Physica A 509 (2018) 1195–1206

Contents lists available at ScienceDirect

Physica A

journal homepage: www.elsevier.com/locate/physa

Distributed dynamic state estimation with flocking mobile agents

Duosi Xie [a], Shouxu Zhang [b], Jianquan Xie [c,*]

[a] College of Information Science and Technology, Central South University, Changsha, Hunan, 410083, China
[b] School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an, Shaan'xi, 710072, China
[c] Research Institute of Information Security, Hunan University of Finance and Economics, Changsha, Hunan, 410205, China

ARTICLE INFO

Article history:
Received 20 January 2018
Received in revised form 31 March 2018
Available online 13 June 2018

Keywords:
Distributed estimation
Flocking
Networked systems

ABSTRACT

In this paper, a distributed dynamic state estimation problem with networked flocking mobile agents to actively sense the target, is studied. By combining the distributed estimation strategy proposed in [1] and the well known flocking control algorithm, an innovative algorithm is derived. In the proposed algorithm, each agent only uses the local information, estimates not only its own state, but also the target state. On the top of those, each agent designs its own control algorithm, and eventually physically track the target in the sense that: (i) it keeps the target in its sensing range while avoid collision with the target or any other agent; (ii) it moves along with the target to maintain a better collective observability. The whole group of agents eventually flocks with the target. With the communication topology becoming complete and each agent seeing the target, the centralized Kalman filter solution is analytically shown to be recovered. Simulations are used to illustrate the proposed algorithm.

1. Introduction

The problem of state estimation using networked local sensors [1–13] has become popular in recent years. The task is for a set of networked local sensors to use scalable peer-to-peer communication schemes in replacement of the traditional fusion-center-based communication scheme. Such a replacement aims at increasing the robustness against unexpected local node/link failures. Early algorithms including the consensus-based algorithms [2,3], which in general require multiple communication iterations during each sampling time interval; and the diffusion-based algorithms first proposed in [4] and later improved in [5]. See comprehensive comparisons in [1,8]. On top of the early results, newly proposed algorithms in recent years focus on either milder convergence conditions [1,6,9], or extended scenarios such as nonlinear process/sensing models [7,11] and uncertain process models [11–14]. It is worth mentioning that the above literatures consider the framework of discrete-time system, which is the framework considered in this paper. The readers are referred to [15,16] for related work in the framework of continuous-time system or [17–19] for the framework of hybrid dynamic system. Other work related to the topic of large-scale networks can be found in [20–22].

The flocking behavior is commonly observed in nature. Examples include migration of birds, schooling of fish and swarming of bacteria [23–28]. From the control's point of view, it is described as a cooperative behavior such that multiple "agents" use information exchanged among each other as well as certain rules (e.g., control laws), to jointly achieve a mutual task [29–34]. Three rules of the flocking system [23] are introduced as follows: (1) Flock centering – attempting to stay

* Corresponding author.
E-mail addresses: xieduosi_89@126.com (D. Xie), zhangshouxu@nwpu.edu.cn (S. Zhang), xiejianquan@sina.com (J. Xie).

https://doi.org/10.1016/j.physa.2018.05.146

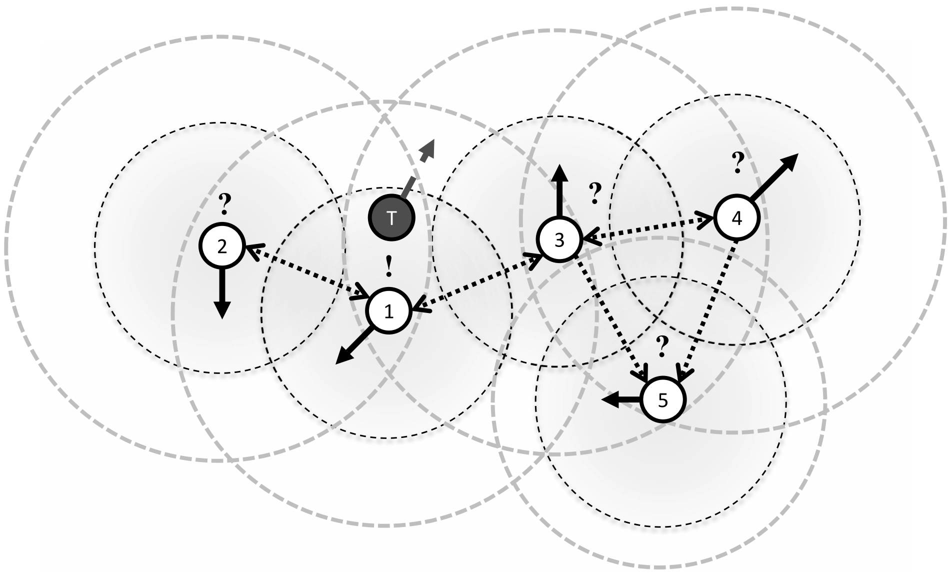

Fig. 1. Active target tracking using flocking mobile agents.

close to nearby agents; and (2) Collision avoidance – avoiding collisions with nearby agents; and (3) Velocity matching – attempting to match velocity with nearby agents. The author in [29] proposes a theoretical framework for designing and analyzing distributed flocking algorithms based on the Reynolds' model. It is considered therein both the scenarios with and without obstacle avoidance. In [32,35,36], the flocking algorithms with a virtual leader for goal seeking problem are investigated. A flocking system with group leader is considered in [37], where the followers do not even know which the leaders are. Methods to select these leaders are proposed in [30,38]. In [39], the authors propose algorithms with partial sense of the virtual leader's information. A more generalized algorithm is introduced in [40] based on the self-driving results in [29]. Moreover, the flocking problem is widely studied from many aspects including flocking with pinning control [41], flocking in noisy environment [42], flocking with time varying delays [43], flocking with parameter uncertainties [44,45], flocking with external disturbances [33] and flocking with information weighted Kalman filter [46].

In the aforementioned framework of distributed estimation algorithms, it is usually assumed that all local agents are static and hence know their own positions perfectly. On the other hand, the framework of flocking problems usually assume perfect knowledge of the neighboring agents' state if obtained. However, this is almost impossible in more realistic scenario with the sources of noise everywhere. Motivated by these facts, this paper aims at combining these two famous problems, and investigating the scenario in which the mobile agents estimate the target by moving along with the target and keeping it in the sensing range. Such a strategy is motivated by the intuition that with more mobile agents physically following the target, the measurements of the target can be obtained by more local agents and hence lead to a better overall estimation performance in the network. Based on the unified framework of distributed estimation proposed in [1], referred therein as the *distributed hybrid information fusion* (DHIF) algorithm, we show that the DHIF algorithm actually recover the centralized Kalman filter if the communication topology between local agents is complete.[1] Motivated by this, we adopt the flocking control algorithm, and drive each agent to chase the target of interest and to try to keep it within its own sensing range. Simulation is used to illustrate the effectiveness of proposed algorithm.

2. Problem formulation

The problem of interest in this paper is described in the following along with Fig. 1. Suppose that a group of networked moving agents (circles marked with numbers) with sensing equipments and limited sensing range, are used to track the state of a moving target. Besides the sensing capability, each agent is able to send information to the agents in its communication range. Note that the communication ranges of all agents might be different. This could caused by different communication capacities due to different transmission scheme of each agent. As a result, the directed communication topology is formulated as well. Note that as agents move around, the topology changes over time.

At the initial time instant, each agent is not required to be able to directly observe the target. That is, the target might be in the sensing range of only a subset of agents (denoted with exclamation points) while the rest agents have no idea of where the target is (denoted with question marks). By locally sensing the target (if in its sensing range) and communicating with

[1] A graph is complete if every agent can receive information from every other agent.

the neighbors, each agent tries to estimate the state of the target, and try to physically track and keep the target in its sensing range by moving along with it. By physically tracking the target, the entire group should act like a flock around the moving target and keep as many agents directly observing the target as possible. Meanwhile, agents should avoid collisions with the target or with each other. During this entire process, the agents are only exchanging information with their neighbors.

(Notations). Denotes the $I_{n\times n}$ an $n \times n$ identity matrix. Denotes the $0_{n\times n}$ an $n \times n$ zero matrix. Denotes the $\mathrm{sgn}(\cdot)$ the element-wised sign function. The notation $a \sim \mathcal{N}(m_a, R_a)$ implies the random variable $a \in \mathbb{R}^n$ is normally distributed with mean $m_a \in \mathbb{R}^n$ and covariance $R_a \in \mathbb{R}^{n\times n}$. If a random variable a is uniform distributed from b and c with $b < c$, it is denoted as $a \sim \mathcal{U}(b, c)$. Denotes $\mathrm{col}\{A_i\}_{i\in S}$ a vector or matrix obtained by stacking A_i, $i \in S$ for some index set S. Denotes $\mathrm{blkdiag}\{A_i\}_{i\in S}$ a block diagonal matrix with its diagonal blocks being A_i, $i \in S$ for some index set S.

2.1. Models

Hereafter in this paper, the target and the agents are both assumed as particles moving in the 2D plane, where the target is assumed to have constant velocity and the agents are assumed to be able to control the accelerations along x and y directions. Let $\mathcal{V} \triangleq \{1, \ldots, N\}$ be the index set of all local agents. Let $X_0[k] \in \mathbb{R}^4$ and $X_i[k] \in \mathbb{R}^4$, $i \in \mathcal{V}$ be the state vectors of, respectively, the target and agent i, $i \in \mathcal{V}$. The state vector contains, namely, the displacements along x and y directions, and the velocities along x and y directions, denoted respectively as, $p_i[k] \in \mathbb{R}^2$ and $v_i[k] \in \mathbb{R}^2$. Specifically, $X_i[k] = [p_i^\top[k], v_i^\top[k]]^\top$, for each $i \in \mathcal{V} \cup \{0\}$. The dynamics of the target and the agents can be written as,

$$X_i[k+1] = FX_i[k] + BU_i[k] + w_i[k], \tag{1}$$

with $w_i[k]$ being the process noise and $U_i[k] \in \mathbb{R}^2$ being the control input of agent i at time instant k. Note that $U_0[k] = 0$, $\forall k$ by the setup. Moreover, let $\Delta T > 0$ be the sampling time interval and let F and B have the form of

$$F = \begin{bmatrix} I_{2\times 2} & \Delta T I_{2\times 2} \\ 0_{2\times 2} & I_{2\times 2} \end{bmatrix},\ B = \begin{bmatrix} 0_{2\times 2} \\ \Delta T I_{2\times 2} \end{bmatrix}.$$

It is assumed that on one hand, at each time instant k, each agent has the access to a measurement[2] about its own global position, denoted as $z_i[k]$, i.e.,

$$z_i[k] = HX[k] + v_i[k],\ i \in \mathcal{V},$$

where $H \triangleq [I_{2\times 2}, 0_{2\times 2}]$ is the set of indices of all agents hereafter. It is assumed that $v_i[k] \sim \mathcal{N}(0_{2\times 1}, R_i[k])$, where $R_i[k]$ is the covariance matrix associated with $v_i[k]$. On the other hand, it is assumed for each agent that if the target is in its sensing range, it is able to measure the relative position of the target with respect to itself, denoted as $z_i^0[k]$, i.e., for each $i \in \mathcal{V}$,

$$z_i^0[k] = H(X_0[k] - X_i[k]) + v_i^0[k],$$

where $v_i^0[k] \sim \mathcal{N}(0_{2\times 1}, R_i^0[k])$ with $R_i^0[k]$ being the covariance matrix associated with $v_i^0[k]$.

2.2. Assumptions

Some assumptions made through this report are discussed before moving to the next section. For simplicity, the time instant k for all variables are ignored in this subsection.

First of all, all noises are assumed to be white gaussian with zero means. Moreover,

$$\begin{aligned}
\mathbb{E}[w_i w_j^\top] &= \delta(i,j)Q_i \in \mathbb{R}^{4\times 4},\ \forall i,j \in \mathcal{V} \cup \{0\},\\
\mathbb{E}[v_i v_j^\top] &= \delta(i,j)R_i \in \mathbb{R}^{2\times 2},\ \forall i,j \in \mathcal{V},\\
\mathbb{E}[v_i^0 (v_j^0)^\top] &= \delta(i,j)R_i^0 \in \mathbb{R}^{2\times 2},\ \forall i,j \in \mathcal{V},\\
\mathbb{E}[v_i^0 v_j^\top] &= 0_{2\times 2},\ \forall i,j \in \mathcal{V}.\\
\mathbb{E}[v_i^0 w_j^\top] &= 0_{2\times 2},\ \forall i \in \mathcal{V},\ j \in \mathcal{V} \cup \{0\},\\
\mathbb{E}[v_i w_j^\top] &= 0_{2\times 2}, \forall i \in \mathcal{V},\ j \in \mathcal{V} \cup \{0\},
\end{aligned}$$

where $\delta(i,j) = 1$ if and only if $i = j$ for any i or j and $\delta(i,j) = 0$ otherwise.

Secondly, if the target is not directly sensed by a certain agent, say agent i, at time instant k, the measurement noise covariance is assumed as infinity, i.e., $(R_i^0[k])^{-1} = 0$. This is referred as a *blind agent* in [1].

Last, it is assumed that if agent i is in the communication range of agent j's at time instant k, agent i can receive information sent by agent j, hereafter denoted as $(j, i) \in \mathcal{E}[k]$, with $\mathcal{E}[k]$ being the set of edges that model the communication link at time k. The set of in-neighbors and out-neighbors of agent i at time k, denoted respectively as $N_{i,in}[k]$ and $N_{i,out}[k]$, are defined as

$$\begin{aligned}
N_{i,in}[k] &\triangleq \{j \in \mathcal{V} | (j, i) \in \mathcal{E}[k]\},\\
N_{i,out}[k] &\triangleq \{j \in \mathcal{V} | (i, j) \in \mathcal{E}[k]\}.
\end{aligned}$$

2 This could be a GPS location measurement, a localized position using prior map, *etc.*

It is defined that $J_i[k] \triangleq N_{i,in}[k] \cup \{i\}$.

2.3. Motivations

Although with numbers of research directions in the fields of either distributed estimation or flocking multi-agent systems being existed, the research on the combination of them is still limited. In the distributed estimation problems, the local agents are usually assumed to have perfect knowledge about its own state (either being static or not) and hence are able to directly measure the target's state in the same (usually global) frame. In articles related to the topic of flocking multi-agent systems, the exchanged state information between agents usually lack of uncertainty characterization or are even noise-free. Moreover, it is well known that redundant measurements in general give better estimation performance. Motived by these facts, in this paper, we aim at searching for the possibilities of combining the two topics. A 2D-activate sensing problem is studied for the proof of concept. A comprehensive fully-distributed algorithm is proposed to let the local agents estimate and follow the state of interest.

2.4. Objective

Let $\hat{X}_i[k|k-1]$, $\hat{X}_i[k|k]$, $\hat{X}_i^0[k|k-1]$ and $\hat{X}_i^0[k|k]$ be, respectively, the prior self-estimate, the posterior self-estimate, the prior estimate and the posterior estimate of target state by agent i at time instant k. The objective is formulated as follow.

Each agent estimates its own state, cooperates with its local neighbors and estimates the state of the target. By using the information obtained by agent itself, as well as that from the local neighbors, each agent determines its own control action to physically push itself toward the target while avoiding collide with both the target and any other agent. During the entire process, each agent is only allowed to exchange information with its local neighbors for once per sampling time interval. Any information that might subject to locally unknown changes during the process, such as the graph topology or the total number of agents, is not allowed to be used.

3. Proposed algorithm

In this section, a comprehensive recursive algorithm is proposed to achieve the aforementioned objective. For each recursion, the actions executed at each agent can be divided as 5 steps, as will be shown in detail below.

3.1. Self-estimation

As the first step, each agent uses its GPS-like measurement to estimate its own state. As (F, H) is an observable pair, a standard Kalman filter is used. Suppose that agent i has a prior self-estimate $\hat{X}_i[k|k-1]$, and updates its self-estimate and corresponding error covariance using $z_i[k]$:

$$\begin{aligned}
r_i[k] &= z_i[k] - H\hat{X}_i[k|k-1], \\
S_i[k] &= HP_i[k|k-1]H^\top + R_i[k], \\
K_i[k] &= P_i[k|k-1]H^\top S_i^{-1}[k], \\
\hat{X}_i[k|k] &= \hat{X}_i[k|k-1] + K_i[k]r_i[k], \\
P_i[k|k] &= (I_{4\times 4} - K_i[k]H)P_i[k|k-1],
\end{aligned}$$

where $P_i[k|k]$ is the posterior error covariance of the agent i's self-estimation.

3.2. Measurement regulation and packing

Note that $z_i^0[k]$ measures the relative position of the target of interest with respect to agent i. Therefore, in order for every agent to fuse the information received from its neighbors and estimate the target in the next step, each agent needs to regulate the measurement about the target into a standard form before transmitting it to its out-neighbors. That is, transform the information pair $(z_i^0[k], R_i^0[k])$ into another information pair, denoted as $(z_i'[k], R_i'[k])$, such that $z_i'[k]$ can be written as a linear model of the state of interest , *i.e.*,

$$z_i'[k] = HX_0[k] + v_i'[k].$$

Let $z_i'[k] \triangleq z_i[k] + H\hat{X}_i[k|k]$, $v_i'[k] \triangleq H\tilde{X}_i[k|k] + v_i^0[k]$, where $\tilde{X}_i[k|k] \triangleq \hat{X}_i[k|k] - X_i[k]$. It follows that,

$$\begin{aligned}
z_i'[k] &= z_i[k] + H\hat{X}_i[k|k] \\
&= HX_0[k] - HX_i[k] + v_i[k] + H\hat{X}_i[k|k] \\
&= HX_0[k] + H\tilde{X}_i[k|k] + v_i^0[k] \\
&= HX_0[k] + v_i'[k].
\end{aligned}$$

It follows that

$$R_i'[k] \triangleq \mathbb{E}[(v_i'[k])(v_i'[k])^\top] = HP_i[k|k]H^\top + R_i[k].$$

With the regulated local measurement, each agent computes the corresponding information matrix and information vector, denoted respectively as, $Y_i[k]$ and $y_i[k]$:

$$Y_i[k] \triangleq H^\top (R_i'[k])^{-1} H,\ y_i[k] \triangleq H^\top (R_i'[k])^{-1} z_i'[k].$$

Similarly, it computes

$$\Xi_i^0[k] \triangleq (P_i^0[k|k-1])^{-1},\ \xi_i^0[k] \triangleq \Xi_i^0[k] X_i^0[k|k-1],$$

where $\hat{X}_i^0[k|k-1]$ and $P_i^0[k|k-1]$ are, respectively, agent i's local prior estimate about the target state and corresponding approximated error covariance. Then each agent packs an information kit, denoted as $\Pi_i[k]$, and sends to every agent in its communication range. Specifically, $\Pi_i[k]$ contains the following information:

$$\Pi_i[k] \triangleq \{y_i[k], Y_i[k], \xi_i^0[k], \Xi_i^0[k], \hat{X}_i[k|k]\}.$$

With the information kit well packed, for each $i \in \mathcal{V}$, agent i sends $\Pi_i[k]$ to each agent $j \in N_{i,out}[k]$.

3.3. Estimation of the target state using the hybrid information fusion algorithm

To estimate the target with only local communication between neighboring agents, the unified framework, namely, the distributed hybrid information fusion (DHIF) algorithm proposed in [1], is adopted. This is due to its advantages of being fully distributed, requiring only one communication iteration per time instant, giving confident but consistent[3] local estimates, and having guaranteed convergence under very mild conditions.[4] The update step of the DHIF algorithm can be treated as 2 sub-steps.

The first sub-step is to fuse all local prior estimates obtained from the inclusive neighborhood using the Covariance Intersection (CI) algorithm (Julier 97'). The motivation for this is to guarantee the consistency of the fused information due to the lack of tracks about the correlations of each local prior estimate using the distributed scheme. This correlation is not negligible because each pair of local estimates will become highly correlated due to the same process noise of the target. Using agent i for example, it computes the following:

$$\check{\Xi}_i^0[k] = \sum_{j \in J_i[k]} \omega_{ij}^{(k)} \Xi_j^0[k],\ \check{\xi}_i^0[k] = \sum_{j \in J_i[k]} \omega_{ij}^{(k)} \xi_j^0[k].$$

Here $\omega_{ij}^{(k)} > 0$, $\sum_{j \in J_i[k]} \omega_{ij}^{(k)} = 1$ are the weights agent i assigned to the information obtained from agent j at time instant k. The Fast Covariance Intersection (FCI, Niehsen 02') method is used to efficiently determine the weights that increase the confidence of the CI approximation.

The second sub-step is to fuse this pair of information with the regulated local measurements from the inclusive neighborhood. As all these information sources are uncorrelated in errors, the optimal information fusion process is used to update the local posterior estimate about the target state.

$$P_i^0[k|k] = \left(\check{\Xi}_i^0[k] + \sum\nolimits_{j \in J_i[k]} Y_j[k]\right)^{-1}, \tag{2}$$

$$\hat{X}_i^0[k|k] = P_i^0[k|k] \left(\check{\xi}_i^0[k] + \sum\nolimits_{j \in J_i[k]} y_j[k]\right). \tag{3}$$

The fused posterior estimate and corresponding approximated error covariance will be used as the reference of the target state later on in designing the local control input.

The propagation steps of the local estimate of the target state follow the centralized Kalman filter. That is,

$$\hat{X}_i^0[k+1|k] = F\hat{X}_i^0[k|k],\ P_i^0[k+1|k] = FP_i^0[k|k]F^\top + Q_i[k].$$

The propagated local prior estimates will be packed along with the regulated measurements (in their information form) and sent to the out-neighbors in the next time instant.

3.4. Controller design using local information

Recall that in order to have more redundant measurements across the network, it is desired to keep the networked agents as a flock around the moving target. This requires all agents to have the velocities similar to that of the target's. Meanwhile, the distance between each pair of agents or the agent and the target should be maintained around a desired distance in the

3 An estimate is consistent if its approximated covariance is lower bounded (in the positive semi-definite sense) by the true one.

4 The network topology could be time-varying and never have a spanning tree at any single time instant. See [1] for more details.

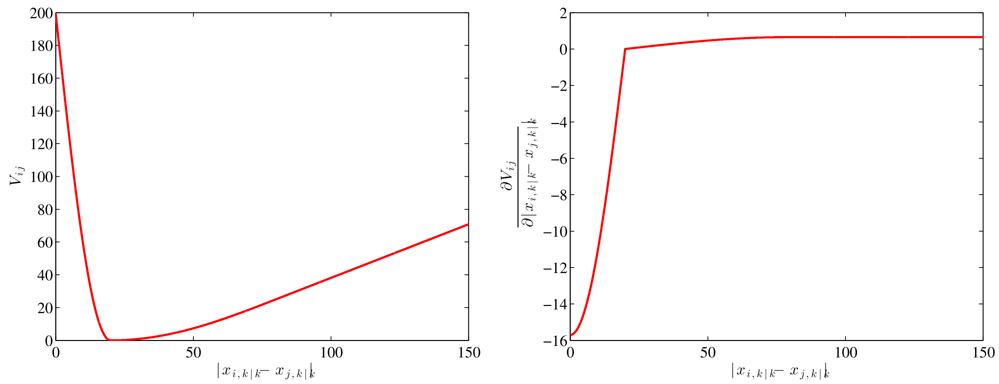

Fig. 2. $V(d)$ (left) and $\frac{\partial V}{\partial d}$ (right) with $d^* = 20$, $a = 10d^*$, $b = 25$ and $R = 3b$.

sense that: (i) more agents get close to the target so that more redundant collective measurements can be obtained; (ii) the collision should be avoided.

Let $\hat{p}_i[k|k]$, $\hat{v}_i[k|k]$, $\hat{p}_i^0[k|k]$ and $\hat{v}_i^0[k|k]$ be agent i's posterior estimate of, respectively, its own position, velocity, the target's position and velocity at time instant k. The following local control input is designed for each $i \in \mathcal{V}$.

$$\begin{aligned} U_i[k] &= c_3 \ln(\|U_i'\|_2 + 1)\mathrm{sgn}(U_i'), \\ U_i' &\triangleq U_{i1} + U_{i2}, \\ U_{i1} &= c_1 \partial V_{i0}(\hat{p}_i^0[k|k] - \hat{p}_i[k|k]) \\ &\quad + c_2(\hat{v}_i^0[k|k] - \hat{v}_i[k|k]), \\ U_{i2} &= c_1 \sum_{j\in J_i[k]} \partial V_{ij}(\hat{p}_j[k|k] - \hat{p}_i[k|k]) \\ &\quad + c_2 \sum_{j\in J_i[k]} (\hat{v}_j[k|k] - \hat{v}_i[k|k]), \end{aligned} \tag{4}$$

where $\partial V_{i0} \triangleq \frac{\partial V}{\partial d}\big|_{d=\hat{d}_{i0}}$ and $\partial V_{ij} \triangleq \frac{\partial V}{\partial d}\big|_{d=\hat{d}_{ij}}$ with

$$\hat{d}_{i0} \triangleq \|\hat{p}_i^0[k|k] - \hat{p}_i[k|k]\|_2, \ \hat{d}_{ij} \triangleq \|\hat{p}_i[k|k] - \hat{p}_j[k|k]\|_2. \tag{5}$$

The function $V(d)$ is referred as the potential function whose value depends on the distance between each pair of agents or an agent and the target. Let d^* be the desired distance. The potential function needs to have the following properties:

(i) $\frac{\partial V}{\partial d}\big|_{d=d^*} = 0$ so that $V(d^*)$ has the minimum value;
(ii) $V(d)$ increases to large values as $d \to 0$;
(iii) $V(d)$ increases as $d \to \infty$ if $d > d^*$.

The potential function $V(d)$ selected for the simulation later on and its corresponding $\frac{\partial V}{\partial d}$ has the form of

$$V(d) = \begin{cases} a(1 - \sin \dfrac{\pi d}{2d^*}), & d \le d^*; \\ b(1 - \cos \dfrac{\pi(d-d^*)}{2(R-d^*)}), & d^* < d \le R; \\ \dfrac{b\pi(d-R)}{2(R-d^*)}, & R < d, \end{cases} \tag{6}$$

where $a, b, R \in \mathbb{R}^+$. In Fig. 2, the $V(d)$ as defined in (6) and the corresponding $\frac{\partial V}{\partial d}$ are plotted with $d^* = 20$, $a = 10d^*$, $b = 25$ and $R = 3b$. The intuition of the controller designed in (4) is explained here. When $\hat{d}_{i0} < 20$, the fact that $\partial V_{i0} < 0$ will push agent i itself away from the target to avoid collision; on the other hand, when $\hat{d}_{i0} > d^*$, the fact that $\partial V_{i0} > 0$ will push agent i toward the target. Moreover, using the difference in velocity, agent i pushes its own velocity toward its neighbors'. Similarly, U_{2i} in Eq. (4) pushes agent i toward or away from its neighbors based on the estimated distances in between. By applying the control input in Eq. (4) with appropriately designed control gains (c_1, c_2 and c_3), each agent cooperates with its neighbors and eventually the group of networked agents will form a flock and move along with the target.

Remark 1. The logarithm operator in Eq. (4) is to scale the control input into a feasible range for actuation. This is because the value of the potential function could become infeasibly large if the estimated distance is too far. Note that this would not be an issue for U_{i2} as agents will not be neighbors if they are physically far away from each other. However, in can be seen from U_{i1} in Eq. (4) that the target is virtually an in-neighbor of each agent, which may bring the issue. By using the logarithm operator multiplied with sign function, the relativity of the position differences between each pair of agents are "mapped" to a feasible scale.

3.5. Actuation & self-estimate propagation

With the local control input determined in the previous step and sent to the actuator, each agent evolves using Eq. (1). Moreover, using the same equation, agent i also propagates the estimate of its own state for the self-estimation step in the next time instant. That is,

$$\hat{X}_i[k+1|k] = F\hat{X}_i[k|k] + BU_i[k].$$

Before wrapping up this section, the authors would like to clarified that, although the proposed algorithm does not require every agent to be able to directly sense the target at the initial time instant, the target should be in the sensing range of at least a subset (with at least one agent) of agents. As for the communication topology, every agent in the network should have a direct path from at least one agent belonging to this subset. Note that these conditions are very mild in the sense that they do not even require a directed spanning tree in the network. This is inherited from the very mild conditions of the original DHIF algorithm explicitly formulated in [1].

4. Recovery of the Centralized Kalman filter using the DHIF Algorithm

The motivation for the strategy of active sensing using flocking mobile agents, as discussed in the previous section is that, with more redundant measurements of the target of interest, a better estimate could be obtained intuitively. It is worth mentioning that in [1], it is explicitly shown by mathematical derivations that more redundant measurements would lead to more confident (and hence more accurate given consistencies) estimates. In this section, we show that if the communication is complete, the DHIF algorithm proposed in [1] actually recovers the solution obtained by the centralized Kalman filter, and hence is the optimal (in the MSE sense) estimate in the scenario of linear models and Gaussian white noises.

Consider a hypothetic centralized Kalman filter (CKF), which collects all the measurements obtained by all agents across the network at every time instant. Denoted $z'_c[k] \triangleq \mathrm{col}\{z'_i[k]\}_{i\in\mathcal{V}}$ the collective measurement used by the CKF. Let $v'_c[k] \triangleq \mathrm{col}\{v'_i[k]\}_{i\in\mathcal{V}}$. Let H_c be the matrix obtained by stacking N of H matrices. It follows that

$$z'_c[k] = H_c X_0[k] + v'_c[k]. \tag{7}$$

One can easily find that $v'_c[k] \sim \mathcal{N}(0_{2N\times 1}, R'_c[k])$, where $R'_c[k] \triangleq \mathrm{blkdiag}\{R'_i[k]\}_{i\in\mathcal{V}}$.

Theorem 1. *Let $\hat{X}_c[k|k-1]$ and $\hat{X}_c[k|k]$ be, respectively, the prior and posterior estimate obtained by the CKF using $z'_c[k]$ defined in (7). Let $P_c[k|k-1]$, $P_c[k|k]$ be the corresponding estimate error covariance matrices. If the graph is complete, each local agent i, $i \in \mathcal{V}$ is able to use the DHIF algorithm to recover the solution obtained by the CKF.*

Proof. Note that the information filter is mathematically equivalent to the CKF. Therefore, one can use the information filter to find the same solution as that of the CKF. Hence the update step of the CKF can be written as

$$\begin{aligned}(P_c[k|k])^{-1} &= (P_c[k|k-1])^{-1} + H_c^\top (R'_c[k])^{-1} H_c,\\ (P_c[k|k])^{-1} X_c[k|k] &= (P_c[k|k-1])^{-1} X_c[k|k-1] + H_c^\top (R'_c[k])^{-1} z'_c[k].\end{aligned} \tag{8}$$

Note that due to the structure of H_c and $R'_c[k]$,

$$\begin{aligned}H_c^\top (R'_c[k])^{-1} H_c &= \sum\nolimits_{i\in\mathcal{V}} H^\top (R'_i[k])^{-1} H,\\ H_c^\top (R'_c[k])^{-1} z'_c[k] &= \sum\nolimits_{i\in\mathcal{V}} H^\top (R'_i[k])^{-1} z'_i[k].\end{aligned}$$

Other the other hand, if the graph is complete at time k, one has $J_i[k] = \mathcal{V}$. It follows from (2) and (3) that the update steps of the DHIF algorithm has the form of

$$\begin{aligned}(P_i^0[k|k])^{-1} &= \check{\Xi}_i^0[k] + \sum\nolimits_{j\in\mathcal{V}} Y_j[k] = \check{\Xi}_i^0[k] + H_c^\top (R'_c[k])^{-1} H_c,\\ (P_i^0[k|k])^{-1}\hat{X}_i^0[k|k] &= \check{\xi}_i^0[k] + \sum\nolimits_{j\in\mathcal{V}} y_j[k] = \check{\xi}_i^0[k] + H_c^\top (R'_c[k])^{-1} z'_i[k].\end{aligned}$$

From the above equations, one can observe that, with a complete graph, the local estimates obtained by the DHIF algorithm uses the same measurement as that of the CKF. Therefore, each local estimate at agents i can be regarded as a CKF with initialized estimate pair $(\hat{X}_i^0[k+1|k], P_i^0[k+1|k])$ at $k = k'$, where k' is the time instant such that the graph is complete for all $k \geq k'$. Hence the time invariance property [47] of the CKF implies that at $k \gg k'$, one has $P_i^0[k+1|k] \to P_c[k+1|k]$ and $P_i^0[k+1|k] \to P_c[k|k]$. This concludes the proof.

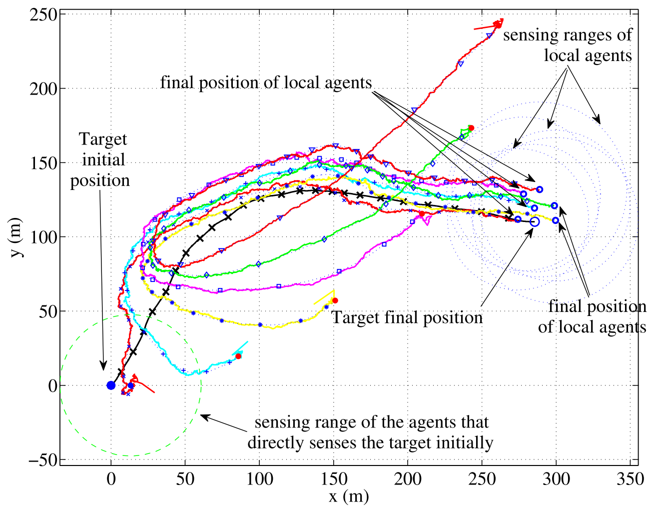

Fig. 3. Trajectories of the target (black slide line) and all local agents (blue dotted lines) in 25 s. The solid lines in different colors stands for the self-estimated trajectories of each agent. For each agent's estimated trajectory, a marker in different type is also plotted every second (*i.e.*, $100\Delta T$). The initial positions are in hollow circles, where the bigger one stands for the target's and the smaller ones stand for the agents'. The small hollow circle in blue implies that the target is in its sensing range (denoted in green dashed circle with radius being $\mathcal{R}_i^s$) initially. The small hollow circles in red means that it cannot observe the target initially. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Remark 2. Note that the proof in Theorem 1 also applies in the case if only a subgraph rather than the whole graph is complete. In such case, the CKF also has less number of redundant measurements.

5. Simulation

This section shows the simulation result using the algorithm proposed in the previous section. A group of $N = 6$ agents are selected to track the target. The sensing range and communication range of each agent, respectively denoted as, $\mathcal{R}_i^s$ and $\mathcal{R}_i^c$, with $i \in \{1, \dots, N\}$, are randomly selected such that, $\mathcal{R}_i^s \sim \mathcal{U}(40, 60)$ and $\mathcal{R}_i^c \sim \mathcal{U}(70, 100)$. The desired distance between each pair of agents or an agent and the target d^* is selected as 10 m. The potential function is selected as the one shown in Fig. 2. The parameters are selected as follow. The sampling time interval $\Delta T = 0.01$ s. The covariances of the process noise of the target or each agent, the noise of the GPS-like signal and the noise of the measured relative positions are, respectively,

$$Q_0 = Q_i = \begin{bmatrix} \frac{5\Delta T^3}{3} I_{2\times 2} & \frac{5\Delta T^2}{2} I_{2\times 2} \\ \frac{5\Delta T^2}{2} I_{2\times 2} & \frac{5\Delta T^3}{3} I_{2\times 2} \end{bmatrix}, \ R_i = 100 I_{2\times 2},$$

and $R_i^0 = 225 I_{2\times 2}$, $\forall i \in \{1, \dots, N\}$. The initial position of the target and agents are selected as $p_0[0] = [0, 0]^\top$, $p_1[0] = [r_1, 0]^\top$ and

$$p_i[0] = \begin{bmatrix} r_i \cos(\frac{\pi(i-1)}{2N}) \\ r_i \sin(\frac{\pi(i-1)}{2N}) \end{bmatrix}, \forall i \in \{2, \dots, N\},$$

where $r_1 \sim R_1^s \times \mathcal{U}(0.8, 1)$ and $r_i \sim \min(R_{i-1}^s, R_i^s) \times \mathcal{U}(0.8, 1)$, $\forall i \in \{2, \dots, N\}$.

The initial velocity of the target along each direction is selected from 0 to 10 based on the uniform distribution. The initial velocities of all agents are selected as zeros.

Remark 3. By selecting the initial position in the described manner, it is guaranteed that agent 1 will be able to directly see the target at the beginning while the other agents cannot see the target. Moreover, the topology is very sparse in the sense that it is a 'chain' topology. Note that this initialization results in very week conditions in terms of both observability and communication. This is to show the strength of the proposed algorithm as it will be shown that every agent will move toward the target and flock with it, which in turns dramatically increase the estimation quality.

The trajectories of the target and all agents for 25 s are plotted in Fig. 3. As observed in Fig. 3, as time goes on, each agent moves toward the target and follows the target along with the other agents as a flock. The final positions are plotted in circles

Table 1
Final distances between agents ($i \in \mathcal{V}$) and the target.

Agent i	1	2	3	4	5	6
2	13.4	–	–	–	–	–
3	25.2	16.3	–	–	–	–
4	17.5	12.1	27.9	–	–	–
5	26.3	13.6	9.9	22.4	–	–
6	24.8	13.1	23.4	11.2	14.9	–
Target	11.4	9.2	14.0	20.2	17.2	22.0

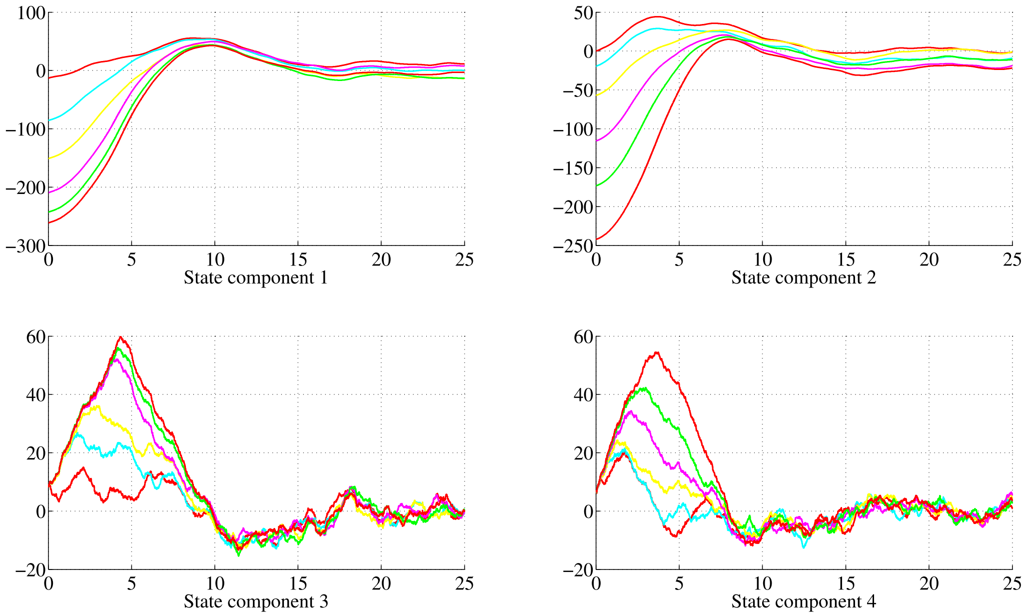

Fig. 4. Differences of all 4 state components of $(X_i[k] - X_0[k])$, for each $i \in \mathcal{V}$. The color for each agent is consistent with those in Fig. 3.

with the bigger one being the target's and the smaller ones being the agents'. If the target is in an agent's sensing range, a big blue dotted circle with the radius being $\mathcal{R}_i^s$ is plotted. It can be observed that all agents can observe the target directly in the end.

The final distances between agents and the target are shown Table 1, where one can observe that the collision is avoided.

The differences of all 4 states between each agent and the target are plotted in Fig. 4, where it is seen that the relative position of each agent with respect to the target converges to a value around a constant. The relative velocities of all agents converge to values around zero with fluctuations in the steady state. These are due to the process noise that affects both the target and the agents. The control inputs along each direction applied by all 6 agents are plotted in Fig. 5. It can be seen that the input values are in a reasonable range all the time. This is due to the use of the logarithm function in the control input.

Fig. 6 shows the tracking errors (solid line in different colors that are consistent with previous figures) of all 4 states in estimating the target state by each agent, as well as the corresponding 3σ bound of the approximated error covariances (dashed lines in the same color with respect to each agent). Fig. 7 shows the zoomed-in version of the top-left subfigure of Fig. 6. It can be observed from Fig. 7 that the error covariances of all agents dramatically drop at some time instants, and eventually overlap with that of the centralized Kalman filter's. This is because as more agents getting closer to each other, every agent is in the communication range of every other agent's. As a result, the communication topology becomes a complete (all-to-all) graph. This matches the proof in the previous section.

6. Conclusion

In this paper, a distributed active target tracking task using networked moving agents is studied. Using the DHIF algorithm proposed in [1] with certain modifications, each agent only uses local information, estimates not only its own state, but also the target state. On the top of those, each agent designs its own control algorithm, and eventually physically track the target in the sense that: (i) it keeps the target in its sensing range while avoid collision with the target or any other agent; (ii) it moves along with the target to maintain a better collective measurement redundancy. The whole group of agents eventually

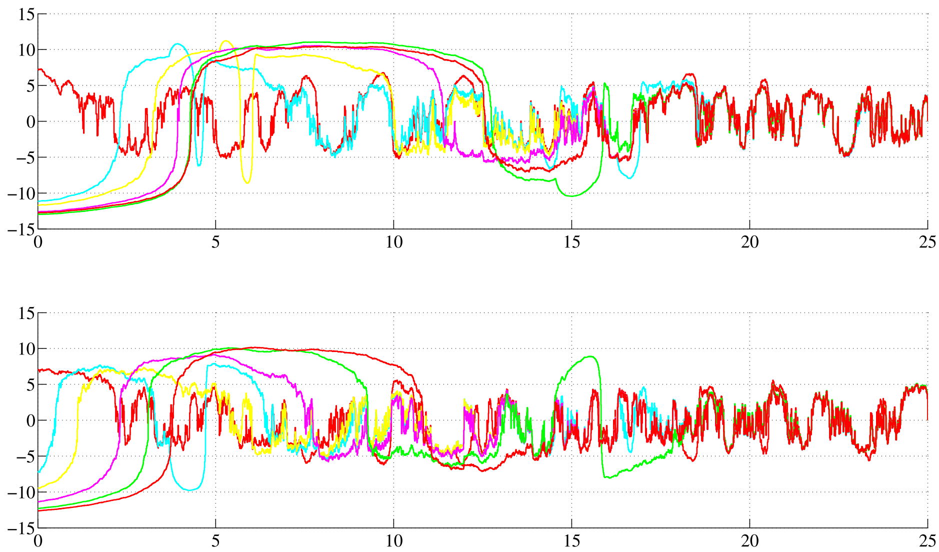

Fig. 5. Control input applied by all 6 agents (top: x-direction; bottom: y-direction). The color for each agent is consistent with those in Fig. 3.

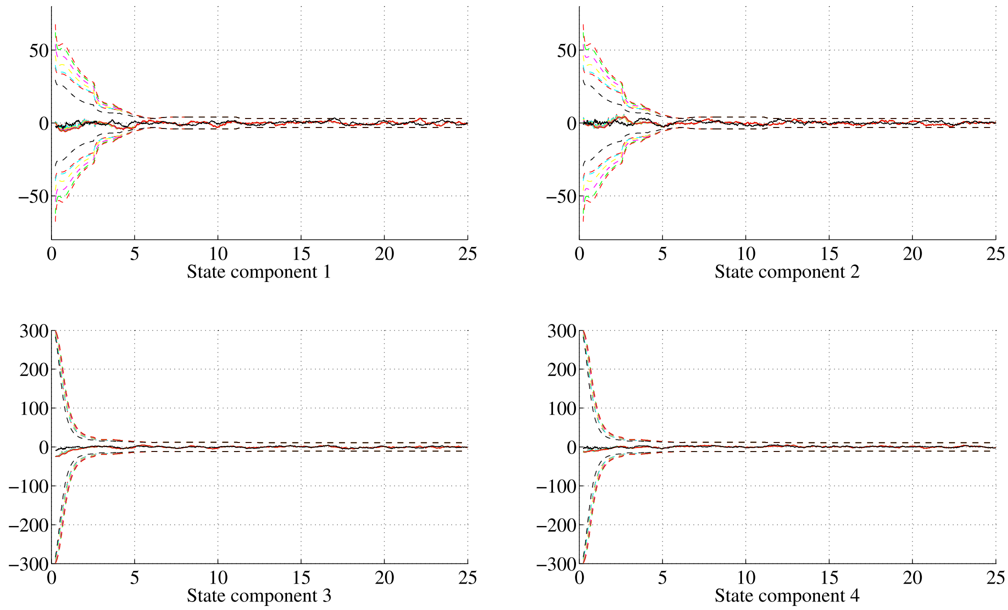

Fig. 6. Tracking errors (solid lines) in all 4 state components in estimating the target state by each agent with $\pm3\sigma$ bounds. The color for each agent is consistent with those in Fig. 3. A virtual centralized Kalman filter, which uses the collective measurements from all agents at all time instants, is used as the benchmark for the purpose of comparison. The estimation error and corresponding $\pm3\sigma$ bounds obtained by the centralized Kalman filter are plotted in black. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

flocks with the target. With the communication topology becomes complete and each agent seeing the target, the centralized solution is recovered.

Acknowledgments

This work is supported by Hunan Provincial Science and Technology Program under Grant 2015GK3022, Central South University Post-Doctor Scientific Research Foundation under Grant 185643, The Fundamental Research Funds for the Central Universities.

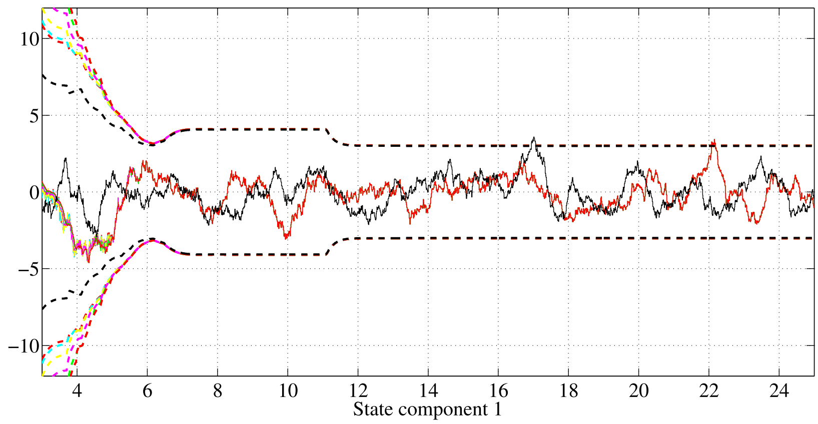

Fig. 7. Zoomed-in of the top-left subfigure of Fig. 6.

References

[1] S. Wang, W. Ren, On the convergence conditions of distributed dynamic state estimation using sensor networks: a unified framework, IEEE Trans. Control Syst. Technol., 2017 http://dx.doi.org/10.1109/TCST.2017.2715849.

[2] R. Olfati-Saber, Kalman-consensus filter: Optimality, stability, and performance, in: Proceedings of the IEEE Conference on Decision and Control, Shanghai, China, 2009, pp. 7036–7042.

[3] A.T. Kamal, J.A. Farrell, A.K.R. Chowdhury, Information weighted consensus filters and their application in distributed camera networks, IEEE Trans. Automat. Control 58 (12) (2013) 3112–3125.

[4] F. Cattivelli, A. Sayed, Diffusion strategies for distributed Kalman filtering and smoothing, IEEE Trans. Automat. Control 55 (9) (2010) 2069–2084.

[5] J. Hu, L. Xie, C. Zhang, Diffusion Kalman filtering based on covariance intersection, IEEE Trans. Signal Process. 60 (2) (2012) 891–902.

[6] G. Battistelli, L. Chisci, Kullback–Leibler average, consensus on probability densities, and distributed state estimation with guaranteed stability, Automatica 50 (3) (2014) 707–718.

[7] G. Battistelli, L. Chisci, G. Mugnai, A. Farina, A. Graziano, Consensus-based linear and nonlinear filtering, IEEE Trans. Automat. Control 60 (5) (2015) 1410–1415.

[8] S. Wang, W. Ren, On the consistency and confidence of distributed dynamic state estimation in wireless sensor networks, in: Proceedings of the IEEE Conference on Decision and Control, Osaka, Japan, 2015, pp. 3069–3074.

[9] S. Wang, W. Ren, On the convergence of distributed estimation of LTV dynamic system with switching directed topologies and time-varying sensing models, in: Proceedings of the American Control Conference, Boston, MA, 2016, pp. 5437–5442.

[10] S. Wang, W. Ren, J. Chen, Fully distributed state estimation with multiple model approach, in: Proceedings of the IEEE Conference on Decision and Control, Las Vegas, NV, 2016, pp. 2920–2925.

[11] W. Li, Y. Jia, Consensus-based distributed multiple model UKF for jump Markov nonlinear systems, IEEE Trans. Automat. Control 57 (1) (2012) 227–233.

[12] C. Fantacci, G. Battistelli, L. Chisci, A. Farina, A. Graziano, Multiple-model algorithms for distributed tracking of a maneuvering target, Information Fusion (FUSION), 2012 15th International Conference on, 2012, pp. 1028–1035.

[13] A. Mohammadi, K. Plataniotis, Distributed widely linear multiple-model adaptive estimation, IEEE Trans. Signal Inf. Process. Netw. 1 (3) (2015) 164–179.

[14] S. Wang, W. Ren, J. Chen, Fully distributed dynamic state estimation with uncertain process models, IEEE Trans. Control Netw. Syst. PP (99) (2017) 1.

[15] V. Ugrinovskii, Distributed robust estimation over randomly switching networks using h_∞ consensus, Automatica 49 (1) (2013) 160–168.

[16] V. Ugrinovskii, Detectability of distributed consensus-based observer networks: an elementary analysis and extensions, in: Control Conference (AUCC), 2014 4th Australian, IEEE, 2014, pp. 188–192.

[17] Y. Li, R.G. Sanfelice, Interconnected observers for robust decentralized estimation with performance guarantees and optimized connectivity graph, IEEE Trans. Control Netw. Syst. 3 (1) (2016) 1–11.

[18] Y. Li, S. Phillips, R.G. Sanfelice, Robust distributed estimation for linear systems under intermittent information, IEEE Trans. Automat. Control PP (99) (2017) 1.

[19] S. Phillips, Y. Li, R.G. Sanfelice, A hybrid consensus protocol for pointwise exponential stability with intermittent information, IFAC-PapersOnLine 49 (18) (2016) 146–151, 10th IFAC Symposium on Nonlinear Control Systems NOLCOS 2016.

[20] Z.-K. Zhang, C. Liu, X.-X. Zhan, X. Lu, C.-X. Zhang, Y.-C. Zhang, Dynamics of information diffusion and its applications on complex networks, Phys. Rep. 651 (2016) 1–34, Dynamics of information diffusion and its applications on complex networks, http://www.sciencedirect.com/science/article/pii/S0370157316301600.

[21] C. Liu, Z.-K. Zhang, Information spreading on dynamic social networks, Commun. Nonlinear Sci. Numer. Simul. 19 (4) (2014) 896–904, http://www.sciencedirect.com/science/article/pii/S100757041300378X.

[22] L. Yu, J. Huang, G. Zhou, C. Liu, Z.-K. Zhang, Tiirec: a tensor approach for tag-driven item recommendation with sparse user generated content, Inform. Sci. 411 (2017) 122–135, http://www.sciencedirect.com/science/article/pii/S002002551730734X.

[23] C.W. Reynolds, Flocks, herds and schools: a distributed behavioral model, in: Conference on Computer Graphics and Interactive Techniques, 1987, pp. 25–34.

[24] G.-Q. Sun, S.-L. Wang, Q. Ren, Z. Jin, Y.-P. Wu, Effects of time delay and space on herbivore dynamics: linking inducible defenses of plants to herbivore outbreak, Sci. Rep. 5 (2015) 11246.

[25] G.-Q. Sun, M. Jusup, Z. Jin, Y. Wang, Z. Wang, Pattern transitions in spatial epidemics: mechanisms and emergent properties, Phys. Life Rev. 19 (2016) 43–73.

[26] G.-Q. Sun, J.-H. Xie, S.-H. Huang, Z. Jin, M.-T. Li, L. Liu, Transmission dynamics of cholera: mathematical modeling and control strategies, Commun. Nonlinear Sci. Numer. Simul. 45 (2017) 235–244.

[27] G.-Q. Sun, C.-H. Wang, Z.-Y. Wu, Pattern dynamics of a gierer–meinhardt model with spatial effects, Nonlinear Dynam. 88 (2) (2017) 1385–1396.

[28] L. Li, Bifurcation and chaos in a discrete physiological control system, Appl. Math. Comput. 252 (2015) 397–404.
[29] R. Olfati-Saber, Flocking for multi-agent dynamic systems: algorithms and theory, IEEE Trans. Automat. Control 51 (3) (2006) 401–420.
[30] D. Canepa, M.G. Potopbutucaru, Stabilizing flocking via leader election in robot networks, in: 9th International Conference on Stabilization, Safety, and Security of Distributed Systems, 2007, pp. 52–66.
[31] H. Shi, L. Wang, T. Chu, F. Fu, Flocking of multi-agent systems with a virtual leader, Internat. J. Control 82 (1) (2009) 43–58.
[32] H. Su, X. Wang, Z. Lin, Flocking of multi-agents with a virtual leader, IEEE Trans. Automat. Control 54 (2) (2009) 293–307.
[33] Y. Dong, J. Huang, Flocking with connectivity preservation of multiple double integrator systems subject to external disturbances by a distributed control law, Pergamon Press, Inc., 2015, pp. 197–203.
[34] B.J. Wellman, J.B. Hoagg, A flocking algorithm with individual agent destinations and without a centralized leader, Systems Control Lett. 102 (2017) 57–67.
[35] X. Lu, F. Austin, S. Chen, Flocking in multi-agent systems with active virtual leader and time-varying delays coupling, Commun. Nonlinear Sci. Numer. Simul. 16 (2) (2011) 1014–1026.
[36] S. Yazdani, M. Haeri, Flocking of multi-agent systems with multiple second-order uncoupled linear dynamics and virtual leader, IET Control Theory Appl. 10 (8) (2016) 853–860.
[37] D. Gu, Z. Wang, Leader? Follower flocking: algorithms and experiments, IEEE Trans. Control Syst. Technol. 17 (5) (2009) 1211–1219.
[38] N. Ganganath, C.T. Cheng, K.T. Chi, X. Wang, Cluster-based informed agents selection for flocking with a virtual leader, in:IEEE International Symposium on Circuits and Systems, 2015, pp. 2692–2695.
[39] M.M. Zavlanos, H.G. Tanner, A. Jadbabaie, G.J. Pappas, Hybrid control for connectivity preserving flocking, IEEE Trans. Automat. Control 54 (12) (2009) 2869–2875.
[40] J. Zhou, H. Qian, X. Lu, Multi-agent flocking via generalized control algorithms: existence and properties, in: International Conference on Control, Automation, Robotics and Vision, 2017, pp. 1–6.
[41] J. Gao, X. Xu, N. Ding, E. Li, Flocking motion of multi-agent system by dynamic pinning control, IET Control Theory Appl. 11 (5) (2017) 714–722.
[42] S. Li, X. Liu, W. Tang, J. Zhang, Flocking of multi? agents following a leader with adaptive protocol in a noisy environment, Asian J. Control 16 (6) (2015) 1771–1778.
[43] F. Sun, R. Wang, Y. Li, F. Liu, Flocking for leader-following multi-agent systems with time-varying delay, Intell. Control Autom. 07 (1) (2016) 9–15.
[44] S. Ghapani, J. Mei, W. Ren, Y. Song, Fully distributed flocking with a moving leader for Lagrange networks with parametric uncertainties, Automatica 67 (2016) 67–76.
[45] J. Luo, C. Cao, Flocking for multi-agent systems with unknown nonlinear time-varying uncertainties under a fixed undirected graph, Internat. J. Control 88 (5) (2015) 1051–1062.
[46] X. Luo, X. Li, S. Li, Z. Jiang, X. Guan, Flocking for multi-agent systems with optimally rigid topology based on information weighted kalman consensus filter, Int. J. Control Autom. Syst. 15 (1) (2016) 1–11.
[47] Y. Bar-Shalom, X.-R. Li, T. Kirubarajan, Estimation with Applications to Tracking and Navigation, John Wiley & Sons, Inc. cop., New York (N.Y.), Chichester, Weinheim, 2001.