[4] W. Ren, R. W. Beard, and D. B. Kingston, "Multi-agent kalman consensus with relative uncertainty," in *Proc. American Control Conference*, Portland, OR, 2005, pp. 1865–1870.

[5] Y. Kim, D. W. Gu, and I. Postlethwaite, "Fult-tolerant cooperative target tracking in distributed UAV networks," in *Proc. 17th World Congress of the International Federation of Automatic Control*, Seoul, July 2008, pp. 8878–8883.

[6] L. Merino, F. Caballero, J. R. M. Dios, J. Ferruz, and A. Ollero, "A cooperative perception system for multiple UAVs: application to automatic detection of forest fires," *J. of Field Robotics*, vol. 23, 2006, pp. 165–184.

[7] T. H. Chung, V. Gupta, J. W. Burdick, and R. M. Murray, "On a decentralized active sensing strategy using mobile sensor platforms in a network," in *Proc. IEEE Conference on Decision and Control*, Paradise Island, Bahamas, 2004, pp. 1914–1919.

[8] Y. Mostofi, T. Chung, R. Murray, and J. Burdick, "Communication and sensing tradeoffs in decentralized mobile sensor networks: A cross-layer design approach," in *Proc. International Conference on Information Processing in Sensor Networks*, April 2005. [Online]. Available: http://www.cds.caltech.edu/yasi/papers/IPSN.pdf

[9] MATLAB®, Web, April 2006. [Online]. Available: http://www.mathworks.com/products/matlab/

[10] L. F. Shampine, M. Reichelt, and J. Kierzenka, "Solving boundary value problems for ordinary differential equations in matlab with bvp4c," [Online]. Available: http://www.mathworks.com/support/solutions/files/s8314/bvppaper.pdf

[11] S. M. Roberts and J. S. Shipman, "*Two-Point Boundary Value Problems: Shooting Methods*, Elsevier, New York, NY., 1972, pp. 1–231.

[12] H. J., Reinhardt, *Analysis of Approximation Methods for Differential and Integral Equations*, Springer-Verlag, New York, NY., 1985, pp. 72–73.

[13] R. Holsapple, R. Venkataraman, and D. Doman, "A New, fast numerical method for solving two-point boundary-value problems," *Journal of guidance, control, and dynamics*, vol. 27, no2, 2004, pp. 301–304.

[14] L. F. Shampine, R. Ketzscher, S. A. Forth, "Using AD to solve BVPs in MATLAB," *ACM Transactions on Mathematical Software (TOMS)*, vol. 31, Mar. 2005, pp. 79–94.

[15] S. Forth, L. F. Shampine, "bvp4cAD- An Automatic Differentiation Enabled Boundary Value Solver," [Online]. Available: http://www.amorg.co.uk/AD/ADODE/bvp4cAD/

[16] S. A. Forth, "User guide for MAD a Matlab automatic differentiation toolbox," *Applied Mathematics and Operational Research Report AMOR*, Cranfield University (RMCS Shrivenham), Swindon, SN6 8LA, UK., June 2001.