Distributed Networked Control System with Lossy Links: State Estimation and Stabilizing Communication Control

Songhwai Oh and Shankar Sastry

***Abstract*— This paper introduces a distributed networked control system (DNCS) consisting of multiple agents communicating over a lossy communication channel, *e.g.*, wireless channel. Two aspects of DNCSs are studied in this paper – state estimation and stabilizing communication control. Based on the Kalman filter, optimal linear filtering algorithms are derived for the discrete-time linear dynamic models of the DNCS with lossy links. Then, the problem of finding a communication control which stabilizes a DNCS is considered. The stabilizing communication control problem seeks the acceptable ranges of packet loss rates at which the overall system is stable. Efficient algorithms based on convex optimization are developed for solving the stabilizing communication control problem.**

I. Introduction

With the recent developments in communication, computing, and control systems, a networked control system (NCS) has received a fair amount of attention recently. In a general sense, an NCS consists of spatially distributed multiple systems or agents equipped with sensors, actuators, and computing and communication devices. The operation of each agent is coordinated over a communication network. The examples of an NCS include sensor networks [1], networked autonomous mobile agents [2], *e.g.*, a team of UAVs, and arrays of micro or micro-electromechanical sensors (MEMS) devices.

Recently, different aspects of NCSs have been studied extensively. Sinopoli *et al.* [3] showed the phase transition behavior of the Kalman filter when the measurement packet loss is modeled by a Bernoulli random process and established the relationship between the speed of dynamics and the packet loss rate for the stable estimation of the system. Similar estimation problems are discussed in [4], [5]. The control problems over an unreliable communication channel have been studied by many authors, including [6], [7], [8]. The stability of NCSs has been also studied in [9], [10].

There is a growing interest in consensus and coordination of networked systems inspired by the model by Vicsek *et al.* [11], in which a large number of particles (or autonomous agents) move at a constant speed but with different headings. At each discrete time, each particle updates its heading based on the average heading of its neighboring particles. The analysis of the Vicsek model in different forms are reported in [12], [13], [14]. In this paper, we extend NCSs to model a distributed multi-agent system such as the Vicsek model.

In general, a single plant is assumed in an NCS and the links between the plant and the estimator or controller is closed by a common (unreliable) communication channel. This paper extends this notion of NCSs by introducing a distributed networked control system (DNCS) consisting of multiple agents communicating over a lossy communication channel. The best examples of such system include ad-hoc wireless sensor networks and a network of mobile agents. We first consider the estimation problem appears in DNCSs and develop optimal linear filtering algorithms based on the Kalman filter.

Then, we consider the problem of finding a communication control which stabilizes a DNCS. This problem is called the *stabilizing communication control* problem and its goal is to find the acceptable ranges of packet loss rates at which the overall system is stable. We use the stability results for jump linear systems by Costa and Fragoso [15] to derive a sufficient condition for the stability of a DNCS. We then develop an efficient algorithm for checking the existence of stabilizing communication control using linear programming and discuss a method for solving the stabilizing communication control problem using geometric programming, a convex optimization method [16], [17].

The remainder of this paper is structured as follows. The dynamic models of DNCSs are described in Section II. The filtering algorithms for DNCSs are described in Section III. The stabilizing communication control problem is described in Section IV.

II. Distributed Networked Control Systems with Lossy Links

Let us first consider a distributed control system consisting of N agents, in which there is no communication loss. The discrete-time linear dynamic model of the agent j can be described as following:

$$x_j(k+1) = \sum_{i=1}^{N} A_{ij} x_i(k) + G_j w_j(k) \tag{1}$$

where $k \in \mathbb{Z}^+$, $x_j(k) \in \mathbb{R}^{n_x}$ is the state of the agent j at time k, $w_j(k) \in \mathbb{R}^{n_w}$ is a white noise process, $A_{ij} \in \mathbb{R}^{n_x \times n_x}$, and $G_j \in \mathbb{R}^{n_x \times n_w}$. Hence, the state of the agent j is governed by the previous states of all N agents. We can also consider $A_{ij} x_i(k)$ as a control input from the agent i to the agent j for $i \neq j$.

The authors are with the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA 94720, {sho,sastry}@eecs.berkeley.edu.

This material is based upon work supported by the National Science Foundation under Grant No. EIA-0122599 and the Team for Research in Ubiquitous Secure Technology (TRUST), which receives support from the National Science Foundation (NSF award number CCF-0424422) and the following organizations: Cisco, ESCHER, HP, IBM, Intel, Microsoft, ORNL, Pirelli, Qualcomm, Sun, Symantec, Telecom Italia and United Technologies.

Now consider a distributed networked control system (DNCS), in which agents communicate with each other over a lossy communication channel, *e.g.* wireless channel. We assume an erasure channel between a pair of agents. At each time k, a packet sent by the agent i is correctly received by the agent j with probability p_{ij}. We form a communication matrix $P_{\text{com}} = [p_{ij}]$. Let $Z_{ij}(k) \in \{0,1\}$ be a Bernoulli random variable, such that $Z_{ij}(k) = 1$ if a packet sent by the agent i is correctly received by the agent j at time k, otherwise, $Z_{ij}(k) = 0$. Since there is no communication loss within an agent, $p_{ii} = 1$ and $Z_{ii}(k) = 1$ for all i and k. For each (i,j) pair, $\{Z_{ij}(k)\}$ are i.i.d. (independent identically distributed) random variables such that $P(Z_{ij}(k) = 1) = p_{ij}$ for all k; and $Z_{ij}(k)$ are independent from $Z_{lm}(k)$ for $l \neq i$ or $m \neq j$. Then we can write the dynamic model of the agent j under lossy links as following:

$$x_j(k+1) = \sum_{i=1}^{N} Z_{ij}(k) A_{ij} x_i(k) + G_j w_j(k). \tag{2}$$

Let $x(k) = [x_1(k)^T, \ldots, x_N(k)^T]^T$ and $w(k) = [w_1(k)^T, \ldots, w_N(k)^T]^T$, where y^T is a transpose of y. Let $\bar{A}_{ij}$ be a $Nn_x \times Nn_x$ block matrix. The entries of $\bar{A}_{ij}$ are all zeroes except the (j,i)-th block is A_{ij}. For example, when $N = 2$

$$\bar{A}_{12} = \begin{bmatrix} \mathbf{0}_{n_x} & \mathbf{0}_{n_x} \\ A_{12} & \mathbf{0}_{n_x} \end{bmatrix},$$

where $\mathbf{0}_{n_x}$ is a $n_x \times n_x$ zero matrix. Then the discrete-time linear dynamic model of the DNCS with lossy links can be represented as following:

$$x(k+1) = \left(\sum_{i=1}^{N} \sum_{j=1}^{N} Z_{ij}(k) \bar{A}_{ij} \right) x(k) + G w(k), \tag{3}$$

where G is a block diagonal matrix of $G_1, \ldots, G_N$.

For notational convenience, we introduce a new index $n \in \{1, \ldots, N^2\}$ such that ij is indexed by $n = N(i-1) + j$. With this new index n, the dynamic model (3) can be rewritten as

$$x(k+1) = \left(\sum_{n=1}^{N^2} Z_n(k) \bar{A}_n \right) x(k) + G w(k). \tag{4}$$

By letting $A(k) = \left(\sum_{n=1}^{N^2} Z_n(k) \bar{A}_n \right)$, we see that (4) is a time-varying linear dynamic model:

$$x(k+1) = A(k) x(k) + G w(k). \tag{5}$$

Until now we have assumed that $\bar{A}_n$ is fixed for each n. We relax this assumption by letting $A(k) = A(Z(k))$, where $Z(k) = [Z_1(k), \ldots, Z_{N^2}(k)]^T$. This relaxed dynamical system is

$$x(k+1) = A(Z(k)) x(k) + G w(k). \tag{6}$$

The dynamic model (6) or (4) is a special case of the linear hybrid model or a jump linear system [15] since $A(k)$ takes an element from a set of a finite number of matrices. We will call the dynamic model (4) as the "simple" DNCS dynamic model and (6) as the "general" DNCS dynamic model.

III. State Estimation

In this section, we describe recursive filtering algorithms for the dynamic models (4) and (6) using the Kalman filter (KF). Since $Z(k)$ is independent from $Z(t)$ for $t \neq k$, we derive optimal linear filters for both cases. Notice that we denote $Z(k)$ by Z when there is no confusion.

A. KF for Simple DNCS

Consider the simple DNCS dynamic model (4), where $w(k)$ is a Gaussian noise with zero mean and covariance Q, and the following measurement model:

$$y(k) = C x(k) + v(k), \tag{7}$$

where $y(k) \in \mathbb{R}^{n_y}$ is a measurement at time k, $C \in \mathbb{R}^{n_y \times N n_x}$, and $v(k)$ is a Gaussian noise with zero mean and covariance R. Hence, we are assuming that the measurements are collected by a remote sensor or by a sensor in one of the agents.

The following terms are defined to describe the modified Kalman filter.

$$\begin{aligned} \hat{x}(k|k) &:= \mathbb{E}\left[x(k) | \mathbf{y_k} \right] \\ P(k|k) &:= \mathbb{E}\left[e(k) e(k)^T | \mathbf{y_k} \right] \\ \hat{x}(k+1|k) &:= \mathbb{E}\left[x(k+1) | \mathbf{y_k} \right] \\ P(k+1|k) &:= \mathbb{E}\left[e(k+1|k) e(k+1|k)^T | \mathbf{y_k} \right], \end{aligned}$$

where $\mathbf{y_k} = \{y(t) : 0 \leq t \leq k\}$, $e(k|k) = x(k) - \hat{x}(k|k)$, and $e(k+1|k) = x(k+1) - \hat{x}(k+1|k)$.

Suppose that we have estimates $\hat{x}(k|k)$ and $P(k|k)$ from time k. At time $k+1$, a new measurement $y(k+1)$ is received and our goal is to estimate $\hat{x}(k+1|k+1)$ and $P(k+1|k+1)$ from $\hat{x}(k|k)$, $P(k|k)$, and $y(k+1)$. First, we compute $\hat{x}(k+1|k)$ and $P(k+1|k)$.

$$\begin{aligned} \hat{x}(k+1|k) &= \mathbb{E}\left[x(k+1) | \mathbf{y_k} \right] \\ &= \mathbb{E}\left[\left(\sum_{n=1}^{N^2} Z_n(k) \bar{A}_n \right) x(k) + w(k) | \mathbf{y_k} \right] \\ &= \left(\sum_{n=1}^{N^2} p_n \bar{A}_n \right) \hat{x}(k|k), \end{aligned}$$

where $p_n = P(Z_n(k) = 1)$. Let $A(k) = \sum_{n=1}^{N^2} Z_n(k) \bar{A}_n$ and $\hat{A} = \sum_{n=1}^{N^2} p_n \bar{A}_n$.

$$\begin{aligned} P(k+1|k) &= \mathbb{E}\left[e(k+1|k) e(k+1|k)^T | \mathbf{y_k} \right] \\ &= \mathbb{E}[A(k) x(k) (A(k) x(k))^T | \mathbf{y_k}] \\ &- \hat{A} \hat{x}(k|k) (\hat{A} \hat{x}(k|k))^T + G Q G^T \end{aligned}$$

Since $\mathbb{E}[Z_n(k) Z_n(k)] = p_n$ and $\mathbb{E}[Z_n(k) Z_m(k)] = p_n p_m$ for $m \neq n$,

$$\begin{aligned} &\mathbb{E}[A(k) x(k) (A(k) x(k))^T | \mathbf{y_k}] \\ &\quad = \mathbb{E}[\textstyle\sum_{n=1}^{N^2} \sum_{m=1}^{N^2} Z_n(k) Z_m(k) \bar{A}_n x(k) x(k)^T \bar{A}_m^T | \mathbf{y_k}] \\ &\quad = \textstyle\sum_{n=1}^{N^2} \sum_{m=1}^{N^2} \mathbb{E}[Z_n(k) Z_m(k)] \bar{A}_n \mathbb{E}[x(k) x(k)^T | \mathbf{y_k}] \bar{A}_m^T \\ &\quad = \textstyle\sum_{n=1}^{N^2} p_n \bar{A}_n \mathbb{E}[x(k) x(k)^T | \mathbf{y_k}] \bar{A}_n^T \\ &\qquad + \textstyle\sum_{n=1}^{N^2} \sum_{m=1, m \neq n}^{N^2} p_n p_m \bar{A}_n \mathbb{E}[x(k) x(k)^T | \mathbf{y_k}] \bar{A}_m^T. \end{aligned}$$

On the other hand,

$$\hat{A}\hat{x}(k|k)(\hat{A}\hat{x}(k|k))^T = \sum_{n=1}^{N^2}\sum_{m=1}^{N^2} p_n p_m \bar{A}_n \hat{x}(k|k)\hat{x}(k|k)^T \bar{A}_m^T.$$

Combining previous two results into the equation for $P(k+1|k)$, we get

$$\begin{aligned}
&P(k+1|k) \\
&= GQG^T + \textstyle\sum_{n=1}^{N^2} p_n \bar{A}_n \mathbb{E}[x(k)x(k)^T|\mathbf{y_k}]\bar{A}_n^T \\
&\quad + \textstyle\sum_{n=1}^{N^2}\sum_{m=1,m\neq n}^{N^2} p_n p_m \bar{A}_n \mathbb{E}[x(k)x(k)^T|\mathbf{y_k}]\bar{A}_m^T \\
&\quad - \textstyle\sum_{n=1}^{N^2}\sum_{m=1}^{N^2} p_n p_m \bar{A}_n \hat{x}(k|k)\hat{x}(k|k)^T \bar{A}_m^T \\
&= GQG^T + \textstyle\sum_{n=1}^{N^2} p_n \bar{A}_n \mathbb{E}[x(k)x(k)^T|\mathbf{y_k}]\bar{A}_n^T \\
&\quad - \textstyle\sum_{n=1}^{N^2} p_n^2 \bar{A}_n \mathbb{E}[x(k)x(k)^T|\mathbf{y_k}]\bar{A}_n^T \\
&\quad + \textstyle\sum_{n=1}^{N^2}\sum_{m=1}^{N^2} p_n p_m \bar{A}_n P(k|k)\bar{A}_m^T \\
&= GQG^T + \textstyle\sum_{n=1}^{N^2} p_n(1-p_n)\bar{A}_n \mathbb{E}[x(k)x(k)^T|\mathbf{y_k}]\bar{A}_n^T \\
&\quad + \textstyle\sum_{n=1}^{N^2}\sum_{m=1}^{N^2} p_n p_m \bar{A}_n P(k|k)\bar{A}_m^T.
\end{aligned}$$

We can also write it as

$$\begin{aligned}
&P(k+1|k) \\
&= GQG^T + \hat{A}P(k|k)\hat{A}^T \\
&\quad + \textstyle\sum_{n=1}^{N^2} p_n(1-p_n)\bar{A}_n(P(k|k) + \hat{x}(k|k)\hat{x}(k|k)^T)\bar{A}_n^T.
\end{aligned}$$

Given $\hat{x}(k+1|k)$ and $P(k+1|k)$, $\hat{x}(k+1|k+1)$ and $P(k+1|k+1)$ are computed as in the regular Kalman filter.

$$\begin{aligned}
&\hat{x}(k+1|k+1) \\
&\quad = \hat{x}(k+1|k) + K(k+1)(y(k+1) - C\hat{x}(k+1|k)) \\
&P(k+1|k+1) \\
&\quad = P(k+1|k) - K(k+1)CP(k+1|k),
\end{aligned}$$

where $K(k+1) = P(k+1|k)C^T(CP(k+1|k)C^T + R)^{-1}$.

B. KF for General DNCS

Now let us consider the general DNCS dynamic model (6) with the measurement model described in (7). We have

$$\begin{aligned}
\hat{x}(k+1|k) &= \mathbb{E}\left[x(k+1)|\mathbf{y_k}\right] \\
&= \mathbb{E}\left[A(Z)x(k) + Gw(k)|\mathbf{y_k}\right] \\
&= \hat{A}\hat{x}(k|k),
\end{aligned}$$

where

$$\hat{A} = \sum_{z\in\mathcal{Z}} p_z A(z)$$

is the expected value of $A(Z)$. Here, $p_z = P(Z = z)$, and $\mathcal{Z}$ is a set of all possible outcome vectors for Z.

The prediction covariance can be computed as following.

$$\begin{aligned}
&P(k+1|k) \\
&= \mathbb{E}[e(k+1|k)e(k+1|k)^T|\mathbf{y_k}] \\
&= \mathbb{E}[A(Z)x(k)x(k)^T A(Z)^T|\mathbf{y_k}] \\
&\quad -\hat{A}\hat{x}(k|k)\hat{x}(k|k)^T\hat{A}^T + GQG^T \\
&= \textstyle\sum_{z\in\mathcal{Z}} p_z A(z)\mathbb{E}[x(k)x(k)^T|\mathbf{y_k}]A(z)^T \\
&\quad -\hat{A}\hat{x}(k|k)\hat{x}(k|k)^T\hat{A}^T + GQG^T \\
&= GQG^T + \textstyle\sum_{z\in\mathcal{Z}} p_z A(z)P(k|k)A(z)^T \\
&\quad + \textstyle\sum_{z\in\mathcal{Z}} p_z A(z)\hat{x}(k|k)\hat{x}(k|k)^T(A(z) - \hat{A})^T.
\end{aligned}$$

Lastly, $\hat{x}(k+1|k+1)$ and $P(k+1|k+1)$ are computed as shown in the simple DNCS case.

C. Simulation Results

The simulation results of the modified Kalman filtering algorithms are summarized in this section. We first compare the performance of the KF for the simple DNCS system against independent KFs. Then we compare the performance of the KF for the general DNCS system against independent KFs.

1) Simple DNCS: Consider the simple DNCS dynamic model (4) and the measurement model (7). Let $N = 3$ and the state vector of each agent is $x = [x, y, \dot{x}, \dot{y}]^T$, where (x, y) and $(\dot{x}, \dot{y})$ are the position and the velocity components of the vehicle along the x and y axes, respectively. For $i = 1, 2, 3$,

$$A_{ii} = \begin{bmatrix} 1 & 0 & \delta & 0 \\ 0 & 1 & 0 & \delta \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad G_i = \begin{bmatrix} \delta/2 & 0 \\ 0 & \delta/2 \\ \delta & 0 \\ 0 & \delta \end{bmatrix},$$

where δ is the sampling interval, while

$$A_{12} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0.1 \end{bmatrix},$$

$A_{13} = -A12$, $A_{21} = -A12$, $A_{23} = A12$, $A_{31} = A12$, and $A_{32} = -A12$. The communication matrix is

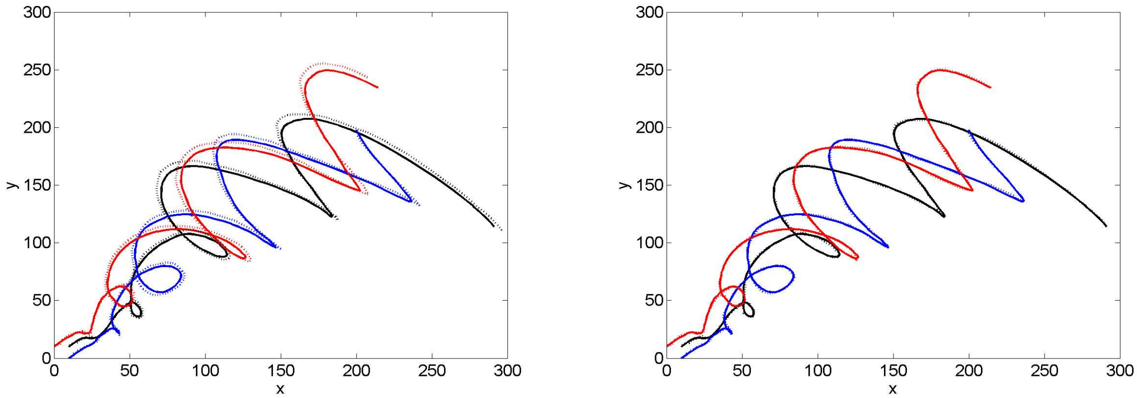
$$P_{\text{com}} = \begin{bmatrix} 1 & 0.75 & 0.75 \\ 0.75 & 1 & 0.75 \\ 0.75 & 0.75 & 1 \end{bmatrix}.$$

The measurement matrix $C = [c_{i,j}]$ is a $2N \times 4N$ matrix, where all entries are zeroes except $c_{1,1} = c_{2,2} = c_{3,5} = c_{4,6} = c_{5,9} = c_{6,10} = 1$. So we are observing only the positions of the agents. In addition, $\delta = 1$, $Q_i = \text{diag}(0.01, 0.01)$, and R is a $2N \times 2N$ identity matrix. The initial states of the agents are $x_1(0) = [10, 10, 1, 1]^T$, $x_2(0) = [10, 0, 1, 1]^T$, and $x_3(0) = [0, 10, 1, 1]^T$. The simulation is run from $k = 0$ to $k = 200$.

The performance of the modified KF for the simple DNCS is compared against N independent KFs where each KF is used to track each agent. The trajectories and estimates from both approaches are shown in Figure 1. The mean square error (MSE) of N independent KFs was 7.57 while the MSE of the modified KF for the simple DNCS was 1.85. This example illustrates how the knowledge of the communication matrix can improve the state estimation when the communication links are lossy. Also notice that due to the interaction among agents in the DNCS dynamic model, the dynamics of each agent shows high nonlinearity.

2) General DNCS: This is an example inspired by the model by Vicsek *et al.* [11]. Consider a general DNCS system (6) consisting of $N = 5$ agents. The same state vector as in Section III-C.1 is assumed for each agent. The communication matrix is

$$P_{\text{com}} = \begin{bmatrix} 1 & 0.967 & 0 & 0 & 0.141 \\ 0.337 & 1 & 0.248 & 0 & 0 \\ 0 & 0.732 & 1 & 0.426 & 0 \\ 0 & 0 & 0.088 & 1 & 0.135 \\ 0.189 & 0 & 0 & 0.046 & 1 \end{bmatrix}.$$



(a) (b)

Fig. 1. (a) Trajectories of the agents are shown in solid lines and estimates from N independent KFs are shown in dotted lines. (b) Trajectories of the agents are shown in solid lines and estimates from the modified KF for the simple DNCS are shown in dotted lines. The estimates of the modified KF almost overlap with the actual trajectories. (This figure is best viewed in color.)

The dynamics of the agent i is

$$x_i(k+1) = \sum_{j=i-1}^{i+1} A_{\kappa(j)i}(Z) x_{\kappa(j)}(k) + G_i w_i(k),$$

where $\kappa(j) = (j-1 \mod N) + 1$. For $\kappa(j) = i$,

$$A_{ii}(Z) = \begin{bmatrix} 1 & 0 & \delta & 0 \\ 0 & 1 & 0 & \delta \\ 0 & 0 & 1/S(i) & 0 \\ 0 & 0 & 0 & 1/S(i) \end{bmatrix},$$

and, for $\kappa(j) \neq i$,

$$A_{\kappa(j)i}(Z) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & Z_{\kappa(j)i}/S(i) & 0 \\ 0 & 0 & 0 & Z_{\kappa(j)i}/S(i) \end{bmatrix}$$

with $S(i) = \sum_{m=i-1}^{i+1} Z_{\kappa(m)i}$. Hence, when the agent i communicates with its neighboring agents $\kappa(i-1)$ and $\kappa(i+1)$, its new velocity is the average of its velocity and velocities received from its neighboring agents.

The measurement matrix $C = [c_{i,j}]$ is a $(2 \cdot 3) \times 4N$ matrix, where all entries are zeroes except $c_{1,1} = c_{2,2} = c_{3,9} = c_{4,10} = c_{5,13} = c_{6,14} = 1$. So we are observing the positions of agents 1, 3 and 4 and the positions of agents 2 and 5 are not observed. In addition, $\delta = 1$, $Q_i = \text{diag}(0.04, 0.04)$, and $R = \text{diag}(25, 25, 25, 25, 25, 25)$. The initial states of the agents are $x_1(0) = [50, 50, 1, 1]^T$, $x_2(0) = [50, 0, -0.9, 0.989]^T$, $x_3(0) = [0, -50, 0.227, 0.665]^T$, $x_4(0) = [-50, 0, 0.161, -0.188]^T$, and $x_5(0) = [0, 50, -0.83, 0.617]^T$. The simulation is run from $k = 0$ to $k = 500$.

We compared the performance of the modified KF for the general DNCS against the regular KF which assumes no lossy links. The trajectories and estimated by KFs are shown in Figures 2. Figure 2(a) shows the estimates from the regular KF which assumes no lossy links, Figure 2(b) shows the estimates from the modified KF for the general DNCS. The MSE of the KF which assumes no lossy links was 64.89 while the MSE of the modified KF for the general DNCS was 22.12. Notice that although the positions of the agent 2 and 5 were not observed, the KF designed for the general DNCS was able to estimate their positions better.

IV. Stabilizing Communication Control

In this section, we consider the problem of finding a communication control which stabilizes the general DNCS (6) for given $\{A(z) : z \in \mathcal{Z}\}$, *i.e.*, finding a communication matrix P_{com} such that the general DNCS (6) is stable. For example, in wireless communication, one can control the transmission power to increase or decrease entries of P_{com}. We use the stability results for jump linear systems by Costa and Fragoso [15]. We use the notation $A \succ 0$ if A is a positive definite matrix and $A \succeq 0$ if A is a positive semidefinite matrix. The spectral radius of A is denoted by $\rho(A)$.

Definition 1: The DNCS model (6) is mean square stable (MSS) if, for any initial condition x_0 and second-order independent wide sense stationary random process $\{w(k)\}$, there exist x^* and P^* independent of x_0 such that:

(a) $\| \mathbb{E}[x(k)] - x^* \| \to 0$ as $k \to \infty$
(b) $\| \mathbb{E}[x(k)x(k)^T] - P^* \| \to 0$ as $k \to \infty$.

Theorem 1 (Corollary 1 of [15]): The DNCS model (6) is MSS if and only if there exists $G \succ 0$ such that

$$G - \sum_{z=\mathcal{Z}} p_z A(z)^T G A(z) \succ 0.$$

Theorem 2: The DNCS model (6) is MSS if

$$\sum_{z \in \mathcal{Z}} p_z \rho(A(z)^T A(z)) < 1.$$

Proof: Fix $\alpha > 0$ and let $G = \alpha I_n$ where $n = N n_x$ and I_n is a $n \times n$ identity matrix. Clearly, $G \succ 0$.

$$\begin{aligned} & G - \textstyle\sum_{z=\mathcal{Z}} p_z A(z)^T G A(z) \\ & = \alpha I_n - \alpha \textstyle\sum_{z=\mathcal{Z}} p_z A(z)^T A(z) \\ & \succeq \alpha \left(1 - \textstyle\sum_{z=\mathcal{Z}} p_z \rho(A(z)^T A(z))\right) I_n \\ & \succ 0, \end{aligned}$$

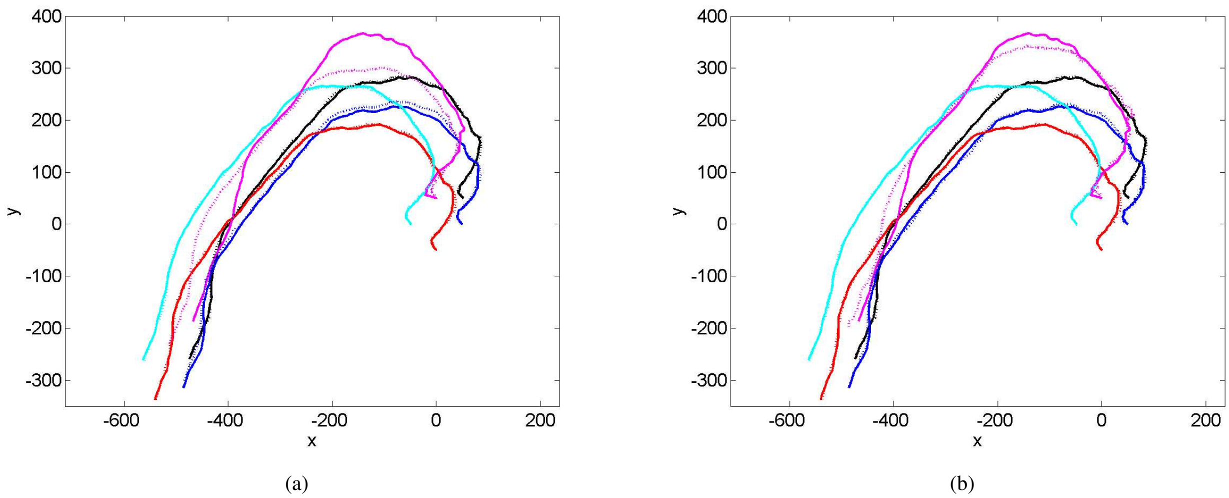


Fig. 2. (a) Trajectories of the agents are shown in solid lines and estimates from the KF which assumes no lossy links are shown in dotted lines. (b) Trajectories of the agents are shown in solid lines and estimates from the modified KF for the general DNCS are shown in dotted lines. (This figure is best viewed in color.)

since $\sum_{z=\mathcal{Z}} p_z \rho(A(z)^T A(z)) < 1$. Hence, by Theorem 1, (6) is MSS. ■

Using Theorem 2, one can easily check if there exists a stabilizing communication control. Let $\rho(z) = \rho(A(z)^T A(z))$ and consider the following linear programming (LP) problem.

$$\begin{array}{ll} \text{minimize} & c = \sum_{z\in\mathcal{Z}} p_z \rho(z) \\ \text{subject to} & \sum_{z\in\mathcal{Z}} p_z = 1 \\ & 0 \le p_z \le 1, \qquad z \in \mathcal{Z}. \end{array} \tag{8}$$

Notice that we can add restrictions on p_z to reflect physical constraints in the DNCS. If there exists a feasible solution with $c < 1$, we know for sure that there exists a stabilizing communication control based on Theorem 2. However, it is important to note that Theorem 2 is only a sufficient condition. Hence, when the LP (8) does not have a feasible solution with $c < 1$, we can not say there is no communication control which stabilizes the DNCS model (6).

Example 1: Consider a 2-agent DNCS system where

$$A_{11} = \begin{bmatrix} -.138 & .414 \\ .598 & .219 \end{bmatrix} \quad A_{12} = \begin{bmatrix} -.075 & -.006 \\ -.505 & -.34 \end{bmatrix}$$
$$A_{21} = \begin{bmatrix} -.35 & -.245 \\ -.495 & .049 \end{bmatrix} \quad A_{22} = \begin{bmatrix} .185 & -.137 \\ -.298 & -.653 \end{bmatrix}.$$

Note that $Z_1 = Z_{11} = 1$, $Z_2 = Z_{12}$, $Z_3 = Z_{21}$, and $Z_4 = Z_{22} = 1$. Let $z_1 = [1,0,0,1]^T$, $z_2 = [1,0,1,1]^T$, $z_3 = [1,1,0,1]^T$, and $z_4 = [1,1,1,1]^T$. Then $\mathcal{Z} = \{z_1, z_2, z_3, z_4\}$ and $\rho(z_1) = 0.517$, $\rho(z_2) = 1.038$, $\rho(z_3) = 1.044$, and $\rho(z_4) = 0.925$. We also have constraints on p_z: $0 \le p_{z_1} \le 0.6$, $0.1 \le p_{z_2} \le 0.5$, $0.1 \le p_{z_3} \le 0.5$, and $0.1 \le p_{z_4} \le 0.3$. Since not all ρ are less than 1, it is not clear that the DNCS system is MSS with the constraints on p_z. By solving the LP (8) for this problem, one finds that there is a feasible solution: $p^* = [p_{z_1}, p_{z_2}, p_{z_3}, p_{z_4}]^T = [0.6, 0.1, 0.1, 0.2]^T$ with $c = 0.704$. Hence, the DNCS system is MSS with p^*.

Now consider the same example as before except $1.5A_{22}$ is used instead of A_{22}. Then $\rho(z_1) = 1.164$, $\rho(z_2) = 1.56$, $\rho(z_3) = 1.558$, and $\rho(z_4) = 1.531$ and there is no feasible solution to the LP (8) with $c < 1$ and the system is not MSS. The state evolutions of these two systems are shown in Figure 3. □

The LP (8) is an efficient way to check the existence of a stable communication control. But it does not provide the solution in the form we want. We want to find the communication matrix P_{com}, not $\{p_z\}$. For the notational convenience, we again use the index $n \in \{1, \ldots, M = N^2\}$ described in Section II, where ij is indexed by $n = N(i-1)+j$. Then

$$\begin{aligned} p_z &= \prod_{i,j} p_{ij}^{z_{N(i-1)+j}} (1-p_{ij})^{1-z_{N(i-1)+j}} \\ &= \prod_{n}^{M} p_n^{z_n} (1-p_n)^{1-z_n}. \end{aligned}$$

The problem we want to solve is:

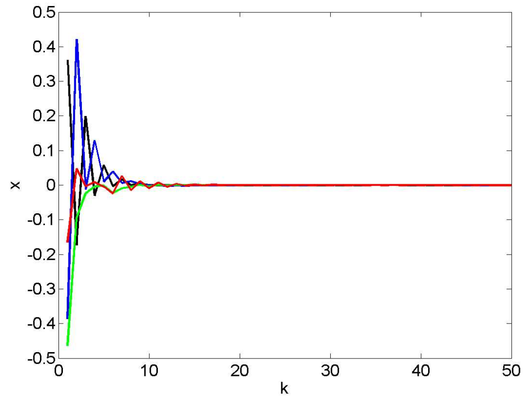
$$\begin{array}{ll} \text{find} & \{p_n\} \\ \text{subject to} & \sum_{z\in\mathcal{Z}} \rho(z) \prod_n^M p_n^{z_n} (1-p_n)^{1-z_n} < 1 \\ & 0 \le p_n \le 1, \qquad \forall n \in \{1, \ldots, M\}. \end{array} \tag{9}$$

The problem (9) is a special case of signomial programming which is non-convex and only a locally optimal solution can be computed efficiently [17]. Instead of solving the problem (9) directly, we relax the problem and use geometric programming (GP), which is a convex optimization problem [16], [17].

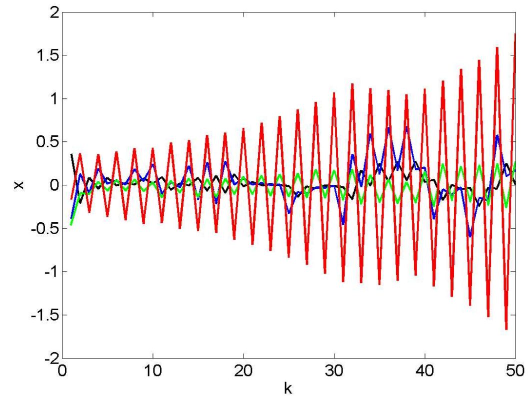
The relaxed GP is

$$\begin{array}{ll} \max & \frac{1}{\epsilon} \prod_{m=1}^{2M} q_n \\ \text{subject to} & \sum_{z\in\mathcal{Z}} \rho(z) \prod_n^M q_{2(n-1)+1}^{z_n} q_{2n}^{1-z_n} + \epsilon \le 1 \\ & q_{2(n-1)+1} + q_{2n} \le 1, \qquad \forall n \in \{1, \ldots, M\} \\ & 0 \le q_m \le 1, \qquad \forall m \in \{1, \ldots, 2M\} \end{array} \tag{10}$$

As in the LP (8), we can add restrictions on q_n to reflect physical constraints in the DNCS. Using the GP (10), we



(a)



(b)

Fig. 3. (a) The state evolution of the MSS system given in Example 1. (b) The state evolution of the non-MSS system given in Example 1.

can find ranges of P_{com} by running (10) multiple times with different upper bounds on q_m.

Example 2: Consider the DNCS system described in Example 1 but replace A_{22} by $1.35A_{22}$. When the upper bounds $p_{12} \leq 0.4$ and $p_{21} \leq 0.4$ are used, the GP finds $q = [0.4, 0.6, 0.4, 0.6]^T$ and $\sum_{z \in \mathcal{Z}} p_z \rho(z) = 0.987$. Since $q_1 + q_2 = q_3 + q_4 = 1$, this is a feasible communication control. However, when the upper bounds $p_{12} \leq 1$ and $p_{21} \leq 1$ are used, the GP finds $q = [0.426, 0.571, 0.436, 0.562]^T$. But $q_1 + q_2 \neq 1$ and $q_3 + q_4 \neq 1$, hence, this is not a feasible communication control. □

V. Conclusions and Future Work

In this paper, we have described a distributed networked control system (DNCS) consisting of multiple agents communicating over a lossy communication channel, *e.g.* wireless channel. A DNCS is an extension of an NCS to model a distributed multi-agent system such as the Vicsek model. Optimal linear filtering algorithms based on the Kalman filter are developed to estimate the states of DNCSs. Due to the interaction among agents in the DNCS dynamic model, the dynamics of each agent can show high nonlinearity. But the filtering algorithm was able to estimate the states correctly using the knowledge of the communication matrix. In the second part of this paper, the stabilizing communication control problem is described, where one finds the acceptable ranges of packet loss rates at which the overall system is stable. We developed algorithms based on convex optimization to check the existence of stabilizing communication control and for solving the stabilizing communication control problem.

We have assumed that the communication matrix is independent from other parameters. In our future work, we will relax this assumption by making the matrix a function of the states of the agents. This is a valid model for wireless communication, since the transmission power decreases with an increasing distance between a transmitter and a receiver. We are currently extending the stabilizing communication control problem using the necessary condition for stability.

References

[1] D. Culler, D. Estrin, and M. Srivastava, "Overview of sensor networks," *IEEE Computer, Special Issue in Sensor Networks*, Aug. 2004.

[2] B. Grocholsky, V. Kumar, and H. Durrant-Whyte, "Anonymous cooperation in robotic sensor networks," in *Proc. of the AAAI-04 Workshop on Sensor Networks*, 2004.

[3] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. Jordan, and S. Sastry, "Kalman filtering with intermittent observations," *IEEE Trans. Automatic Control*, vol. 49, no. 9, pp. 1453–1464, 2004.

[4] Y. Xu and J. Hespanha, "Estimation under uncontrolled and controlled communications in networked control systems," in *Proc. of the 44th IEEE Conference on Decision and Control*, Seville, Spain, Dec. 2005.

[5] L. Shi, M. Epstein, A. Tiwari, and R. M. Murray, "Estimation with information loss: Asymptotic analysis and error bounds," in *Proc. of the 44th IEEE Conference on Decision and Control*, Seville, Spain, Dec. 2005.

[6] J. Nilsson and B. Bernhardsson, "LQG control over a Markov communication network," in *Proc. of the 36th IEEE Conference on Decision and Control*, Dec. 1997.

[7] O. Imer, S. Yuksel, and T. Basar, "Optimal control of dynamical systems over unreliable communication links," in *Proc. of the NOLCOS*, Stutgart, Germany, 2004.

[8] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, and S. Sastry, "Optimal control with unreliable communication: the tcp case," in *Proc. of the American Control Conference*, Portland, OR, June 2005.

[9] W. Zhang, M. Branicky, and S. Phillips, "Stability of networked control systems," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 84–96, 2001.

[10] D. Liberzon, "On stabilization of linear systems with limited information," *IEEE Trans. Automatic Control*, vol. 48, no. 2, pp. 304–307, 2003.

[11] T. Vicsek, A. Czirok, E. B. Jacob, I. Cohen, and O. Schochet, "Novel type of phase transitions in a system of self-driven particles," *Physical Review Letters*, vol. 75, no. 6, pp. 1226–1229, 1995.

[12] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.

[13] V. Blondel, J. Hendrickx, A. Olshevsky, and J. Tsitsiklis, "Convergence in multiagent coordination, consensus, and flocking," in *Proc. of the 44th IEEE Conference on Decision and Control*, Seville, Spain, Dec. 2005.

[14] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[15] O. Costa and M. Fragoso, "Stability results for discrete-time linear systems with Markovian jumping parameters," *Journal of Mathematical Analysis and Applications*, vol. 179, pp. 154–178, 1993.

[16] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[17] S. Boyd, S. Kim, L. Vandenberghe, and A. Hassibi, "A tutorial on geometric programming," Stanford University EE Technical Report, Tech. Rep., 2005.