Mobile Netw Appl (2009) 14:336–349
DOI 10.1007/s11036-008-0151-4

Cooperative Estimation for Coordinated Target Tracking in a Cluttered Environment


Benjamin I. Triplett · Daniel J. Klein · Kristi A. Morgansen

Published online: 5 February 2009

Abstract The purpose of the work in this paper is to gain insight into several fundamental questions that arise whenever a distributed multiple-vehicle control system is endowed with communication capabilities. These fundamental questions include: what data should each vehicle share?, how frequently should communication take place?, and what benefit does communication provide? These questions are evaluated with respect to a target tracking task in which multiple pursuit vehicles estimate the state of a target vehicle. This task has three main components: communication, estimation, and control. Communication takes place on a broadcast network, estimation is achieved with an Unscented Kalman Filters, and the controller is behavior-based. Simulation results show that communication always improves distributed estimate results. Which information to transmit depends on available bandwidth, and more frequent communication generally yields better estimates. Simulation results also show how coordinated control can be beneficial to target tracking in a cluttered environment.

Keywords cooperative estimation · coordinated target tracking · nonlinear filtering

1 Introduction

As autonomous vehicle systems become ubiquitous, means of exploiting the potential advantages of these systems are being investigated. Many applications involve a single autonomous agent, such as an unmanned airplane or submarine, often performing a task which is dangerous or monotonous for a human pilot. Additional capabilities can be achieved by using multiple autonomous vehicle systems coupled by sensing and communication. However, a number of fundamental, yet open questions arise when autonomous agents are endowed with communication capabilities. For example, what data should each agent transmit and when should the communication take place? What benefit does communication provide? The purpose of this paper is to shed light on these and other questions within the framework of a coordinated target tracking problem. Results are generated by simulation.

Coordinated target tracking is a task in which one or more autonomous pursuit vehicles estimate the state of a target vehicle. While a single pursuer may be able to adequately perform this task in some situations, a decrease in state estimate error can be achieved by using multiple pursuit vehicles. Further, in a cluttered environment, a single vehicle may not be able to keep track of the target vehicle due to sensor occlusions, whereas multiple pursuers can observe the target from several vantage points simultaneously.

The three main components of the coordinated target tracking task are: inter-vehicle communication, target state estimation, and coordinated control. The communication component addresses what data each vehicle should transmit in order to diffuse information efficiently throughout the network. The estimation component functions to combine local measurements with received information into a single fused estimate of the target state. Communication bandwidth, topology, and delay constraints correspond to a scenario

B. I. Triplett (✉) · D. J. Klein · K. A. Morgansen
Department of Aeronautics and Astronautics,
University of Washington, Seattle, WA 98195-2400, USA
e-mail: triplett@u.washington.edu

in which the probability that any two pursuit vehicles will have identical target state estimates is unlikely. Finally, the coordinated control problem addresses how to drive each vehicle so as to leverage the potential estimation benefits of a team while simultaneously avoiding obstacles.

In this paper, constant-speed and non-holonomic pursuit vehicles communicate with each other through a sequence of broadcast transmissions from one vehicle, selected sequentially at each time step, to all other vehicles. This network topology is consistent with a shared communication medium. To explore the question of what data each vehicle should transmit, three possible communication protocols are considered: transmit nothing, transmit one, two, or three recent measurements, and transmit a state estimate. Communicated data arrives delayed, and sensors fail both stochastically and when the line of sight to the target is occluded by an obstacle. Fusion of local measurements and received communications is achieved by a per-vehicle Unscented Kalman Filter (UKF). A behavior-based controller on each vehicle acts to disperse the vehicles about the target while avoiding collisions with other vehicles and obstacles.

Perhaps the work most closely related to this investigation is the square root sigma point information filter developed by Campbell and Whitacre [2, 13]. In Campbell's work, each pursuit vehicle shares information with every other pursuit vehicle after each observation. By sharing information so frequently, each agent is effectively able to recover the centralized target state estimate. Occasional communication dropouts and delay were addressed with a queuing system. This approach is ideal when the number of agents is sufficiently low and the available bandwidth is sufficiently high. The control objective was to maintain a fixed clock angle between vehicles, and obstacles were not considered.

Other related work on target tracking has been done by Eickstedt and Benjamin [5]. In their work, a pair of pursuit vehicles, coupled through a central database, track a target using a centralized EKF. The control was behavior-based, but also centralized, and obstacles were not considered. Some work on coordinated target tracking has been done with constant-speed and non-holonomic pursuit vehicles [7], however only the control component and resulting equilibrium formations were addressed.

Olfati-Saber and others created a series of Distributed Kalman Filtering (DKF) algorithms for sensor networks [10]. The objective of the DKF is to drive the target state estimate on each node towards a common state estimate. As compared to local (non-consensus) Kalman filtering, the DKF results in lower estimate error and reduced estimate disagreement between nodes. However, the algorithm was designed for linear target dynamics and communication delay was not considered.

The results presented in this paper differ from these previous works in that allowable communication bandwidth per vehicle is assumed to be sufficiently low that recovering the centralized state estimate is not possible. Another difference is the explicit inclusion of sensor-occluding obstacles. The behavior-based control is, unlike previous work, computed in a distributed manner.

This work is organized as follows. In the following section, the vehicle dynamics, sensor model, and communication network are introduced. Coordinated target estimation and pursuit are described in Sections 3 and 4, respectively. Simulation results and conclusions follow in the remaining two sections.

2 System

The target tracking system is composed of N pursuit vehicles and a target vehicle. The purpose of the pursuit vehicles is to track the target, and to that end, they are equipped with sensors that return the target position with limited precision and reliability. The pursuit vehicles are coupled by a communication network over which local and target state information is exchanged The target tracking environment is planar, and may contain obstacles that occlude sensor view of the target and impede pursuit vehicle trajectories.

2.1 Vehicle dynamics

The pursuit and target vehicles could be airplanes, submarines, automobiles, or other vehicles of this nature. These types of vehicles all use some form of steering control in order to follow a desired trajectory. Furthermore, for the sake of simplicity, and because most earth bound vehicles have dynamics that naturally decouple into vertical and horizontal planes, vehicle dynamics will be restricted to a plane perpendicular to gravity. Systems of this sort can be modeled with "unicycle" dynamics, in which the vehicle trajectory is manipulated by heading and speed controls, while the nonholonomic constraint dictates that the vehicle cannot have any component of its velocity in a direction perpendicular to its heading. A model of these vehicle dynamics is shown in Fig. 1, in which $r = [x, y]^T$ is the position vector of the vehicle, v is its velocity vector, and θ is

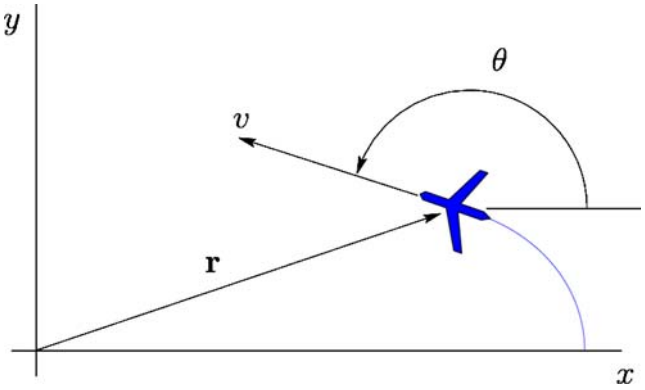

Fig. 1 States of the pursuit and target vehicles. The vector $\mathbf{r} = [x, y]^T$ is the vehicle position vector

its heading.The unicycle dynamics are typically written in Cartesian coordinates as

$$\begin{aligned}
\dot{x}^i &= v^i \cos\left(\theta^i\right), \ i = 1 \ldots N \\
\dot{y}^i &= v^i \sin\left(\theta^i\right) \\
\dot{\theta}^i &= u^i_{\text{steer}} \\
\dot{v}^i &= u^i_{\text{accel}}, \qquad (1)
\end{aligned}$$

where (see Fig. 1) x^i and y^i are the Cartesian coordinates of the i^{th} vehicle, θ^i is its heading, v^i is its speed, and u^i_{steer} and u^i_{accel} are its steering and speed controls, respectively. The speed control is included in the model for generality, but in this work the speed control is not used and hence set to zero. These dynamics govern the movement of the pursuit vehicles as they move in the plane.

Alternatively, the unicycle model can be written with second order dynamics. This choice is advantageous for the target model as it allows the UKF to estimate the turn rate of the target vehicle. The target model unicycle dynamics are written as

$$\begin{aligned}
\dot{x}^t &= v^t \cos\left(\theta^t\right) \\
\dot{y}^t &= v^t \sin\left(\theta^t\right) \\
\dot{\theta}^t &= \phi^t \\
\dot{v}^t &= w_v \\
\dot{\phi}^t &= w_\phi, \qquad (2)
\end{aligned}$$

where x^t and y^t are the Cartesian coordinates of the target, θ^t is its heading, v^t is its speed, and ϕ^t is its turn rate. The quantities w_v and w_ϕ are zero-mean Gaussian white noise terms that allow the estimator to change its estimates of the target velocity and heading rate, given that the target vehicles speed and heading commands are unknown to the observing agents.

2.2 Sensor reliability model

The sensors onboard the pursuit vehicles measure the planar position of the target relative to some inertial reference frame at regular discrete-time intervals of length T_s. The sensors used to identify the target are assumed to be omnidirectional with limited probability of target identification, at a given sensor measurement, which depends on the distance between the sensor and the target. When the sensor is in its "optimal range", the probability of successful target position measurement is constant and less than or equal to one. Outside this optimal range, the probability of target identification drops off quadratically until a second threshold is passed, beyond which the target is never sensed. The sensor probability model shown in Fig. 2 is used because it captures some simple limitations of fixed-resolution sensors, and can give interesting and relevant simulation results accordingly.

2.3 Communication network

The pursuit vehicles are assumed to have synchronized clocks for the sake of simplicity in this study. While full clock synchronization would be difficult to achieve in practice, the purpose of this study is not to explore the properties of unsynchronized clocks between communicating and coordinating agents. Communication is accomplished on a sequential broadcast network. A broadcast network is one in which at each time step h, of period T_b, one agent in the group transmits its data to all the other members. The communication

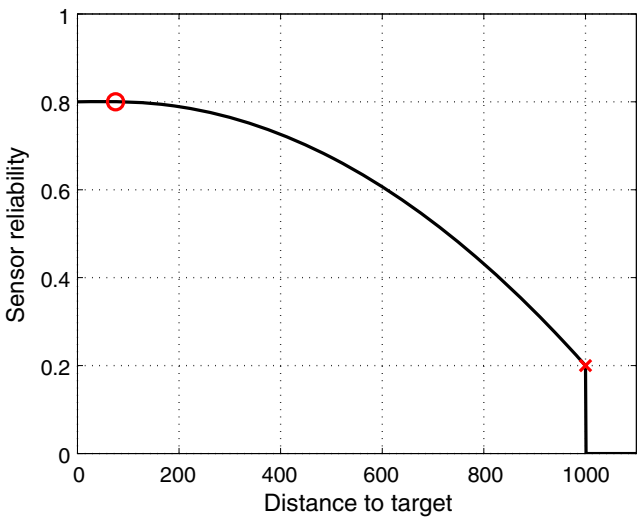

Fig. 2 Example sensor reliability dependence on vehicle distance to target. The *circle* denotes the end of the optimal sensor range, while the *x* denotes the absolute sensor range

period is assumed to be equal to an integer multiple of the sensing period, or $T_b = n_b T_s$, where $n_b \in \mathbb{N}$, so that communication and sensing are synchronized, with $T_b \geq T_s$. When the broadcasting agent is chosen in the simple sequence $\{1, 2, \ldots, N, 1, 2 \ldots\}$, the time between data transmissions for a given agent will be NT_b.

The broadcasting vehicle always transmits its current position, which in reality could be known from direct measurement or estimation, but is taken to be the actual simulated vehicle position in this work. In addition, information about the target vehicle is transmitted in one of three possible scenarios. First, in order to establish the effects of communication on estimator performance, simulations are conducted without any inter-vehicle communication. Second, a subset of available sensor measurements are transmitted by the broadcasting agent. Because the sensors can have less than perfect reliability, the broadcasting agent sends the most recent data it has, unless the data is older than its previous broadcast event, in which case no data is sent. Last, estimates, and associated covariance information, are transmitted by the broadcasting agent.

Communication is not instantaneous and is assumed to take an interval of length T_s (i.e. one sensor period) for transmission of the target measurement or estimate data. Furthermore, when a given agent becomes the broadcaster, its measurement data can be older than the time interval T_s because its sensors can be less than perfectly reliable. By the time the data sent by the broadcasting agent reaches one of the receiving vehicles, the data received will be delayed by a time interval between T_s and NT_b. Data older than NT_b is never transmitted. The relationship between communication and sensing events is depicted in Fig. 3.

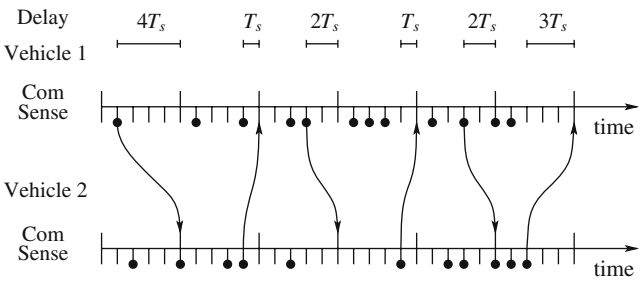

Fig. 3 An example of the temporal relationship between communication and sensing is depicted for two vehicles, each sharing their most recent measurement. Com refers to communication events which occur at the *hash marks* above the time lines. Sense refers to sensing events which are indicated by the *hash marks below the time lines*. *Black circles* denote successful measurement events and *arrows* point from the last measurement to the time the communicated data arrives at the other vehicle

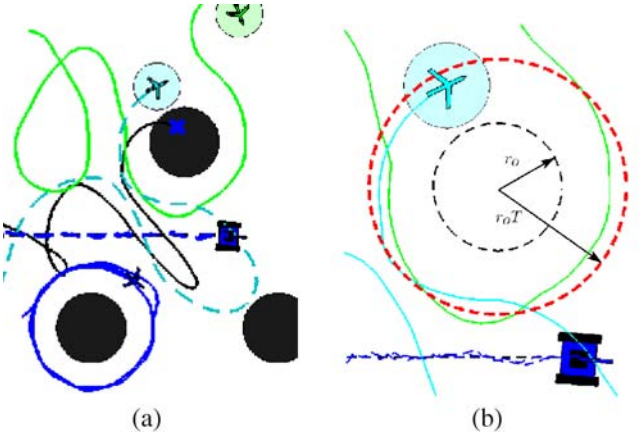

Fig. 4 **a** The target tracking environment. Obstacles are represented by *black circles* spaced 30 units apart, pursuit vehicles are depicted as airplanes, and the tank is the target. *Semi-transparent circles* on pursuit vehicles indicate no sensor view of target. **b** Depiction of obstacle radius and corresponding interaction radius

2.4 Clutter model

Clutter consists of identical circular obstacles, of radius r_o, that impede the pursuit vehicle trajectories and occlude their sensors from viewing the target. The pursuit vehicles do not interact with the obstacles unless they are within the obstacle threshold radius $r_{oT} > r_o$. Because no a priori knowledge is assumed about the obstacle locations, they are simply arranged in a regular pattern along the target trajectory. The spacing of this pattern is varied between experiments in order to investigate the effects of clutter density on the target tracking algorithms. A typical target tracking scenario is depicted in Fig. 4a.

3 Coordinated target estimation

An Unscented Kalman Filter (UKF) is used by each pursuit vehicle to compute target state estimates from sensor data. Target state measurement or estimate data from inter-vehicle communication arrives at the same rate, or a slower rate, than the local sensor data, and is always delayed by one or more discrete time steps. This data from communication is fused into the UKF at its correct time slot [9], after which the appropriate updates and corrections are reapplied so that the current estimate reflects the additional data.

3.1 Unscented Kalman filter

In prior work, the Extended Kalman Filter (EKF) was used for target state estimation [12]. However, the EKF

is an imperfect solution to the problem of nonlinear system estimation because it relies on linearization of the system dynamics in order to propagate the estimate error covariance. In practice, the EKF was vastly inferior to an alternative method of nonlinear state estimation: the Unscented Kalman Filter (UKF). The UKF does not rely on linearization (i.e. explicit computation of the system Jacobian) in order to propagate the estimate error covariance. This property proves advantageous when working with highly nonlinear systems, such as the unicycle dynamics problem. Several estimation techniques for nonlinear systems exist, some of which use the EKF algorithm but with higher order terms in the propagation of the estimate error covariance, and others, such as particle filters, that have very high accuracy but come with a correspondingly high computational cost. The UKF was deemed the best fit to the problem at hand in which accurate estimates of the target state were desired but without the computational cost of a particle filter.

The UKF algorithm used in this work is presented in entirety in [11] and is briefly reviewed here. To begin, consider a target vehicle system with dynamics written generally as $f(x(t)) \in \mathbb{R}^n$ with $x(t) \in \mathbb{R}^n$, where $n = 5$ for the unicycle dynamics given above. Sensors are used to make measurements of a subset of the target state. In this case, measurements, $\tilde{y}_k^i \in \mathbb{R}^m$, are made of the planar coordinates of the target, (x, y), so that $m = 2$. The target process and measurement models used for the estimator are then

$$\dot{x}(t) = f(x(t)) + Gw(t),\ w(t) \backsim N(0, Q_c) \tag{3}$$

$$\tilde{y}_k^i = Hx_k + v_k^i,\ v_k^i \backsim N\left(0, R^i\right), \tag{4}$$

where $w(t) \in \mathbb{R}^{n_w}$, with $n_w \leq n$, is the process noise with covariance $Q_c \in \mathbb{R}^{n_w \times n_w}$. The measurement made by the i^{th} agent at time step k is denoted $\tilde{y}_k^i \in \mathbb{R}^m$, and $v_k \in \mathbb{R}^m$ is the measurement noise which has covariance $R^i \in \mathbb{R}^{m \times m}$. Both noise terms are zero mean with Gaussian distributions. In Eq. 4, the measurement model of the i^{th} vehicle, $\tilde{y}_k^i$ is a sensor measurement. The matrix $G \in \mathbb{R}^{n \times n_w}$ models how process noise is incorporated into the target dynamics. For the systems considered here

$$G = \begin{bmatrix} 0\,0\,0\,1\,0 \\ 0\,0\,0\,0\,1 \end{bmatrix}^T$$

$$Q_c = \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_\phi^2 \end{bmatrix}$$

$$R^i = \begin{bmatrix} \left(\sigma_x^i\right)^2 & 0 \\ 0 & \left(\sigma_y^i\right)^2 \end{bmatrix}. \tag{5}$$

The measurement matrix is denoted $H \in \mathbb{R}^{m \times n}$. The pursuit vehicle sensors make a measurement of the target coordinates (x, y), so that $H = \left[\begin{smallmatrix} 1\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0 \end{smallmatrix}\right]$.

While the linearization of the dynamics is not needed in order to compute state estimates for the system, it can be used to check observability of the states for the given measurements. The target system linearization about a particular state, x_k, is given by $F(x_k) \equiv \partial f/\partial x|_{x_k}$, or

$$F(x_k) = \begin{bmatrix} 0\,0 & -\sin\theta_k & v_k\cos\theta_k & 0 \\ 0\,0 & \cos\theta_k & v_k\sin\theta_k & 0 \\ 0\,0 & 0 & 0 & 1 \\ 0\,0 & 0 & 0 & 0 \\ 0\,0 & 0 & 0 & 0 \end{bmatrix}. \tag{6}$$

When considering observations from a single pursuer, the system (Eqs. 3–4) has the observability matrix

$$\mathcal{O} = \begin{bmatrix} H \\ HF \\ HF^2 \\ HF^3 \\ \vdots \end{bmatrix} = \begin{bmatrix} 1\,0 & 0 & 0 & 0 \\ 0\,1 & 0 & 0 & 0 \\ 0\,0 & -\sin\theta & v\cos\theta & 0 \\ 0\,0 & \cos\theta & v\sin\theta & 0 \\ 0\,0 & 0 & 0 & -\sin\theta \\ 0\,0 & 0 & 0 & \cos\theta \\ 0\,0 & 0 & 0 & \vdots \end{bmatrix}, \tag{7}$$

which has full rank for any value of θ and $v \neq 0$, so the system is observable provided that the target vehicle is not stationary. If the target vehicle is not moving, then the estimator will not have knowledge of the targets heading, but the target position will still be observable.

A state estimate consists of an estimate mean $\hat{x}_k \in \mathbb{R}^n$, and an estimate error covariance $P_k \in \mathbb{R}^{n \times n}$. For each pursuit vehicle, the target state estimate is initialized with an assumed estimate mean, $\hat{x}_0^i$, and error covariance, P_0^i. In order to propagate the state estimate, the UKF employs *sigma points*, $x_k^{(i)}$, which are a small number of points that approximate the first two moments of the state estimate $(\hat{x}_k, P_k)$. The sigma points are computed as

$$\begin{aligned} x_k^{(i)} &= \hat{x}_k + \tilde{x}_k^{(i)} && i = 1, \ldots, 2n \\ \tilde{x}_k^{(i)} &= \left(\sqrt{nP_k}\right)_i^T && i = 1, \ldots, n \\ \tilde{x}_k^{(i)} &= -\left(\sqrt{nP_k}\right)_i^T && i = n+1, \ldots, 2n, \end{aligned} \tag{8}$$

where $\sqrt{nP_k}$ is the matrix square root of nP_k, and $(\sqrt{nP_k})_i$ is its i^{th} row. Instead of propagating the estimate mean directly, the UKF proceeds at this point by propagating each of the sigma points using Eq. 3 with $G = 0$. Denote the propagated sigma points by

$x_k^{(i)-}$. The predicted estimate mean is then recovered by computing the average of the propagated sigma points:

$$\hat{x}_k^- = \frac{1}{2n}\sum_{i=1}^{2n} x_k^{(i)-}. \tag{9}$$

The predicted estimate error covariance is computed from the propagated sigma points as

$$P_k^- = \frac{1}{2n}\sum_{i=1}^{2n}\left(x_k^{(i)-} - \hat{x}_k\right)\left(x_k^{(i)-} - \hat{x}_k\right)^T + Q_k, \tag{10}$$

where Q_k is added to the estimate error covariance to take the process noise into account. Incorporating the process noise in this way assumes that the process noise adds linearly to the system dynamics as in Eq. 3.

When new data arrives from the sensors at time step k, the propagated values of the mean state estimate and error covariance are $\hat{x}_k^-$ and P_k^-, respectively. In order to update the state estimate with this additional information, new sigma points, $x_k^{(i)+}$, are calculated using Eq. 8 to capture the statistical moments of the current state estimate. A predicted measurement is computed at this point:

$$\hat{y}_k = \frac{1}{2n}\sum_{i=1}^{2n} y_k^{(i)}. \tag{11}$$

where $y_k^{(i)} = Hx_k^{(i)+}$. The error covariance of the predicted measurement is then

$$P_k^{yy} = \frac{1}{2n}\sum_{i=1}^{2n}\left(y_k^{(i)} - \hat{y}_k\right)\left(y_k^{(i)} - \hat{y}_k\right)^T + R_k, \tag{12}$$

where R_k is added to include the effect of the measurement noise. Next, the cross covariance between the predicted measurement error covariance and the predicted state estimate is

$$P_k^{xy} = \frac{1}{2n}\sum_{i=1}^{2n}\left(x_k^{(i)+} - \hat{x}_k^-\right)\left(y_k^{(i)} - \hat{y}_k\right)^T. \tag{13}$$

The UKF gain can now be computed simply as

$$K_k = P_k^{xy}\left(P_k^{yy}\right)^{-1}. \tag{14}$$

Then the estimate mean is updated in the usual way:

$$\hat{x}_k^+ = \hat{x}_k^- + K_k\left(y_k - \hat{y}_k\right), \tag{15}$$

and the estimate error covariance is update by

$$P_k^+ = P_k^- - K_k P_k^{yy} K_k^T, \tag{16}$$

which gives the posterior target state estimate at time step k.

Consider now the sharing of information between pursuit agents. If no target data is received by one vehicle from another via communication, then the estimation proceeds as outlined above. If measurement or estimate data are received, then the UKF above must be modified to incorporate this information. The target state information transmitted by agent j contains a timestamp, t_c^j, that denotes the time step c at which the data was valid (i.e. the time at which the measurement was made or the estimate was determined). This delayed data must be slotted into the UKF at the time step t_c^j. The UKF is then re-run from that point forward to create a new estimate that is corrected to reflect the additional data.

3.2 Shared measurement data fusion

Data fusion is relatively straight-forward for the case of shared measurements between agents because the associated errors are uncorrelated. In this case, data fusion is accomplished by two correction steps applied using available data at time step c.

After Eqs. 15–16, the gain at time step c, when the received measurement $\tilde{\mathbf{y}}_c^j$ was taken, is computed as

$$K_c^i = P_c^{xy}\left(P_c^{yy}\right)^{-1}, \tag{17}$$

where P_c^{yy} is determined with Eq. 12 but with R_c^j as the measurement error covariance of the measurement associated with agent j. The state estimate corrections at time step c are then computed as

$$\begin{aligned}\hat{x}_c^{i++} &= \hat{x}_c^{i+} + K_c^i\left[\tilde{y}_c^j - \hat{y}_c^i\right],\\ P_c^{i++} &= P_c^{i+} - K_c P_c^{yy} K_c^T.\end{aligned} \tag{18}$$

This posterior estimate, $\hat{x}_c^{i++}$, is then carried through to the current time step k with the UKF algorithm outlined above and the local measurements made by agent i after time step c.

3.3 Shared estimate data fusion

The case of shared estimates is not so simple as that of shared measurements because the estimates of agent i are correlated with those of agent j through the common target process noise. As noted in [1], independence of the measurement noise sequences of the pursuit vehicles is not sufficient for independence of their estimation errors. Ideally, the estimate cross correlation should be accounted for in the fusion in order to arrive at a better fused estimate and corresponding error covariance. However, in practice determining the cross covariance between estimates derived from three or more independent sensors that are monitoring a

nonlinear system is quite difficult, and no clear way to do this computation exists at this time. Methods for determining the cross covariance between estimates for a linear system with two independent sensors exist, but they require the sharing of the estimate error covariance for all time between the two sensor estimates [1].

For fusion of estimates from two independent sensors that are observing the same linear process, the estimate cross covariance at time step k is given by the recursion

$$P_k^{ij} \triangleq E\left[\tilde{\mathbf{x}}_k^i \tilde{\mathbf{x}}_k^{jT}\right] = \left(I - K_k^i H\right)\left(F_{k-1}^i P_{k-1}^{ij} F_{k-1}^{jT} + Q_{k-1}\right)\left(I - K_k^j H\right), \tag{19}$$

where Q_{k-1} is the discrete time representation of the target process noise covariance Q_c at time step $k-1$. Then the estimate fusion is computed as

$$\hat{x}_k^f = \hat{x}_k^i - P_k^\beta\left(\hat{x}_k^i - \hat{x}_k^j\right)$$
$$P_k^f = P_k^i - P_k^\beta\left(P_k^i - P_k^{ij}\right)^T, \tag{20}$$

where the factor P_k^β is given by

$$P_k^\beta = \left(P_k^i - P_k^{ij}\right)\left(P_k^i + P_k^j - P_k^{ij} - P_k^{ji}\right)^{-1}. \tag{21}$$

This formulation gives the maximum likelihood fused estimate. An alternative formula gives the minimum mean squared error fused estimate [3].

The system considered in this paper is nonlinear, and the assumption is made that bandwidth is not sufficient to share all available data at every sensor step. These complications matter because neither the UKF gains used by agent j at every sensor event, nor the corresponding state estimates (needed to compute F_k^j), are available to agent i for computation of the cross covariance, P_k^{ij}. Furthermore, the cross covariance P_k^{ji} is not directly available to agent i either (although the assumption that $P_k^{ij} = P_k^{ji}$ is sometimes used). Computation of the cross covariance even for linear systems is not practical for distributed estimation [4].

In order to gain traction into this problem, the cross covariance will be neglected in the estimate fusion. However, some degradation in the estimate quality is expected to occur [8]. Because the estimate cross covariance is a result of the common process noise, the adversity of the effect of neglecting the cross covariance will depend on the magnitude of the process noise. When the cross covariance is neglected, the resulting fused estimates in Eq. 20 are arrived at equivalently by both minimum mean squared error and maximum likelihood formulations, and are written as

$$\hat{x}_k^f = \hat{x}_k^i - P_k^i\left(P_k^i - P_k^j\right)^{-1}\left(\hat{x}_k^i - \hat{x}_k^j\right)$$
$$P_k^f = P_k^i\left[I - \left(P_k^i - P_k^j\right)^{-1}\right]. \tag{22}$$

Recall that in Section 2, communicated estimates were assumed to be delayed by one sensor time-step, so that the estimate fusion takes place on the receiving agent, and uses both local data and communicated data that is one sensor time step old. Furthermore, which information to use in communication, and which estimate to use in the estimate fusion, must be determined. Because of the neglection of the cross covariance, the estimate sent from one agent to the others should be one that has not been combined with information from the other agents. In this way, the effect of neglecting the cross covariance in the estimate fusion can be mitigated to some degree. The term a *clean estimate* will be used for this case. On the other hand, if the assumption of negligible cross covariance turns out to be fair, then the fused estimates should continue to be combined with incoming data. The term *combined estimates* will be used for this case. In either case, unfused estimates (i.e. ones that were created without the use of any information from communication) will be transmitted when a pursuit agent communicates its estimate data.

4 Coordinated target pursuit

Because of the limitations of the target tracking sensors, such as finite range and inability to sense through obstacles, the target pursuit vehicles must remain within sensor range of the target in order to make measurements of its position. Furthermore, because the environment in which the target is moving could include "hostile entities" and sensor or vehicle obstructions, target tracking would be best served by distributing the sensors in some fashion such that they each have a unique view of the target, and such that the group as a whole is less vulnerable. The term *coordinated target pursuit* (CTP) is used here to denote the target pursuit aspect of the coordinated target tracking task.

In this paper, coordinated target pursuit is accomplished by a behavioral control algorithm in which pursuit control is combined with inter-vehicle spacing controls and obstacle avoidance behaviors. The idea behind the algorithm is to maintain an even spatial distribution of agents in the group with some nominal relative inter-vehicle spacing, while simultaneously

bringing the center of the group to the target. The CTP algorithm amounts to a weighted sum of control terms:

$$u^i_{\text{steer}} = (1-w)\left(u^i_{\text{ct}} + u^i_{\text{space}}\right) + u^i_{\text{avoid}}$$
$$u^i_{\text{speed}} = 0, \tag{23}$$

where u^i_{ct} is the centroid to target control, u^i_{space} affects intervehicle spacing, and u^i_{avoid} is the obstacle avoidance term. All terms are assumed to be at time step k unless otherwise noted. The controls in the weighted sum have first been normalized to lie within $[-1, 1]$. The weighting, $w = |u^i_{\text{avoid}}|$, ensures that obstacle avoidance takes precedence in the overall target pursuit behavior. Each of these control terms is explained below.

4.1 Centroid to target control

The component of the tracking behavior responsible for bringing the group centroid to the target is given for each pursuit vehicle as

$$u^i_{\text{ct}} = k_{\text{ct}} \sin\left(\theta_{\text{ct}} - \theta^i\right), \tag{24}$$

where θ_{ct} is the angle of the vector from the group centroid to the target (assumed known), and k_{ct} is the centroid to target control gain.

The angle θ_{ct} is computed using data from all vehicles in the group and the target. Because of the broadcast network, data from inter-vehicle communications is held until the next communication update. However, each agent has current information for its own state so that each agent computes a slightly different group centroid as

$$\begin{bmatrix} \bar{x}^i_k \\ \bar{y}^i_k \end{bmatrix} = \frac{1}{N}\left(\sum_{j\neq i}^{N} \begin{bmatrix} x^j_{\text{b}} \\ y^j_{\text{b}} \end{bmatrix} + \begin{bmatrix} x^i_k \\ y^i_k \end{bmatrix}\right), \tag{25}$$

and

$$\theta^i_{\text{ct}} = \tan^{-1}\left(\frac{\bar{y}^i}{\bar{x}^i}\right), \tag{26}$$

where subscript b denotes the most recent time step in which data was broadcast.

4.2 Inter-vehicle spacing

Inter-vehicle spacing control is based on the work in [6] and is defined as

$$u^i_{\text{s}} = \sum_{j\neq i}^{N}\left(1 - \left(\frac{r_{s0}}{d^{ij}}\right)^2\right)\sin\left(\theta^{ij} - \theta^i\right). \tag{27}$$

Here, $\theta^{ij} \in [-\pi, \pi)$ is the angle of the vector from agent i to agent j, $d^{ij} \in \mathbb{R}^+$ is the distance between them, and r_{s0} is the nominal desired inter-vehicle spacing. The sinusoidal term determines the direction in which the vehicle will steer.

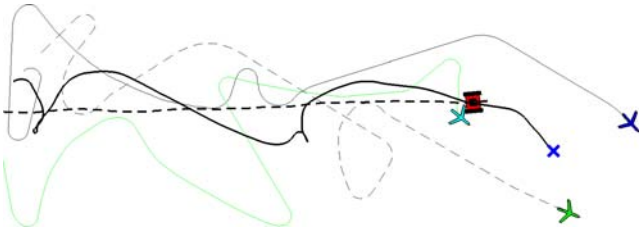

Fig. 5 Typical coordinated target pursuit behavior with $N = 3$, and target vehicle speed roughly half that of the pursuers. The "X" and its *dark solid line* shows the trajectory of the pursuit vehicle group centroid

Figure 5 shows typical results of a simulation run with the coordinated pursuit and intervehicle spacing control behaviors. In the figure, the pursuit vehicle group centroid is denoted by the "X" and the target is represented by the tank. The centroid trajectory attempts to follow the target, while the pursuers maintain intervehicle spacing according to Eq. 27.

4.3 Obstacle avoidance

Because targets often move among a background of other objects, the pursuit vehicles should be able to track a target vehicle moving in a cluttered environment in which the obstacles not only impede sensor views, but also restrict the pursuit vehicle trajectories. Therefore, an additional behavioral element is required that will allow the pursuit vehicles to avoid obstacles while they follow the target. The pursuit vehicles do not interact with the obstacles unless they are within a threshold radius, $r_{\text{oT}} > r_{\text{o}}$, which is depicted in Fig. 4b. The obstacle avoidance control is computed as

$$u^i_{\text{oa}} = \begin{cases} 0 & \text{if } d^i_{\text{oj}} > r^j_{\text{oT}}, \\ f_{\text{oa}}\left(\Delta\theta^i_{\text{oj}}, d^i_{\text{oj}}\right) & \text{otherwise.} \end{cases} \tag{28}$$

Here, superscript i refers to the i^{th} vehicle, *oa* implies "obstacle avoidance," and superscript j refers to the j^{th} obstacle. The obstacle avoidance function in Eq. 28 is given by

$$f_{\text{oa}}\left(\Delta\theta^i_{\text{oj}}, d^i_{\text{oj}}\right) = \sum_{j\in\mathcal{N}_{\text{o}}} \cos\left(\Delta\theta^i_{\text{oj}}\right)\text{sgn}\left(\Delta\theta^i_{\text{oj}}\right) f_d\left(d^i_{\text{oj}}\right), \tag{29}$$

where $\Delta\theta^i_{\text{oj}} = \theta^i_{\text{oj}} - \theta^i$, and θ^i_{oj} is the angle of the vector from vehicle i to the center of obstacle j. The obstacle

distance function, which determines the magnitude of the obstacle avoidance control in Eq. 29, is given by

$$f_d\left(d^i_{\mathrm{oj}}\right) = -2\frac{\left(r^2_{\mathrm{oT}} - r^2_{\mathrm{o}}\right)^2}{\left(r^2_{\mathrm{o}} - d^{i2}_{\mathrm{oj}}\right)^2}. \tag{30}$$

The obstacle avoidance control causes a pursuit vehicle that has entered the threshold radius of a particular obstacle to turn away from that obstacle until it is heading perpendicular to a vector from the center of the object. Under this control, a pursuit vehicle will circle the obstacle toward which it is drawn by the other parts of the pursuit control, rather than be repelled directly away from it. The proof that this control acting alone on a single vehicle will prevent collision with an obstacle, provided that the initial position of the vehicle is not too close to the obstacle, is a simple matter of geometry. However, acting in concert with the other controls and the full group of pursuit vehicles, proving obstacle avoidance is more difficult.

5 Results

Initial simulations were conducted without pursuit agent dynamics and without obstacles in the environment. The experiments were designed to isolate the behavior of the target tracking estimator in order to study the effects of intervehicle communication and the precision and reliability of the sensors, which were fixed for the duration of these tests. These simulations were conducted with $N = 3$ pursuit vehicles, and were run until the mean (over both time and N) log likelihood of the target states neared steady state. Matlab was used as the numerical simulation tool. After establishing the basic behavior of the estimator as described above, subsequent experiments involve pursuit vehicle dynamics in conjunction with a cluttered environment. In all simulations, the target vehicle runs with zero control on heading and speed, but with random zero-mean process noise, with standard deviations $\sigma_v = 0.05$, $\sigma_\phi = 0.1$, acting as inputs. While the resulting heading and heading rate of the target are unconstrained, the speed of the target was constrained to lie between 50% and 150% of its initial value which was half that of the pursuit vehicles.

In the figure legends, *No Comm* refers to data that was generated without communication of target data between agents. *Meas i*, where $i = 1, 2, 3, 4, 5$, refers to the maximum number of measurements that could be shared between pursuit vehicles. *Fus A* and *Fus B* refer to data that was computed using clean estimates or combined estimates respectively. *Centralized* refers to fully centralized estimate data in which estimates were computed using all measurement information available to the pursuit agents without delays.

5.1 Isolated estimator experiments

5.1.1 Sensor reliability

The first simulation examines the effects of the quantity and form of the data shared by communication. These experiments were run for a range of fixed sensor reliability rates from 10% up to 100%. The simulation was run for the data transmission cases of: (1) no communication, (2) one to five of the most recent measurements broadcast, (3) the most recent estimate broadcast and fused with clean estimates at the receiving agent, (4) most recent estimate broadcast and fused with the combined estimates at the receiving agent, and (5) fully centralized estimates. The fully centralized estimates fuse all the measurements available from all agents and without any data delays, and are used as to bound the results of the other data fusion methods. The metrics used in this study to measure and compare estimate performance are the average log likelihood of the distributed estimates and the integrated error in the target position estimates.

As shown in Figs. 6 and 7, estimate performance improves with increasing sensor reliability, and with increasing communication data. Including more than two or three measurements give diminishing returns in terms of improvement in estimate likelihood or integrated error because the additional data is substantially delayed so that it does not contribute to a significant improvement in the accuracy of the estimates at the current time. Estimate fusion outperforms measurement fusion, but requires more data be shared by communication than measurement fusion, as noted earlier. The performance of the estimate fusion method may have been helped somewhat by the constraint on target speed, so that its motion is more self-correlated than the estimate models presume.

5.1.2 Sensor precision

Cursory studies indicated that equal changes in the precision of all the sensors of the group of pursuit vehicles would simply shift the overall estimate performance. However, the effect of changing the relative precision of the sensors indicates that a more cost-effective improvement in estimate performance can be achieved by improving the relative precision of the sensors on a subset of the group. In this case, the sensor precision of only one agent was varied while the other two remained

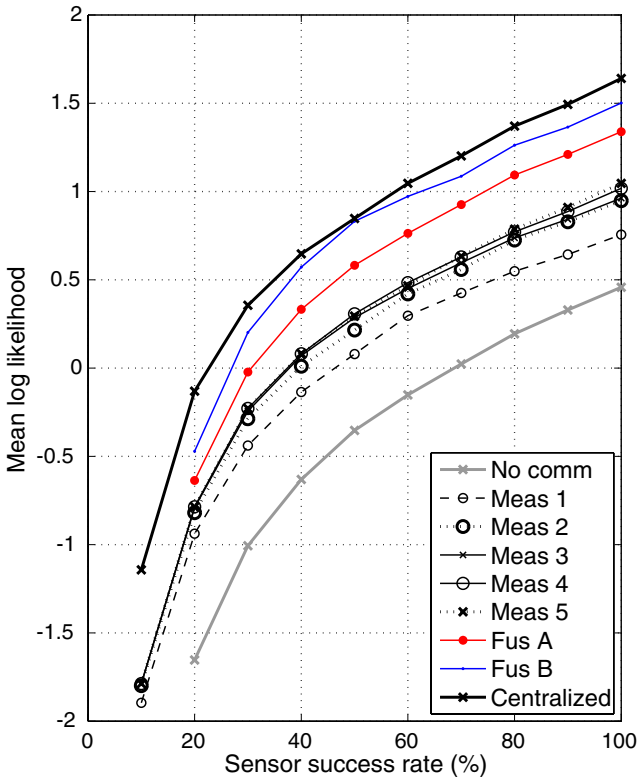

Fig. 6 Sensor performance measured by the mean log likelihood of the estimates as a function of sensor reliability. All pursuit agents have the same sensor precision

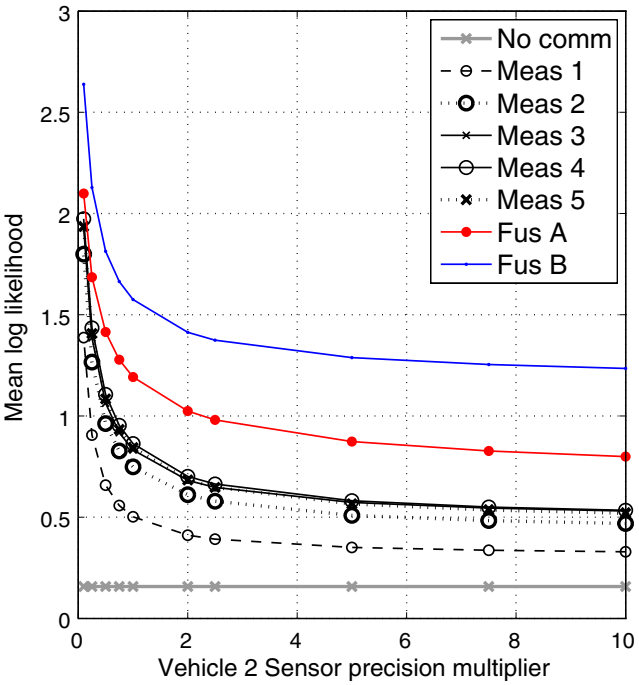

Fig. 8 Effects of changes in relative sensor precision of a one agent on the estimator performance of the other two agents. This figure shows the estimate results for one of the fixed measurement covariance agents, which have $[\sigma_x\ \sigma_y] = [9\ 9]$. Sensor precision on the *lower axis* indicates the factor by which sensor precision was increased. Sensor reliability is fixed at 70%

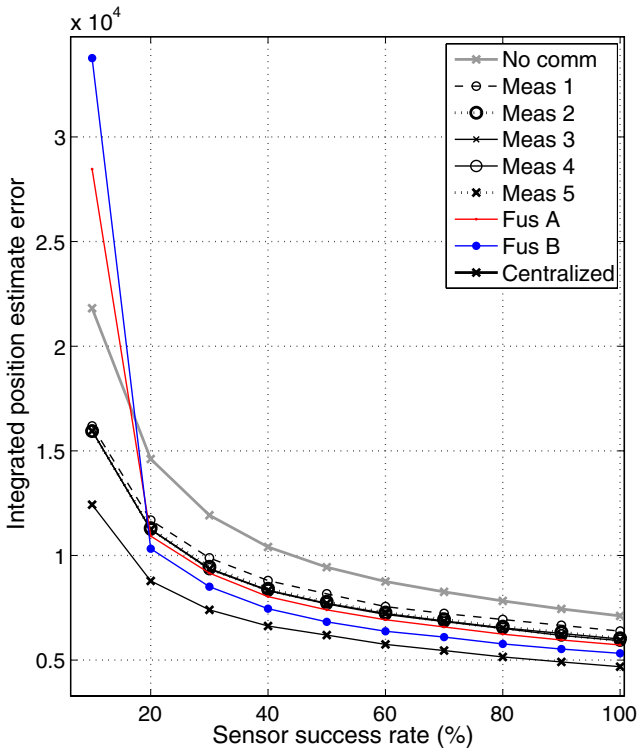

Fig. 7 Sensor performance measured by integrated position estimate error. All pursuit agents have the same sensor precision

constant. The precision of a single agent was increased by dividing its measurement covariance matrix, R^i, by a constant factor. Two cases of sensor reliability, at 30% and 70%, were run.

In Fig. 8, sensor precision is increased by dividing the measurement noise covariance by the factor indicated on the lower axis of the graph. The figure demonstrates how increasing the measurement precision of a single agent in the group, coupled by communication of tracking data, improves the overall mean estimator performance.

When coupled with lower sensor reliability of all pursuit agents, increases in sensor precision for a single pursuit vehicle results in an even greater relative improvement on mean estimator performance. This improvement is apparent by comparing the no communication cases in Fig. 8 with the measurement communication cases in the same figure.

5.1.3 Communication period

Simulation experiments were conducted in order to observe the effects of relative differences between sensor and communication periods. For the sake of simplicity, the sensor period was fixed at one time unit,

and the communication period was varied as an integer multiple of the sensor period. In general, and as expected, decreased communication periods improved target state estimates. This result is shown in Figs. 9 and 10. The only case in which a decrease in communication period did not improve the target estimate results was for the case of recombined estimate fusion (Fus B in Figs. 9 and 10). This condition was true only at the highest communication rates, in which the lack of accounting for the estimate cross covariance appears to have become detrimental to the performance of the estimate fusion. This situation was not true for the clean estimate fusion (Fus A in Figs. 9 and 10). Figure 11 shows the trend in integrated estimation error in the (x,y) planar position of the target vehicle for 70% sensor reliability with communication period. Higher likelihoods in Fig. 9 correspond to lower integrated errors in Fig. 11.

5.2 Pursuit vehicle coordination

Because the target pursuit algorithm is behavioral, the various elements that compose the overall behavior can be added or omitted depending on the circumstances.

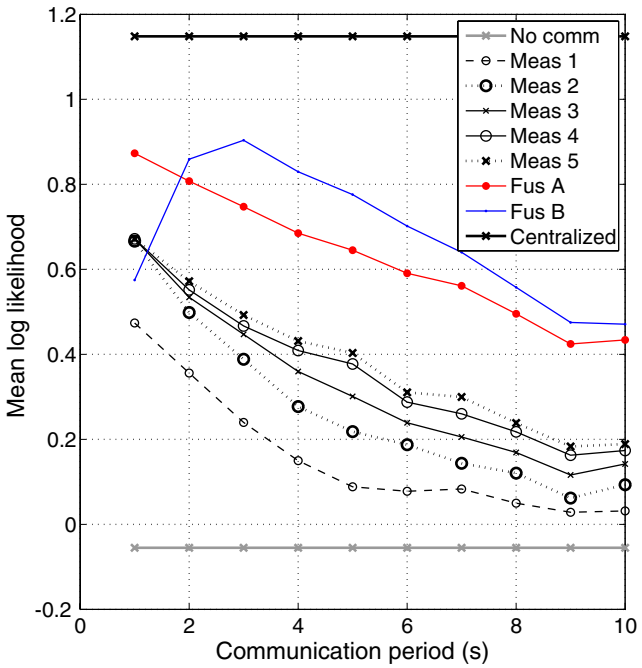

Fig. 9 Estimator performance dependence on communication period for sensor reliability 70%. The sensor measurement period is fixed at $T_b = 1$. Obviously, the No communication case, and the Centralized estimate case are unaffected by communication. Drop-off in mean likelihood at low communication periods for the Fus B case indicates effects of the neglected cross-covariance

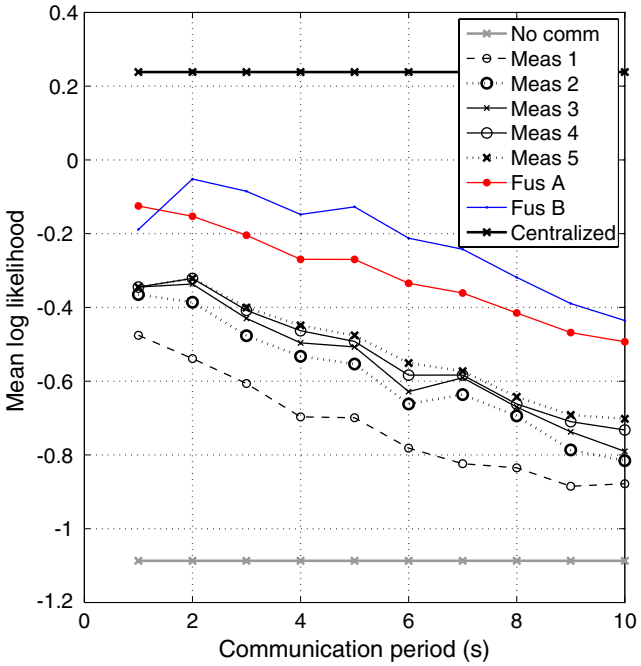

Fig. 10 Estimator performance dependence on communication period for sensor reliability 30%. The sensor measurement period is fixed at $T_b = 1$. Drop-off in mean likelihood at high communication rates for Fus B case is similar to that for the sensor reliability situation shown in Fig. 9

In this subsection, the pursuit trajectories under different behavior compositions are evaluated for an example scenario.

5.2.1 No coordination

If no detriment exists to having the individual pursuit vehicles fly over the target, and no problems for them to overlap in the $x - y$ plane (i.e. due to forced altitude separation), then the centroid to target control, with each agent acting as its own group centroid, should be used. This control has the advantage that it requires the least communication between group members. Trajectories are shown in Fig. 12 in which the obstacles are formed together to create long barriers to sensing and movement. In this scenario, the uncoordinated target pursuit control can lead to situations in which the pursuit vehicles often clump up in a manner that is detrimental to target tracking and re-acquisition of a lost target. Furthermore, this method of pursuit would not be suitable to target tracking if the vehicles need to be separated in the plane for some reason (i.e. they are easier to identify by threats, or if the sensors perform better when they are at separated vantage points).

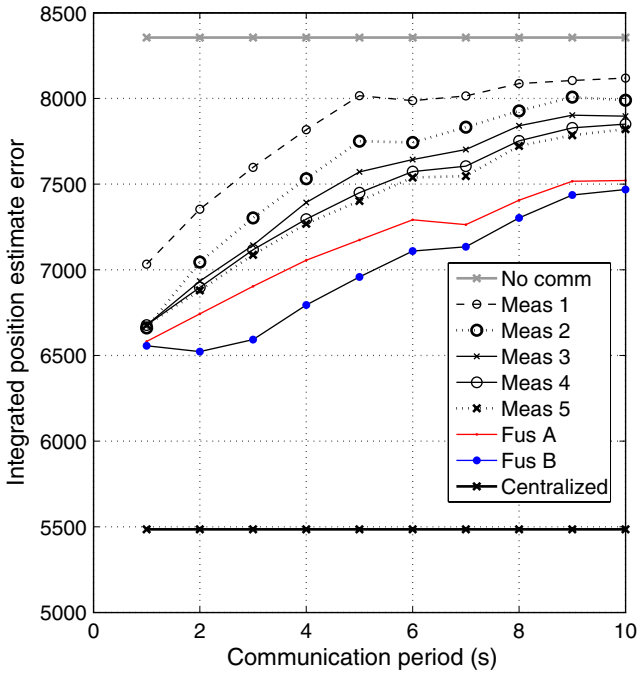

Fig. 11 Estimator performance dependence on communication period for sensor reliability 30%. The sensor measurement period is fixed at 2. Drop-off in mean likelihood at high communication rates for Fus B case similar to that for the sensor reliability situation shown in Fig. 9

5.2.2 Inter-vehicle spacing with CT control

Pursuit vehicle behavior with intervehicle spacing and group centroid to target control will help to distribute the pursuit vehicles more evenly around the target. This

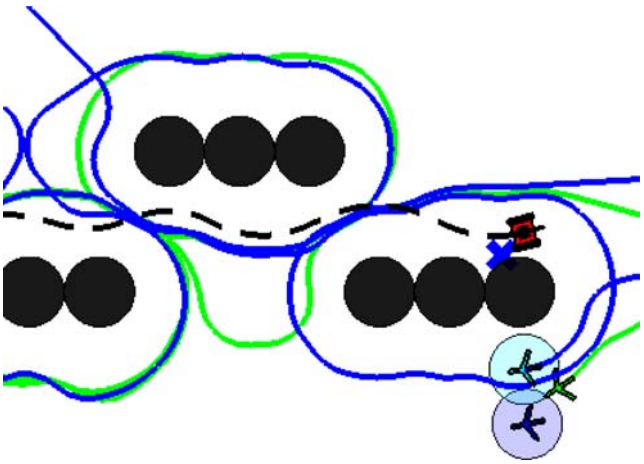

Fig. 12 Pursuit vehicle trajectories under the uncoordinated tracking control. *Shaded circles* about an agent indicate loss of sensor view of the target. *Lines following the vehicles* indicate trajectory history. Pursuit vehicles can become clumped together so that they not only lose view of the target, they will also be in a poor position to find the target if it changes direction

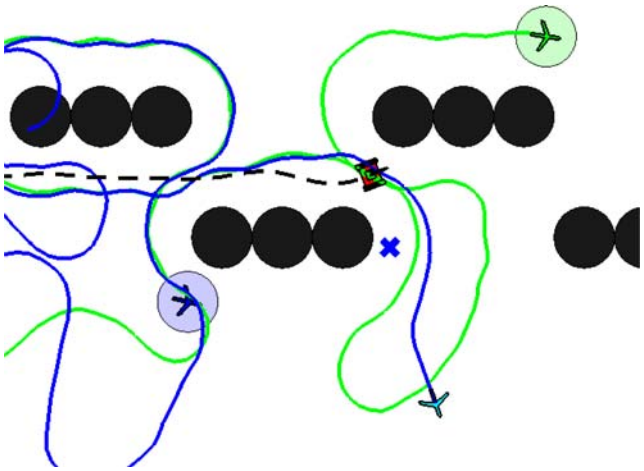

Fig. 13 Pursuit vehicle trajectories when the vehicles are coordinated. Shaded circles about an agent indicate loss of sensor view of the target. Lines following the vehicles indicate trajectory history. Compared to the uncoordinated case, the vehicles are in superior positions to maintain target view, as well as to re-acquire the target should it be lost

situation can be seen in Fig. 13. With this control, each vehicle tends to stay in its own standoff region around the target. While the individuals in the group do not get as close to the target as in the uncoordinated case, they do end up at better distributed vantage points from which to view the target. This situation leads to improved estimates when the obstacles form long barricades as is shown in the figure.

The results in Fig. 14 show how the sharing of data between agents improves their estimates when

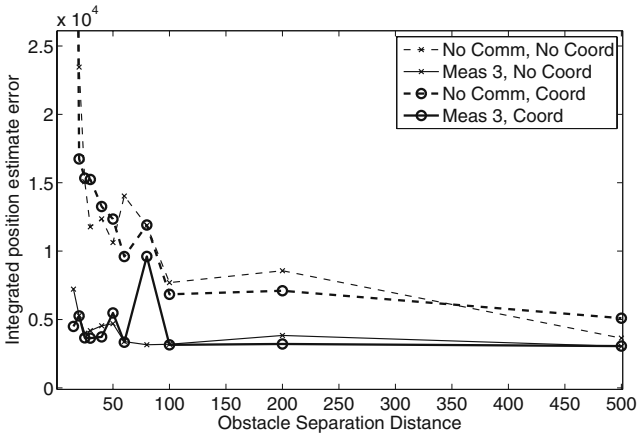

Fig. 14 Target tracking performance for the three communication cases with variable obstacle separation distance. The legend entries are: (a) no communication, (b) transmit last measurement, (c) transmit last three measurements, all with uncoordinated pursuit, while (d), (e) and (f) are the same, respectively, but with coordinated pursuit. Case (g) represents the centralized estimate

obstacles in the environment can block the pursuer's views of the target. The figure shows how the integrated position estimate error varies with obstacle spacing. In general, and as expected, the closer the obstacles in the environment are spaced, the more they impeded the pursuit vehicle views of the target. Use of communication between pursuit agents overcomes this difficulty because the pursuit vehicles maintain better knowledge of the target position and are able to follow the target even though it is not in view. Further, while the coordinated pursuit algorithm did not lead to much improvement in the overall estimate results over the uncoordinated case, it did not degrade the results either. The coordination can be used to good effect in the right situation, such as when spreading the pursuit vehicles out around the target is more advantageous than allowing them to travel directly over it.

6 Conclusions and future work

The results presented here show how communication of target state information can be used to improve estimates in the coordinated target tracking system. The results also show how pursuit vehicle coordination can be used to advantage when the target is moving in an environment that is cluttered with obstacles that impede both the trajectories of the pursuit vehicles and their sensor views of the target.

Increases in the amount of information used always improves the accuracy of the estimates for all of the pursuit vehicles. However, when measurements were shared between pursuit agents, the results demonstrated that minimal improvement in the estimates was gained when more than three measurements were shared. This result is due to the delay in the older measurements, which makes the older data less beneficial to improvements in estimate quality. On the other hand, sharing of estimates results in superior results. The estimates are only ever delayed by the minimal communication delay of the system, and contain the broadcasting agent's best knowledge of the target state at all times, which includes the combination of all of the broadcasting agent's measurements. Sharing of estimates takes about the same amount of data per communication step as does sharing five measurements, so the benefits of estimate fusion need to be weighed against the increase bandwidth requirements.

Increases in sensor reliability were shown to be beneficial to estimator performance. However, for a given sensor reliability rate, improvements in the sensor precision of a single agent in the group can have a significant positive impact on the estimator performance. Increases in communication frequency were also demonstrated to improve estimate results.

When the target tracking system is run in a cluttered environment using the behavioral coordinated pursuit algorithm outlined in the paper, the same effects of communication on target state estimator performance were observed. Which behaviors to emphasize in the coordinated target pursuit was shown to depend on the manner in which the obstacles in the environment are arranged. Whichever pursuit algorithm is used, communication of target state information between pursuit agents significantly improves estimate accuracy.

Many aspects of this study could be explored in future work. In particular, approximation of cross-covariance in the estimate fusion, asynchronous communication and sensing, other clutter/obstacle models, and targets operating under detection avoidance behaviors, are particularly relevant and interesting.

Additionally, a comparison study of different Coordinated Target Pursuit algorithms and their performance dependence on the number of pursuit vehicles in the system would help provide insight into the sort of behaviors that are beneficial to a CTT system.

Acknowledgements This work was supported by NSF grant CMS-0238461 and AFOSR grants FA9550-05-1-0430 and FA9550-07-1-0528.

References

1. Bar-Shalom Y, Li XR (1995) Multitarget-multisensor tracking: principles and techniques. Yaakov Bar-Shalom, Storrs
2. Campbell ME, Whitacre WW (2007) Cooperative tracking using vision measurements on seascan UAVs. IEEE Trans Control Syst Technol 15(4):613–626
3. Chang KC, Saha RK, Bar-Shalom Y (1997) On optimal track-to-track fusion. IEEE Trans Aerosp Electron Syst 33(4):1271–1276
4. Chen H, Bar-Shalom Y (2006) Track fusion with legacy track sources. In: 9th international conference on information fusion, 2006. ICIF'06, pp 1–8
5. Eickstedt DP, Benjamin MR (2006) Cooperative target tracking in a distributed autonomous sensor network. In: OCEANS 2006, pp 1–6, September
6. Justh EW, Krishnaprasad PS (2004) Equilibria and steering laws for planar formations. Syst Control Lett 52:25–38
7. Klein DJ, Morgansen KA (2006) Controlled collective motion for trajectory tracking. In: Proc of the 2006 American control conference, Minneapolis, MN, June
8. La Scala BF, Farina A (2002) Choosing a track association method. Inf Fusion 3(2):119–133
9. Nettleton EW, Durrant-Whyte H (2001) Delayed and asequent data in decentralised sensing networks. In: Proc SPIE Conf, vol 4571, pp 255–266

10. Olfati-Saber R (2007) Distributed kalman filtering for sensor networks. In: Proc. of the 46th conference on decision and control, December
11. Simon D (2006) Optimal state estimation: Kalman, H infinity, and nonlinear approaches. Wiley-Interscience, New York
12. Triplett BI, Klein DJ, Morgansen KA (2007) Distributed estimation for coordinated target tracking in a cluttered environment. In: Proc. of Robocomm, Athens, Greece, October
13. Wheeler M, Schrick B, Whitacre W, Campbell M, Rysdyk R, Wise R (2006) Cooperative tracking of moving targets by a team of autonomous uavs. In: 25th digital avionics systems conference, 2006 IEEE/AIAA, pp 1–9, October

Benjamin I. Triplett received his Ph.D. from the Department of Aeronautics and Astronautics at the University of Washington in Seattle, WA in June 2008. He received his B.A.Sc. degree in Engineering Physics in 2000 and his M.A.Sc. degree in Mechanical Engineering in 2002 from the University of British Columbia, Vancouver, British Columbia. He is currently working as a consultant for Sector 7G Systems. His research interests include distributed nonlinear estimation and coordinated control of nonholonomic vehicles for cooperative target pursuit.

Daniel J. Klein received the Ph.D. and M.S. degrees from the Department of Aeronautics and Astronautics at the University of Washington, Seattle in 2008 and 2005, respectively. He received the B.S. in Mechanical Engineering with Honors in Research degree from the University of Wisconsin, Madison, WI, in 2003. He spent the summer of 2007 working at the Intel Research Lab in Seattle, Washington. Daniel is currently a postdoctoral scholar with the Department of Electrical and Computer Engineering at the University of California, Santa Barbara. His research interests include robotics, coordinated control, computer vision, and distributed estimation.

Kristi A. Morgansen received the B.S. (*summa cum laude*) and the M.S. in Mechanical Engineering from Boston University, Boston, MA, respectively in 1993 and 1994, and the S.M. in Applied Mathematics and Ph.D. in Engineering Sciences respectively in 1996 and 1999 from Harvard University, Cambridge, MA.

After receiving the Ph.D. degree, she was first a postdoctoral scholar then a senior research fellow in Control and Dynamical Systems and Mechanical Engineering at the California Institute of Technology, Pasadena, CA. In August 2002, she joined the faculty of the Department of Aeronautics and Astronautics at the University of Washington where she is currently an assistant professor.

From 2002 to 2007, Professor Morgansen held the chaired position of Clare Boothe Luce Assistant Professor of Engineering at the University of Washington. She received an NSF CAREER Award in 2003.