2011 American Control Conference
on O'Farrell Street, San Francisco, CA, USA
June 29 - July 01, 2011

Observer-based Output Tracking Control for A Class of Linear Networked Control Systems

Dawei Zhang and Qing-Long Han*

Abstract— **This paper is concerned with the problem of model reference tracking control for a class of linear networked control systems (NCSs) in which a controlled plant is connected to an observer-based controller via a communication network. In the presence of network-induced delays and packet dropouts in the sensor-to-controller and controller-to-actuator connections, the inputs of the plant and the controller are updated in an asynchronous way. In this case, the resulting NCS is equivalent to a linear system with two interval time-varying delays. A sufficient stability condition that ensures the NCS with an H_∞ tracking performance is derived by using a Lyapunov-Krasovskii functional approach. Due to the asynchronous input errors, a separation principle cannot be used to design the observer-based controller. A novel design algorithm of tracking control is presented by using the stability condition and a particle swarm optimization (PSO) technique. The effectiveness of the algorithm is illustrated by a numerical example.**

I. INTRODUCTION

Output tracking control has many industrial applications such as flight control, robot control, motor control and so on. Generally speaking, the objective of tracking control is to drive the outputs of a controlled plant to follow those of a reference model or some predefined trajectories as close as possible. In many modern industrial systems, system components (the controlled plant, sensors, actuators and controllers) are often located in different physical places. To exchange information among these system components, a shared communication network is used to interconnect the plant and the controller. Such control systems with the control loop closed through a network are called networked control systems (NCSs) [1]-[2]. NCSs have exhibited a wide range of applications due to their advantages such as flexible deployment, low cost and easy maintenance. The existing research on NCSs is mainly focused on two aspects, i.e., design of network protocols and system architectures [3], [4] and design of performance requirements on NCSs [5]. In particular, much work has been done to deal with stability and stabilization for NCSs [6]-[8]. In recent years, tracking control for NCSs has been increasingly highlighted since the insertion of a network between a physical plant and a controller enables an execution of remote tracking control.

This work was supported in part by the Australian Research Council Discovery Projects under Grant DP1096780 and Grant DP0986376; the Research Advancement Awards Scheme Program (January 2010-December 2012) and the RDI Merit Grant Scheme Project under Gant RDIM1109 (January 2011-December 2011) at Central Queensland University, Australia.

Dawei Zhang and Qing-Long Han are with the Centre for Intelligent and Networked Systems, and the School of Information and Communication Technology, Central Queensland University, Rockhampton QLD 4702, Australia, Tel: +61 7 4930 9270, Fax: +61 7 4930 9729, Email: q.han@cqu.edu.au

For point-to-point wired systems, serval methods have been presented to investigate the output tracking control, for example, an H_∞ tracking control strategy [9]-[11]. Such a strategy is extended to design the state feedback tracking controller for NCSs in [12]-[14]. More specifically, Gao and Chen (2008) apply continuous-time systems with an interval time-varying delay to describe the NCSs and analyze an H_∞ tracking performance by a Lyapunov-Krasovskii functional method [12]. They derive some existence conditions of the tracking controller in terms of linear matrix inequalities (LMIs). Wang and Yang (2008) formulate linear NCSs with a constant or a time-varying sampling period as two kinds of augmented discrete-time systems and consider the H_∞ model reference tracking control [13]. Jia *et al.* (2009) employ a T-S fuzzy model to represent the nonlinear NCSs and design a controller to guarantee the tracking error systems with a desired H_∞ tracking performance [14]. Different from the previous H_∞ method, Van de Wouw *et al.* introduce an input-to-state stability property to achieve a tracking performance for network-based tracking error systems modeled by a discretization technique and a delay impulsive approach, respectively [15]. It should be pointed out that all the states of the controlled plants in [12]-[15] are assumed to be measurable. In fact, it is impossible or prohibitively expensive to measure all of the process variables in many practical situations. Moreover, the state feedback controllers in [12]-[15] depend only on network-induced delays and packet dropouts in the sensor-to-actuator channel because they are designed by specific state information that successfully drives the actuator. Therefore, the first concern of this paper is to study the observer-based tracking control for NCSs by taking the sensor-to-controller channel and the controller-to-actuator channel into account.

It is not uncommon to design an observer-based controller for hardwired systems by a separation principle. To use the separation principle, the matrix variables in Lyapunov functionals are usually set to be diagonal, which introduces some conservatism [9]-[11]. For NCSs with an observer-based controller, some stability and stabilization results are reported in recent years [16]-[18]. Naghshtabrizi and Hespanha [16] characterize the network-induced delays and packet dropouts that occur in the sensor-to-controller and the controller-to-actuator channels by two independent interval time-varying delays. They introduce a numerical procedure to design an observer-based controller by means of some non-convex matrix inequalities. Seuret et al. [17] present a GPS technique to ensure that the control inputs of a controlled plant and an observer-based controller in NCSs are synchronous, but different from the observer inputs. In this framework, control gain and observer gain are first solved separately; then these gains are checked by a stability condition of the closed-loop system [17]. When there is no GPS synchronization, Seuret et al. address the stability of NCSs in [18] and reveal that the separation principle cannot be used to design the control gains and observer gains. Accordingly, another concern of this paper is to develop an observer-based tracking control design method without using the separation principle, which can determine the control gains and observer gains by solving an optimization problem of an H_∞ tracking performance.

Notice that some heuristic search methods play a key role in solving complex design optimization problems. A typical method is the Genetic Algorithm (GA), which was introduced in the mid 1970s by John Holland. Recently, GA has been extended to design the delay-dependent controller [19], [20]. In [19], the implementation architecture of NCSs is established via Profibus-DP protocols and a PID tuning of remote controller is designed by GA [19]. Du and Zhang show the merits of GA in finding solutions of delay-dependent conditions by some numerical examples [20]. Another well-known stochastic optimization algorithm is the Particle Swarm Optimization (PSO) technique invented by Kennedy and Eberhart in the mid 1990s [21]-[22]. PSO is a population-based optimization algorithm which is inspired by the social behavior of animals such as fish schooling and birds flocking. This algorithm has been well studied because of its easy implementation, stable convergence characteristic and computational efficiency. However, the potentials of PSO in finding the solutions of a network-based controller for an NCS has not been explored. In this study, we will develop a new strategy to design the observer-based controller by a PSO algorithm with the feasibility of the LMI-based stability condition without using the separation principle.

II. PROBLEM STATEMENT

Consider the following reference model

$$\begin{cases} \dot{x}_r(t) = A_r x_r(t) + B_r r(t), \\ y_r(t) = C_r x_r(t), \end{cases} \tag{1}$$

where $x_r(t) \in \mathbb{R}^n$ is the state vector, $r(t) \in \mathbb{R}^r$ is the energy bounded input vector and $y_r(t) \in \mathbb{R}^q$ is the output vector, respectively. A_r, B_r and C_r are constant matrices with appropriate dimensions. It is assumed that A_r is Hurwitz and $x_r(t)$ is measurable to be used for control signal.

The controlled plant is described as follows

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + D\omega(t), \\ y(t) = Cx(t), \end{cases} \tag{2}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, $y(t) \in \mathbb{R}^q$ and $\omega(t) \in L_2[0,\infty)$ are the state, the input, the output, the external disturbance, respectively. A, B, C and D are real constant matrices. $x(t_0) = x_0$ is the initial condition.

Suppose that a communication network is used to connect the plant (2) and the following observer-based controller

$$\begin{cases} \dot{\hat{x}}(t) = A\hat{x}(t) + B\hat{u}(t) + L(y(t) - \hat{y}(t)), \\ \hat{y}(t) = C\hat{x}(t), \\ \hat{u}(t) = F(\hat{x}(t) - x_r(t)), \end{cases} \tag{3}$$

where $\hat{x}(t) \in \mathbb{R}^n$ is the state estimate vector, $\hat{u}(t) \in \mathbb{R}^m$ is the input vector, and $\hat{y}(t) \in \mathbb{R}^q$ is the output vector. The observer gain L and the control gain F are to be determined. First, $y(kh)$ and $x_r(kh)$ are augmented as a single packet and transferred over the sensor-to-controller channel, where h is a sampling period. After the sensor-to-controller delay $\tau^{sc}_{b_k}$, $y(b_k h)$ and $x_r(b_k h)$ are available update the following event-driven controller (3) on the interval $[b_k h + \tau^{sc}_{b_k},\ b_{k+1}h + \tau^{sc}_{b_{k+1}})$

$$\begin{cases} \dot{\hat{x}}(t) = A\hat{x}(t) + B\hat{u}(t) + L(y(b_k h) - \hat{y}(t)), \\ \hat{y}(t) = C\hat{x}(b_k h), \\ \hat{u}(t) = \tilde{u}(b_k h) = F(\hat{x}(b_k h) - x_r(b_k h)), \end{cases} \tag{4}$$

where b_k $(k = 1, 2, \ldots,)$ are time-stamps of packets that successfully reach the controller. Notice that the controller outputs the control signals in the update instants instead of re-sampling itself due to the redundant samplings of $x_r(t)$. Similarly, the control signal $\tilde{u}(l_k h)$ is available to input the controlled plant after the controller-to-actuator delay $\tau^{ca}_{l_k}$, where l_k $(k = 1, 2, \ldots,)$ are time-stamps of control signals received by the actuator. Then the actuator holds the signal until next update. Clearly, on $[l_k h + \tau_{l_k},\ l_{k+1}h + \tau_{l_{k+1}})$,

$$u(t) = \tilde{u}(l_k h) = F(\hat{x}(l_k h) - x_r(l_k h)). \tag{5}$$

Let $\tau_1(t) = t - b_k h$ for $t \in [b_k h + \tau^{sc}_{b_k},\ b_{k+1}h + \tau^{sc}_{b_{k+1}})$ and $\tau_2(t) = t - l_k h$ for $t \in [l_k h + \tau_{l_k},\ l_{k+1}h + \tau_{l_{k+1}})$. One obtains

$$\tau^{sc}_{b_k} \le \tau_1(t) \le (b_{k+1} - b_k)h + \tau^{sc}_{b_{k+1}},\ k = 1, 2, \ldots \tag{6}$$

$$\tau_{l_k} \le \tau_2(t) \le (l_{k+1} - l_k)h + \tau_{l_{k+1}},\quad k = 1, 2, \ldots. \tag{7}$$

Remark 1: Packet dropouts may occur in both the sensor-to-controller channel and the controller-to-actuator channel. So one obtains $\{l_k\}_{k=1}^{\infty} \subseteq \{b_k\}_{k=1}^{\infty} \subseteq Z^+$, and Z^+ denotes the set of positive integers. Both the controller (4) and the actuator (5) are assumed to recognize and drop outdated data actively. On $[l_k h + \tau_{l_k},\ l_{k+1}h + \tau_{l_{k+1}})$, the actuator (5) holds $\tilde{u}(l_k h)$ while the controller (4) may witness more than one update. Specifically, the controller (4) holds the signal $y(b_{m^{(k-1)}}h)$ and $x_r(b_{m^{(k-1)}})$ on $[l_k h + \tau_{l_k},\ b_{m^k+1}h + \tau^{sc}_{b_{m^k+1}})$ and are updated by the signals $y(b_{m^k+1}h)$ and $x_r(b_{m^k+1}h)$, $\cdots$, $y(b_{m^k+d^k}h)$ and $x_r(b_{m^k+d^k}h)$ on $[b_{m^k+1}h + \tau^{sc}_{b_{m^k+1}},\ b_{m^k+2}h + \tau^{sc}_{b_{m^k+2}})$, $\cdots$, $[b_{m^k+d^k}h + \tau^{sc}_{b_{m^k+d^k}},\ l_{k+1}h + \tau_{l_{k+1}})$, respectively, where $l_k \le b_{m^{(k-1)}}$, $m^{k+1} = m^k + d^k$, and m^0, m^k, d^k, $k \in Z^+$. Define $\tau_m = \min_{i \in Z^+}\{\tau^{sc}_{b_i}\}$, $\tau_{1M} = \max_{i \in Z^+}\{(b_{i+1} - b_i)h + \tau^{sc}_{b_{i+1}}\}$ and $\tau_{2M} = \max_{k \in Z^+}\{(l_{k+1} - l_k)h + \tau_{l_{k+1}}\}$. Then, for $t \in [l_k h + \tau_{l_k},\ l_{k+1}h + \tau_{l_{k+1}})$, we have

$$\begin{cases} 0 \le \tau_m \le \tau_1(t) \le \tau_{1M}, \\ 0 \le \tau_m \le \tau_1(t) \le \tau_2(t) \le \tau_{2M}. \end{cases} \tag{8}$$

where τ_m is the lower bound of network-induced delays, τ_{1M} and τ_{2M} can be viewed as the synthetical index involving information about network-induced delays and packet dropouts in the sensor-to-controller and sensor-to-actuator channels.

Remark 2: The sensor-to-controller delay and controller-to-actuator delay can be lumped together τ_{l_k} for NCSs with fixed controllers in [6], [12], [13] and [15]. But this is not the case for NCSs with an observer-based controller because of

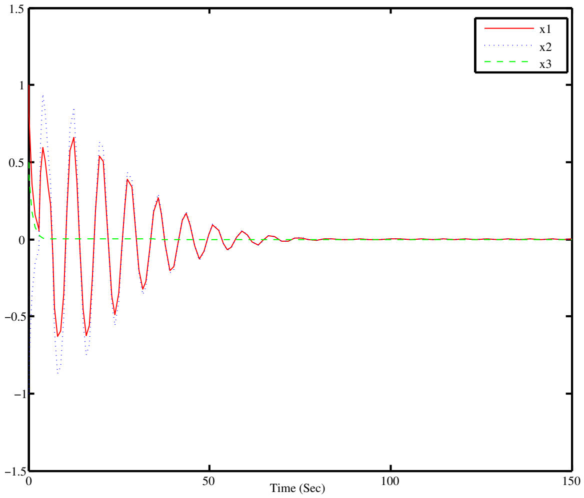

Fig. 1. The response of the system (10) with $\tau_1 = \tau_2 = 3s$

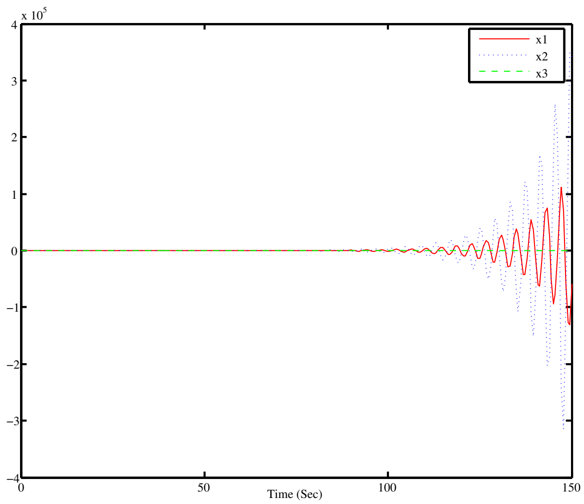

Fig. 2. The response of the system (10) with $\tau_1 = 1.5s$ and $\tau_2 = 3s$

the difference between $\hat{u}(t)$ and $u(t)$: different packets may be available at the controller and the actuator since packet dropouts may occur in the controller-to-actuator channel; due to the controller-to-actuator delay $\tau^{ca}_{l_k}$, the data stamped by $l_k h$ cannot update synchronously the actuator and the controller at the time instant $l_k h + \tau_{l_k}$.

Then, the closed-loop system can be given by

$$\begin{cases} \dot{\xi}(t) = \bar{A}\xi(t) + \bar{B}_1\xi(t-\tau_1(t)) + \bar{B}_2\xi(t-\tau_2(t)) + \bar{D}\bar{\omega}(t), \\ e(t) = \bar{C}\xi(t),\ t \in [l_k h + \tau_{l_k}, l_{k+1}h + \tau_{l_{k+1}}),\ \forall k \in Z^+, \end{cases} \quad (9)$$

where

$$\bar{A} = \begin{bmatrix} A & 0 & 0 \\ 0 & A & 0 \\ 0 & 0 & A_r \end{bmatrix}, \bar{B}_1 = \begin{bmatrix} 0 & 0 & 0 \\ LC & BF-LC & -BF \\ 0 & 0 & 0 \end{bmatrix},$$

$$\bar{B}_2 = \begin{bmatrix} 0 & BF & -BF \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \bar{D} = \begin{bmatrix} D & 0 \\ 0 & 0 \\ 0 & B_r \end{bmatrix}, \bar{C} = \begin{bmatrix} C^T \\ 0 \\ -C_r^T \end{bmatrix}^T,$$

$$\xi(t) = \begin{bmatrix} x^T(t) & \hat{x}^T(t) & x_r^T(t) \end{bmatrix}^T, \bar{\omega}(t) = \begin{bmatrix} \omega^T(t) & r^T(t) \end{bmatrix}^T.$$

The initial condition of the state $x(t)$ on $[t_0 - \tau_{2M},\ t_0]$ is supplemented by $x(t) = \psi(t),\ t \in [t_0 - \tau_{2M},\ t_0]$. Then the initial states of the augmented system NCS can be given by $\xi(t) = \phi(t) = [\psi^T(t)\ 0]^T$ with $\phi(t_0) = \xi_{t_0}$.

Remark 3: Some techniques such as network protocols including round-trip acknowledgement signals [8] and a state predictor [16] are introduced to implement a synchronized control input in the controlled plant (2) and the controller (4), i.e., $\tau_1(t) = \tau_2(t)$ for $t \in [l_k h + \tau_{l_k}, l_{k+1}h + \tau_{l_{k+1}}),\ \forall k \in Z^+$. Then an interesting question arises: can the controller designed for the system (9) with $\tau_1(t) = \tau_2(t)$ guarantee the stability of the system (9) with $\tau_1(t) \le \tau_2(t)$? In particular, we exhibit the answer to this question by a simple numerical example. A linear system with two constant delays (a special case of the system (9)) is given by

$$\dot{\xi}(t) = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \xi(t) + \begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix} \xi(t-\tau_1) + \begin{bmatrix} 0 & -1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \xi(t-\tau_2). \quad (10)$$

The initial state is set to be $\xi(0) = [1\ -1\ 0.5]^T$. Fig. 1. and Fig. 2 show the responses of the system (10) with $\tau_1 = \tau_2 = 3s$ and $\tau_1 = 1.5s$ and $\tau_2 = 3s$, respectively. One can clearly see that the system (10) with $\tau_1 = \tau_2 = 3s$ is asymptotically stable but unstable even τ_1 is less than the delay τ_2. Thus, the synchronization technique in [8] and [16] cannot be used in this paper and the condition (8) must be ensured in the stability analysis and controller synthesis for the system (9).

Throughout this paper, the following H_∞ tracking performance for the system (9) is required [9]-[14].

$$\int_{t_0}^{t_f} e^T(t) M e(t) dt \le V(\xi_0) + \gamma^2 \int_{t_0}^{t_f} \bar{\omega}^T(t)\bar{\omega}(t), \quad (11)$$

where $\gamma > 0$ is the tracking level, $M > 0$ is the weighting matrix, and $V(\xi_0)$ is the energy function of initial states.

The purpose of this paper is to analyze the H_∞ tracking performance (11) and to design the observer-based controller (4) for the system (9).

III. MAIN RESULTS

In this section, a sufficient condition on the existence of the tracking controller for the system (9) is first derived by a Lyapunov-Krasovskii functional method. The stability condition is given by the following proposition.

Proposition 1: For given positive scalars τ_m, τ_{1M}, τ_{2M} and γ, gain matrices F and L, the system (9) is asymptotically stable with the H_∞ tracking performance γ if there exist matrices $P > 0$, $Q_i > 0$ and $R_i > 0$ $(i = 1, 2, 3)$ such that

$$\Psi = \begin{bmatrix} \Psi_{11} & \Psi_{12} \\ * & \Psi_{22} \end{bmatrix} < 0, \quad (12)$$

where

$$\Psi_{11} = \begin{bmatrix} \Psi(1,1) & P\bar{B}_1 & P\bar{B}_2 & R_1 \\ * & -2R_2 - 2R_3 & R_3 & R_2 + R_3 \\ * & * & -2R_3 & 0 \\ * & * & * & \Psi(4,4) \end{bmatrix},$$

$$\Psi_{12}=\begin{bmatrix} 0 & 0 & P\bar{D} & \bar{A}^T\Lambda \\ R_2 & 0 & 0 & \bar{B}_1^T\Lambda \\ 0 & R_3 & 0 & \bar{B}_2^T\Lambda \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\Psi_{22}=\begin{bmatrix} Q_3-Q_2-R_2 & 0 & 0 & 0 \\ * & -Q_3-R_3 & 0 & 0 \\ * & * & -\gamma^2 I & \bar{D}^T\Lambda \\ * & * & * & -\Lambda \end{bmatrix},$$

$$\Lambda=\tau_m^2R_1+(\tau_{1M}-\tau_m)^2R_2+(\tau_{2M}-\tau_m)^2R_3,$$
$$\Psi(1,1)=PA+A^TP+Q_1+\bar{C}^TM\bar{C}-R_1,$$
$$\Psi(4,4)=Q_2-Q_1-R_1-R_2-R_3.$$

Proof: First, we consider the stability of the system (9) with $\bar{\omega}(t)=0$. Choose the following Lyapunov-Krasovskii functional including information about τ_m, τ_{1M} and τ_{2M}.

$$\begin{aligned} V(\xi_t) &= \xi^T(t)P\xi(t)+\int_{t-\tau_m}^{t}\xi^T(s)Q_1\xi(s)ds \\ &+\int_{t-\tau_{1M}}^{t-\tau_m}\xi^T(s)Q_2\xi(s)ds+\int_{t-\tau_{2M}}^{t-\tau_{1M}}\xi^T(s)Q_3\xi(s)ds \\ &+\int_{t-\tau_m}^{t}\int_{s}^{t}\dot{\xi}^T(\theta)\tau_mR_1\dot{\xi}(\theta)d\theta ds \\ &+\int_{t-\tau_{1M}}^{t-\tau_m}\int_{s}^{t}\dot{\xi}^T(\theta)(\tau_{1M}-\tau_m)R_2\dot{\xi}(\theta)d\theta ds \\ &+\int_{t-\tau_{2M}}^{t-\tau_m}\int_{s}^{t}\dot{\xi}^T(\theta)(\tau_{2M}-\tau_m)R_3\dot{\xi}(\theta)d\theta ds. \end{aligned} \quad (13)$$

Taking the time-derivative of $V(\xi_t)$ on $[l_kh+\tau_{l_k},\ l_{k+1}h+\tau_{l_{k+1}})$ along the trajectory of the system (9), we have

$$\begin{aligned} \dot{V}(\xi_t) &= 2\xi^T(t)P[\bar{A}\xi(t)+\sum_{i=1}^{2}\bar{B}_i\xi(t-\tau_i(t))]+\xi^T(t)Q_1\xi(t) \\ &+\xi^T(t-\tau_m)(Q_2-Q_1)\xi(t-\tau_m) \\ &+\xi^T(t-\tau_{1M})(Q_3-Q_2)\xi(t-\tau_{1M}) \\ &-\xi^T(t-\tau_{2M})Q_3\xi(t-\tau_{2M})-\int_{t-\tau_m}^{t}\dot{\xi}^T(s)\tau_mR_1\dot{\xi}(s)ds \\ &+\dot{\xi}^T(t)[\tau_m^2R_1+(\tau_{1M}-\tau_m)^2R_2+(\tau_{2M}-\tau_m)^2R_3]\dot{\xi}(t) \\ &-\int_{t-\tau_{1M}}^{t-\tau_m}\dot{\xi}^T(s)(\tau_{1M}-\tau_m)R_2\dot{\xi}(s)ds \\ &-\int_{t-\tau_{2M}}^{t-\tau_m}\dot{\xi}^T(s)(\tau_{2M}-\tau_m)R_3\dot{\xi}(s)ds. \end{aligned}$$

Considering $\bar{C}^TM\bar{C}+P\bar{D}\bar{D}^TP/\gamma^2\geq 0$ and the LMI (12) by the schur complement technique, we can conclude the asymptotical stability for the system (9) with $\bar{\omega}(t)=0$ by following the similar analysis method in [7].

Next, under the zero initial condition, we consider the H_∞ tracking performance (11) for all nonzero $\bar{\omega}(t)\in L_2[t_0,\ \infty)$. For $[l_kh+\tau_{l_k},\ l_{k+1}h+\tau_{l_{k+1}})$, it follows from (11) that

$$\begin{aligned} \int_{l_kh+\tau_{l_k}}^{t}e^T(s)Me(s)ds &= V(\xi_{l_kh+\tau_{l_k}})-V(\xi_t) \\ &+\int_{l_kh+\tau_{l_k}}^{t}(\dot{V}(\xi_s)+e^T(s)Me(s))ds. \end{aligned} \quad (14)$$

Define $t_f=l_{T+1}h+\tau_{l_{T+1}}$, where T is the time stamp that the last control signal successfully arrived the actuator. And we have $\bigcup_{k=0}^{k=T}[l_kh+\tau_{l_k},\ l_{k+1}h+\tau_{l_{k+1}})=[t_0,\ t_f)$. Then one can see the H_∞ tracking performance (11) is ensured in consideration of the continuity of the LKF $V(\xi_t)$ on $[t_0,\ t_f)$, which completes the proof.

Now, we are in a position to design an observer-based controller for the system (9). It is a routine way to use a separation principle to solve the control gain and observer gain for traditional systems. However, due to the synchronization errors introduced by communication networks, $x(t)$ and $\hat{x}(t)$ in (9) are interconnected so that F and L are coupled in $\bar{B}_1$. As a result, the separation principle does not work for the system (9). In this paper, an algorithm which utilizes the random search of PSO and the feasible solution of the LMI-based stability condition, is presented to obtain the optimal tracking performance γ and the corresponding feedback gains F and L. In the PSO technique, the particle status are characterized by two factors: its position and velocity, which are updated by the following equations [21]-[22]

$$\begin{aligned} v_{id}(k+1) &= \omega\cdot v_{id}(k)+c_1\cdot rand(\ )\cdot(p_{id}(k)-x_{id}(k)) \\ &\quad +c_2\cdot rand(\ )\cdot(p_{gd}(k)-x_{id}(k)), \end{aligned} \quad (15)$$
$$f_{id}(k+1) = f_{id}(k)+v_{id}(k+1), \quad (16)$$
$$\omega = (\omega_M-\omega_m)(m_\eta-c_\eta)/m_\eta+\omega_m, \quad (17)$$

where $v_{id}(k)$ is the d^{th} dimensional velocity $(d=1,2,\ldots,l)$ of the i^{th} particle $(i=1,2,\ldots,N_p)$ at the discrete-time index k; f_{id} and p_{id} are the current position and previous best position of the i^{th} particle, respectively; p_{gd} is the global best position; $rand()$ is a uniformly distributed random variables lied on [0, 1]; c_1 and c_2 are two acceleration coefficients; ω is the inertia weight; ω_M and ω_m represent the maximum and minimum inertia weight respectively; m_η and c_η denote the maximum iteration and the current iteration, respectively.

The PSO algorithm is given as follows.

Procedure 1:

Step 1: Initialization

1.1) Randomly initialize the population of N_p particles. Each particle consists of the elements the feedback matrices F, L and scalar γ, and these elements f_i lie in the range $[\alpha_i,\beta_i]$ $(i=1,2,\ldots,mn+1)$.

1.2) Initialize the parameters c_1, c_2, ω_M, ω_m, v_{max} and iteration number.

Step 2: Repeat until a given maximum number of iteration

2.1) Evaluate the individual fitness. First, decode individual produced in Step 1.1) to obtain the F_j, and L_j $(j=1,2,...,N_p)$. Second, search the γ for every F_j and L_j satisfying the LMI-based stability condition. And take every γ as the objective value corresponding to F_j and L_j.

2.2) Store the global best particle and its fitness.

2.3) Store the previous best particles and their fitnesses.

2.4) Update the velocity and position according to (15).

Step 3: Obtain the minimum γ and the corresponding feedback gains F_j and L_j from the global best particle.

Remark 4: The convergence speed of Procedure 1 mainly depends on the search space $[\alpha_i,\beta_i]$. How to determine the orders of magnitude (α_i and β_i), is vital to the control design. Usually, one can refer to the expert knowledge database, the other is in virtue of the following traditional control strategy.

If no network exists between the controlled plant and the controller we have the following hardwired system from (9)

$$\begin{cases} \dot{\xi}(t) = (\bar{A}+\bar{B})\xi(t) + \bar{D}\bar{\omega}(t), \\ e(t) = \bar{C}\xi(t), \end{cases} \tag{18}$$

where $\bar{B} = \bar{B}_1 + \bar{B}_2$.

Based on the separation principle and the result in [11], we have the following proposition.

Proposition 2: For given scalars $\gamma > 0$, $\varepsilon > 0$ and a weighting matrix $M > 0$, the system (18) is asymptotically stable and satisfies the H_∞ tracking performance γ if there exist matrices $X_i > 0$ $(i = 1,2,3)$ such that $\Theta < 0$, where

$$\Theta = \begin{bmatrix} -2\varepsilon X_1 & 0 & 0 & -(\bar{B}Y)^T & 0 & 0 & \varepsilon I & 0 \\ * & -2\varepsilon & 0 & D^T & 0 & 0 & 0 & \varepsilon I \\ * & * & \Theta_{33} & \Theta_{34} & X_3 B_r & \Theta_{36} & 0 & 0 \\ * & * & * & \Theta_{44} & -B_r & X_1 C^T & 0 & 0 \\ * & * & * & * & -\gamma^2 I & 0 & 0 & 0 \\ * & * & * & * & * & -M^{-1} & 0 & 0 \\ * & * & * & * & * & * & \Theta_{77} & X_2 D \\ * & * & * & * & * & * & * & -\gamma^2 I \end{bmatrix},$$

$\Theta_{33} = X_3 A_r + A_r^T X_3$, $\Theta_{44} = \bar{A}X_1 + \bar{B}Y_1 + (\bar{A}X_1 + \bar{B}Y_1)^T$,
$\Theta_{34} = (\bar{A} - A_r)^T$, $\Theta_{36} = (\bar{C} - C_r)^T$,
$\Theta_{77} = X_2\bar{A} + \bar{A}^T X_2 - Z\bar{C} - \bar{C}^T Z^T$.

Moreover, if the above condition is feasible, the gains are obtained by $F = YX_1^{-1}$ and $L = X_2^{-1}Z$, respectively.

Proof: The proof is a routine case and omitted.

IV. NUMERICAL EXAMPLES

Example 1: The controlled plant is borrowed from [17]

$$\begin{cases} \dot{x}(t) = \begin{bmatrix} 0 & 1 \\ 0 & -11.32 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 11.32 \end{bmatrix} u(t), \\ y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t). \end{cases} \tag{19}$$

The reference model is given as follows [11]

$$\begin{cases} \dot{x}_r(t) = \begin{bmatrix} 0 & 1 \\ -6 & -5 \end{bmatrix} x_r(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r(t), \\ y_r(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x_r(t). \end{cases} \tag{20}$$

Applying Proposition 2 with $M = 1$ and $\varepsilon = 0.5$, one can obtain the minimum tracking index $\gamma = 0.51$, the control gain $F_1 = [-1.1441 \ \ -0.0983]$ and the observer gain $L_1 = [0.2182 \ 6.5645]^T$. Suppose the network setting I to be $\tau_m = 60(ms)$, $\tau_{1M} = 100(ms)$, $\tau_{2M} = 200(ms)$ and the scalars in the algorithm $N_p = 20$, $c1 = 2$, $c2 = 2$, $\omega_M = 0.7$ and $\omega_m = 0.4$, which satisfy the convergence of criterion [22]. Then $\gamma_{min} = 0.0934$, $F_2 = [-3.2061 \ -0.46114]$ and $L_2 = [9.9825 \ 9.9574]^T$ are solved by Procedure 1 with Proposition 1.

In simulation, the sampling period is $h = 20(ms)$ and initial states are $x(0) = \hat{x}(0) = [0.5 \ \ 0]$, $x_r(0) = [-0.5 \ \ -2]$. Moreover, we make brief of packet dropout and network-induced delay by assuming $b_i = 2i$, $30(ms) \le \tau_{b_i}^{sc} \le 60(ms)$, $l_k = 4k$ and $30(ms) \le \tau_{l_k} \le 120(ms)$, which satisfies that $30(ms) \le \tau_1(t) \le 100(ms)$ and $30(ms) \le \tau_2(t) \le 200(ms)$. Then Fig. 3 shows the tracking response of (19)-(20) in the network setting I, where the controller with F_1 and L_1 is designed by Proposition 2. It is noted that an unsatisfactory tracking performance is obtained, though the stability of overall systems with the traditional controller (F_1 and L_1) is satisfied in the network environment. The output response of the controlled plant (19) and the reference model (20) in the network constraint I is depicted in Fig. 4. It can be shown in Fig. 4 that $y(t)$ tracks $y_r(t)$ with a satisfactory accuracy. In addition, we can calculate that

$$\frac{e^T(t)Me(t)}{r^T(t)r(t)} = 0.0101 < \gamma_{min} = 0.0934.$$

For the comparison purpose, setting the network setting II to be $\tau_m = 6(ms)$, $\tau_{1M} = 50(ms)$ and $\tau_{2M} = 100(ms)$, it follows the similar process that $\gamma = 0.04863$, $F_3 = [-9.0985 \ -0.82043]$ and $L_3 = [9.9384 \ 8.8150]^T$ are solved by Procedure 1 with Proposition 1. Applying these gains F_3 and L_3 in sim-

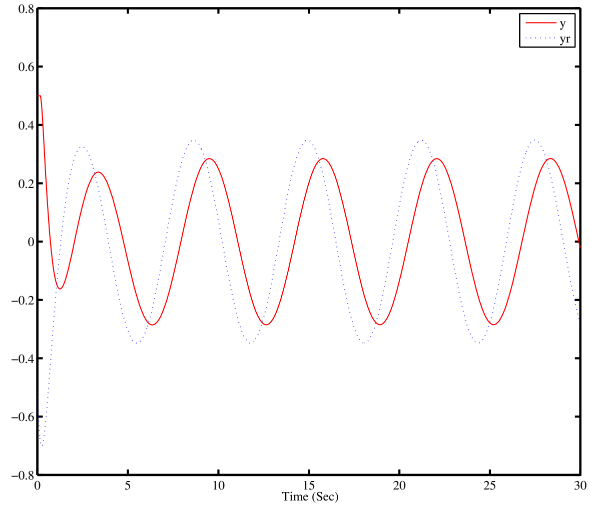

Fig. 3. The tracking response of the systems (19)-(20) by Proposition 2

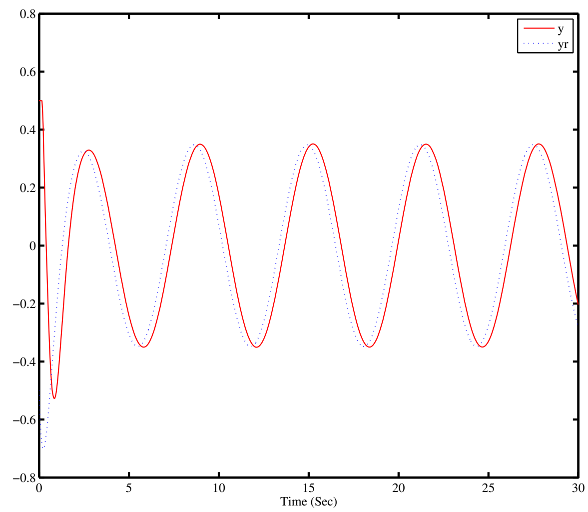

Fig. 4. The outputs of the systems (19)-(20) in the network constraint I

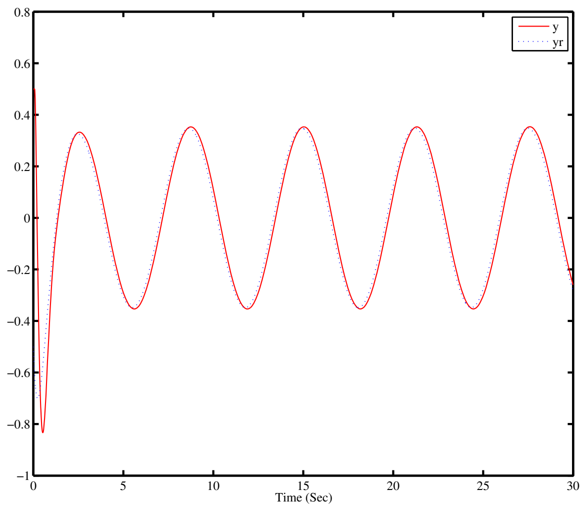

Fig. 5. The outputs of the systems (19)-(20) in the network constraint II

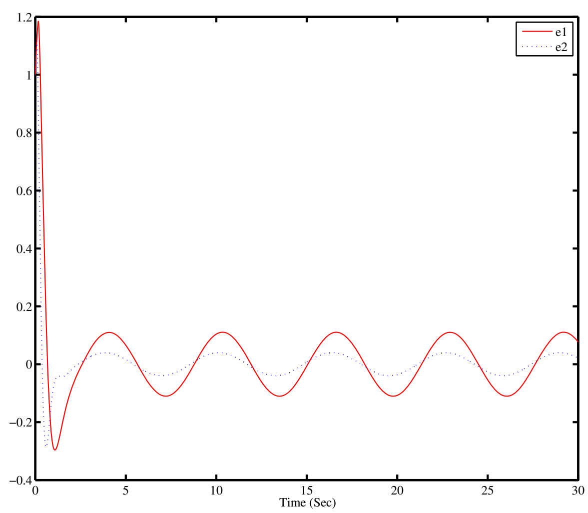

Fig. 6. The comparison of tracking errors in two network constraints

ulation, we can depict the trajectories of the controlled plant (19) and the reference model (20) in Fig. 5. Correspondingly, the comparison of tracking errors between the response by the controller with F_2 and L_2 (curve e1) and the response by the controller with F_3 and L_3 (curve e2) is given in Fig. 6. Apparently, the tracking effect is improved because of the smaller tracking index $\gamma = 0.04863$ and the higher network setting II, which explains the tradeoff between the tracking performance and the network constraint.

V. CONCLUSIONS

This paper has considered the output tracking control for a class of linear NCSs with an observer-based controller. A stability condition that ensures an H_∞ tracking performance is provided in terms of LMIs. To achieve a desired tracking performance, we have transformed the control design into an algorithm by using the PSO technique with the feasibility of LMI-based stability condition. An example has been given to show the effectiveness of the proposed method.

REFERENCES

[1] D. Hristu-Varsakelis and W. S. Levine (eds.), *Handbook of Networked and Embedded Control Systems*, Birkhauser, Boston; 2005.

[2] J. Baillieul and P. Antsaklis, Control and Communication Challenges in Networked Real-time Systems, *Proceedings of the IEEE*, vol. 95, no. 1, Jan. 2007, pp. 9-28.

[3] G.C. Goodwin, D.E. Quevedo, and E.I. Silva, Architectures and Coder Design for Networked Control Systems, *Automatica*, vol. 44, no. 1, Jan. 2008, pp. 248-257.

[4] S. Graham, G. Baliga, and P.R. Kumar, Abstractions, Architecture, Mechanisms, and A Middleware for Networked Control, *IEEE Transactions on Automatic Control*, vol. 54, no. 7, Jul. 2009, pp. 1490-1503.

[5] P. Hespanha, P. Naghshtabrizi, and Y. Xu, A Survey of Recent Results in Networked Control Systems, *Proceedings of the IEEE*, vol. 95, no. 1, Jan. 2007, pp. 138-162.

[6] W. Zhang, M.S. Branicky, and S.M. Phillips, Stability of Networked Control Systems, *IEEE Control Systems Magazine*, vol. 21, no. 1, Feb. 2001, pp. 84-99.

[7] D. Yue, Q.-L. Han, and J. Lam, Network-based Robust H_∞ Control of Systems With Uncertainty, *Automatica*, vol. 41, no. 6, Jun. 2005, pp. 999-1007.

[8] W.-A. Zhang and L. Yu, Output Feedback Stabilization of Networked Control Systems With Packet Dropouts, *IEEE Transactions on Automatic Control*, vol. 52, no. 9, Sep. 2007, pp. 1705-1710.

[9] C. S. Tseng, B. S. Chen, and H. J. Uang, Fuzzy Tracking Control Design for Nonlinear Dynamic Systems via T-S Fuzzy Model, *IEEE Transactions on Fuzzy Systems*, vol. 9, no. 3, Jun. 2001, pp. 381-392.

[10] C. Lin, Q.-G. Wang, and T. H. Lee, H_∞ Output Tracking Control for Nonlinear Systems via T-S Fuzzy Model Approach, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 36, no. 2, Apr. 2006, pp. 450-457.

[11] B. Mansouri, N. Manamanni, K. Guelton, A. Kruszewski, and T. M. Guerra, Output Feedback LMI Tracking Control Conditions With H_∞ Criterion for Uncertain and Disturbed T-S Models," *Information Sciences*, vol. 179, no. 4, Feb. 2009, pp. 446-457.

[12] H. Gao and T. Chen, Network-based H_∞ Output Tracking Control, *IEEE Transactions on Automatic Control*, vol. 53, no. 3, Apr. 2008, pp. 655-667.

[13] Y.-L. Wang and G.-H. Yang, Output Tracking Control for Networked Control Systems With Time Delay and Packet Dropout, *International Journal of Control*, vol. 81, no. 11, Nov. 2008, pp. 1709-1719.

[14] X. Jia, D. Zhang, X. Hao, and N. Zheng, Fuzzy H_∞ Tracking Control for Nonlinear Networked Control Systems in T-S Fuzzy Model, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 39, no.4, Aug. 2009, pp. 1073-1079.

[15] N. Van De Wouw, P. Naghshtabrizi, M. Cloosterman, and J.P. Hespanha, "Tracking Control for Networked Control Systems," *in 46th IEEE Conference on Decision and Control*, New Orleans, US, 2007, pp. 4441-4446.

[16] P. Naghshtabrizi and J. P. Hespanha, "Designing An Observer-based Controller for A Network Control System," *in 44th IEEE Conference on Decision and Control*, Seville, Spain, 2005, pp. 848-853.

[17] A. Seuret, F. Michaut, J.-P. Richard, and T. Divoux, "Networked Control Using GPS Synchronization," *in Proceedings of the 2006 American Control Conference*, Minneapolis, US, 2006, pp. 4195-4200.

[18] A. Seuret, "Networked Control Under Synchronization Errors," *in Proceedings of the 2008 American Control Conference*, Seattle, US, 2008, pp. 3774-3779.

[19] K.C. Lee, S. Lee, and H.H. Lee, Implementation and PID Tuning of Network-based Control Systems via Profibus Polling Network, *Computer Standards and Interfaces*, vol. 26, no. 3, May 2004, pp. 229-240.

[20] H. Du and N. Zhang, Controller Design for Time-delay Systems Using Genetic Algorithms, *Engineering Applications of Artificial Intelligence*, vol. 22, no. 3, Apr. 2009, pp. 397-400.

[21] J. Kennedy and R. Eberhart, "Particle Swarm Optimization," *in Proceedings of the 1995 IEEE International Conference on Neural Networks*, Perth, Australia, 1995, pp. 1942-1948.

[22] M. Clerc and J. Kennedy, The Particle Swarm-explosion, Stability, and Convergence in A Multidimensional Complex Space, *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, Feb. 2002, pp. 58-73.

[23] B. Kuo, *Automatic Control Systems*, Prentice Hall; 1995.