Fault Tolerant Control of Networked Control System with Fixed Data Packet Loss Rate

Liu Xiuzhi[1,2,*] and Dai Yaping[1]

[1]School of Automation, Beijing Institute of Technology, Beijing 100081, China

[2]College of Electrical Engineering and automation, Shandong University of Science and Technology, Qingdao, Shandong 266590, China

Abstract: This paper is concerned with the research on fault tolerant control (FTC) of networked control system (NCS) with fixed data packet loss rate (PLR). A synthetic model of the whole closed-loop is presented which includes time delay, packet loss of NCS and the working condition of sensors and actuators. The time delay is supposed to be shorter than sampling period and packet loss to be random but the packet loss rate is fixed. The fault of sensors and actuators are described by random variables. A sufficient stability condition in the meaning square sense is gained by constructing a useful Lyapunov function. The fault tolerant control algorithm is achieved from the stability condition. The system designed above can work effectively when sensors and actuators in normal condition and can keep stability when some sensors or actuators are in failure. A simulation example is finally given to demonstrate the effectiveness of the proposed method.

1. INTRODUCTION

Networked control systems (NCSs) are closed-loop systems in which the plants, sensors, actuators and controller communicate *via* a real-time shared media network [1]. For NCS has many advantages such as low installation and maintenance costs and reduced system wiring and increased flexibility of the system, it has been used in a wide range of areas such as industrial automation, remote control, robots and unmanned aerial vehicle *etc*. [2, 3]. Time delay and packet loss are the major problems in networked control systems [4, 5]. The research on networked control has been developing rapidly in recent years and great achievements have been gained in this subject [6, 7]. Most achievement focuses on linear networked control system [8, 9]. Some researchers concern with the nonlinear control system and have gained useful achievements [10]. Fault tolerant control is a kind of method that can keep the control system to be stable and meet the requirements of performance indicators when some parts of the system are failed [11]. During the last 30 years, fault tolerant control has gotten great development. Some researchers concentrate on the research on passive tolerant control [12, 13], and there are others who work on active fault tolerant control. [14, 15]. In this paper, we are concerned with the fault tolerant control (FTC) of NCSs with short time delay and fixed packet loss rate. Due to the development of computer technology, computer hardware and software reliability has reached a higher level.

The fault-tolerant control of networked control system mainly considers sensors' and actuators' failure.

The task of this paper is to take both the characters and the working condition of the sensors and actuators into account. Look at the networked control system as a complex object and construct a synthesis model of the whole closed loop system and design a fault tolerant control algorithm for a kind of networked control system.

Some achievements have been made in the research on FTC of NCSs with random data packet loss. When the maximum number of consecutive data packet loss is known, Dexiao Xie *et al* took the actual number of consecutive packet loss as an uncertain parameter within the maximum number of consecutive data packet loss in system modeling. The stability condition is described by a linear matrix inequality (LMI), and the control algorithm is gotten by solving LMI [16, 17]. For this method add a variable in system model. The model becomes more complex and it becomes more difficult to design control algorithm. Zhihong Huo *et al.* constructed the extended model of the system and then designed the control algorithm by using Lyapunov stability theory or Robust Control Theory [18, 19]. If the number of consecutive data packet loss is too large, this method will enlarge the dimension of the model and will lead to the heavy load of calculating amount. When the maximum number of consecutive packet loss is unknown, Jun Ren *et al.*, used the predictive control to help design the fault tolerant control algorithm [20]. With this method, the controller becomes more complex. All of the control algorithms mentioned above are conservative to some extent. In order to reduce the conservatism, J.P. Georges, *et al*, tried to use the active fault tolerant control [21]. Active fault tolerant control needs fault

*Address correspondence to this author at the School of Automation, Beijing Institute of Technology, Beijing 100081, China; Tel: 15011053453; E-mail: l_xiuzhi@sina.com
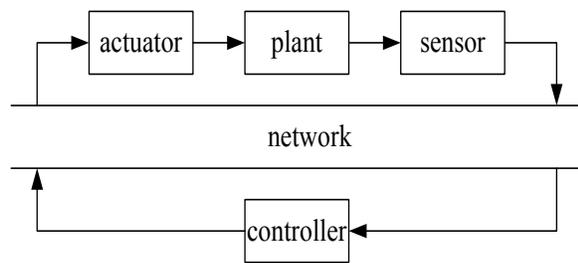
Fig. (1). Block diagram of network control.

diagnosis first and the designing of the control algorithm depends on the exact results of diagnosis. Zhixue Zhang, *et al.*, constructed fuzzy model [22]. Wei Li and Luo Xiao yu-an, *et al.*, assumed that the system parameters are uncertain and designed the robust controllers to ensure the stability even when the sensors or actuators are a failure [23, 24]. Jianguo Yao, *et al*, designed Simplex architecture to accomplish tolerant control software design and implementation [25]. Zuoyou Yin *et al.*, employed T-S fuzzy model to give sufficient condition for the asymptotic stability of closed-loop system based on Lyapunov stability theory and linear matrix inequality method to design fault tolerant algorithm [26]. Adjustable dead bands are explored by Huo Zhihong *et al* as a solution to reduce network traffic in networked control systems and a new modeling method was presented, based on this model, fault tolerant control algorithm was designed [27]. A new switched model based on probability is proposed by ChunXi Yang *et al.* to research problems of reliable control when actuators become ageing or partially disabled [28]. Active fault tolerant control method has been researched in networked control. Luo Xiaoyuan, *et a.l,* did the work via Predictive Control method [29]. A fault tolerant controller based on PI tracking control is designed by Lina Yao and Bo Peng to make the post-fault probability density function still track the given distribution, based on the fault diagnosis information [30]. Andrea Paoli, *et al.*, have researched on active fault tolerant control of discrete event systems using online diagnostics [31].

Although some achievements have been made in the research on fault-tolerant control of networked control system, but it is still in its infancy, and theory is far from mature. There are still many problems to overcome. To present there are mainly two kinds of method in research on fault-tolerant control of NCS. One is to construct the extension model, due to the existence of time delay and packet loss, the dimension will be too large. The other is to use intelligent control method, but this will result in complexity of algorithm. All these factors make the application of the fault-tolerant control method in the actual system.

Motivated by the above observations, in this paper, we intend to find a new way to design fault tolerant control algorithm for networked control system. We use "packet loss rate (PLR)" to characterize the networked control system instead of the maximum number of consecutive data packet loss to reduce the dimension of the model of the system. Suppose that the time delay is shorter than sampling period and describe data packet loss with random variables.

The condition of sensors and actuators are described by diagonal matrix *M* and *N* respectively. Taking time delay, packet loss and faults of sensors and actuators into account, we present the synthesis discrete-time model of the closed-loop system. By constructing a Lyapunov function, the stability of the networked fault tolerant control system is confirmed. Meanwhile the networked fault tolerant controller is designed.

2. SYSTEM MODELING

2.1. Description of the Networked Control System

A typical structure of networked control system is shown as Fig. (**1**).

Same as the traditional local computer control system, networked control system is composed of the plant, the sensor, a controller and the actuator. For a stabilized system, the final state of the object is determined by the reference input. The difference between the two kind of systems is that the data of the network control system is transmitted via network. The sensors detect the state of the object plant, to send it to the controller via network, then the controller gives control signals based on the control algorithm and the control signals are transmitted to actuators via network. Due to that signals are transmitted to long distance and the bandwidth and the quality of network are limited, whereas time delay and packet loss are inevitable. For a certain network, time delay is determined by transmission distance. Due to the uncertainty of the network, the packet loss is random.

Taking all factors mentioned above into account, we build the synthesis model of the NCS as following:

The model of the plant is shown as equation (1).

$$\dot{x}(t){=}\overline{A}x(t){+}\overline{B}u(t) \quad y(t){=}\overline{C}x(t) \tag{1}$$

Among them, x, u and y represent the plant state, control input and output of the plant respectively. $\overline{A}, \overline{B}, \overline{C}$ represent the state matrix, input matrix and output matrix respectively. Using state feedback control, assume that the total time delay of the system is τ and sampling period of the system is T. If $\tau{<}T$ let K represent the feedback gain, then the control input is described as (2).

$$u(t){=}\begin{cases} Kx(k), & t{\in}[kh+,(k+1)h) \\ Kx(k-1), & t{\in}[kh,kh{+}\mathrm{e}) \end{cases} \tag{2}$$

Assume that the sensors are time-driven and actuators are event-driven, the discrete model of the whole system is shown as (3).

$$x(k+1){=}Ax(k){+}B_1Kx(k){+}B_2Kx(k-1) \tag{3}$$

Among them, $A{=}e^{\overline{A}h}, B_1{=}\int_0^{h-\tau} e^{\overline{A}s}Bds, B_2{=}\int_{h-\tau}^{h} e^{\overline{A}s}Bds$.

Now we take packet loss into account. Packet loss may occur in the sensor to controller(S-C) channel and may

also occur in the controller to actuator(C-A) channel; both situations influence the control input. Use the set $\psi=\left\{ \begin{array}{ccccc} t_1 & t_2 & ... & t_l & t_{l+1} & \cdots \end{array} \right\}$ to represent the update time of the control input. Assume the sampling period of the system is T, when $t_{l+1}-t_l>T$, it indicates that packet loss occurs between t_l and t_{l+1} [30], let

$$\theta_k = \begin{cases} 1 & \text{packet loss occur in S-C chanel} \\ 0 & \text{no packet loss in S-C chanel} \end{cases} \quad (4)$$

$$\varphi_k = \begin{cases} 1 & \text{packet loss occur in C–A chanel} \\ 0 & \text{no packet loss in C–A chanel} \end{cases} \quad (5)$$

Both in sensor to controller channel and in controller to actuator channel, packet loss is random. In other words, θ_k and φ_k are both random variables, and they are independent of each other.

Let $\hat{u}(k), u(k), \hat{x}(k), x(k)$ represent the plant's input, controller's output, controller's output, plant's state at moment kT respectively. Assume that the signals are transmitted by single packet, then (6) is right.

$$\begin{cases} \hat{x}(k)=(1-\theta_k)x(k)+\theta_k\hat{x}(k-1) \\ \hat{u}(k)=(1-\varphi_k)u(k)+\varphi_k\hat{u}(k-1) \end{cases} \quad (6)$$

Employing feedback control $u(k)=K\hat{x}(k)$, the control input is shown as (7).

$$\hat{u}(k)=(1-\varphi_k)(1-\theta_k)Kx(k)+(1-\varphi_k)\theta_k K\hat{x}(k-1)+\varphi_k\hat{u}(k-1) \quad (7)$$

Take both time delay and packet loss into account, the model of the closed loop system is shown as (8).

$$\begin{aligned} x(k+1) &= Ax(k)+B_1\hat{u}(k)+B_2\hat{u}(k-1) \\ &= Ax(k)+B_1((1-\varphi_k)(1-\theta_k)Kx(k) \\ &\quad +(1-\varphi_k)\theta_k K\hat{x}(k-1)+\varphi_k\hat{u}(k-1))+B_2\hat{u}(k-1) \\ &= (A+(1-\varphi_k)(1-\theta_k)B_1K)x(k) \\ &\quad +(1-\varphi_k)\theta_k B_1K\hat{x}(k-1)+(\varphi_k B_1+B_2)\hat{u}(k-1) \end{aligned} \quad (8)$$

2.2. Modeling of NCS Considering Working Condition of Sensors and Actuators

Now consider the working condition of sensors and actuators. Assume the number of the sensors is n, sensors are used to detect the plant's state, construct matrix *M* as (9).

$$M = diag\left(\begin{array}{cccc} m_1 & m_2 & \cdots & m_n \end{array} \right) \quad (9)$$

Among them,

$$m_i = \begin{cases} 1 & \text{the ith sensor is normal} \\ 0 \le m_i < 1 & \text{the ith sensor is in failure} \end{cases}, $$

$i=1,2......n$, When the sensors are working properly, the state of the plant is sent to controller *via* network. When sensors are in failure, the signal cannot be transported fully. The state that controller use to get control signal is shown as (10).

$$\hat{x}(k) = (1-\theta_k)Mx(k)+\theta_k\hat{x}(k-1) \quad (10)$$

The model of the closed loop system is shown as (11).

$$\begin{aligned} x(k+1) &= Ax(k)+B_1\hat{u}(k)+B_2\hat{u}(k-1) \\ &= Ax(k)+B_1((1-\varphi_k)(1-\theta_k)KMx(k)+(1-\varphi_k)\theta_k K\hat{x}(k-1) \\ &\quad +\varphi_k\hat{u}(k-1))+B_2\hat{u}(k-1) \\ &= (A+(1-\varphi_k)(1-\theta_k)B_1KM)x(k)+(1-\varphi_k)\theta_k B_1K\hat{x}(k-1) \\ &\quad +(\varphi_k B_1+B_2)\hat{u}(k-1) \end{aligned} \quad (11)$$

In the same way, assuming that actuator provides an m-dimensional control input for the plant, construct matrix N shown as (12).

$$N = diag\left(\begin{array}{cccc} l_1 & l_2 & \cdots & l_m \end{array} \right) \quad (12)$$

Among them,

$$l_j = \begin{cases} 1 & \text{the jth actuator is normal} \\ 0 \le l_j < 1 & \text{the jth actuator is in failure} \end{cases}$$

Then we have the real control input shown as (13).

$$\hat{u}(k) = N((1-\varphi_k)K\hat{x}(k)+\varphi_k\hat{u}(k-1)) \quad (13)$$

The model of the closed loop system is shown as (14).

$$\begin{aligned} x(k+1) &= Ax(k)+B_1\hat{u}(k)+B_2\hat{u}(k-1) \\ &= Ax(k)+B_1N((1-\varphi_k)(1-\theta_k)Kx(k)+(1-\varphi_k)\theta_k K\hat{x}(k-1) \\ &\quad +\varphi_k\hat{u}(k-1))+B_2N\hat{u}(k-1) \\ &= (A+(1-\varphi_k)(1-\theta_k)B_1NK)x(k)+(1-\varphi_k)\theta_k B_1NK\hat{x}(k-1) \\ &\quad +(\varphi_k B_1+B_2)N\hat{u}(k-1) \end{aligned} \quad (14)$$

Through the analysis above, the synthesis dynamic model of the system was established as equation (11) and (14). In both models, $x(k)$ represents the plant's state, $\hat{x}(k-1)$ and $\hat{u}(k-1)$ represents the disturbance. If $x(k)$ is convergent with the disturbance on the system, we can say that the system is stable.

3. THE EQUIVALENT MODEL OF THE SYSTEM

The models shown by (11) and (14) are not useful in controller design, so we construct the extend model.

For system (11), we construct new state vector shown as (15).

$$z(k)=\left[\begin{array}{ccc} x^T(k) & \hat{x}^T(k-1) & \hat{u}^T(k-1) \end{array} \right] \quad (15)$$

Then the expansion model of the system is shown as (16).

$$z(k+1) = \Phi_k z(k) \tag{16}$$

$$\Phi_k = \begin{bmatrix} A+(1-\varphi_k)(1-\theta_k)B_1KM & (1-\varphi_k)\theta_k B_1K & \varphi_k B_1+B_2 \\ (1-\theta_k)M & \theta_k I & 0 \\ (1-\varphi_k)(1-\theta_k)KM & (1-\varphi_k)\theta_k K & \varphi_k I \end{bmatrix}$$

$$= A_{kM} + \theta_k B_{1M} + \varphi_k B_{2M} + \varphi_k \theta_k B_{3M}$$

$$A_{kM} = \begin{bmatrix} A+B_1KM & 0 & B_2 \\ M & 0 & 0 \\ KM & 0 & 0 \end{bmatrix}, B_{1M} = \begin{bmatrix} -B_1KM & B_1K & 0 \\ -M & I & 0 \\ -KM & K & 0 \end{bmatrix},$$

$$B_{2M} = \begin{bmatrix} -B_1KM & 0 & B_1 \\ 0 & 0 & 0 \\ -KM & 0 & I \end{bmatrix}, B_{3M} = \begin{bmatrix} B_1KM & -B_1K & 0 \\ 0 & 0 & 0 \\ KM & -K & 0 \end{bmatrix}$$

For system (14), we construct new state vector shown as (17).

$$\xi(k) = \begin{bmatrix} x^T(k) & \hat{x}^T(k-1) & \hat{u}^T(k-1) \end{bmatrix} \tag{17}$$

The expansion model is shown as (18).

$$\xi_k(k+1) = \Psi_k \xi_k(k) \tag{18}$$

$$\Psi_k = \begin{bmatrix} A+(1-\varphi_k)(1-\theta_k)B_1NK & (1-\varphi_k)\theta_k B_1NK & \varphi_k B_1N+B_2N \\ (1-\theta_k)I & \theta_k I & 0 \\ (1-\varphi_k)(1-\theta_k)K & (1-\varphi_k)\theta_k K & \varphi_k I \end{bmatrix}$$

$$= A_{kN} + \theta_k B_{1N} + \varphi_k B_{2N} + \varphi_k \theta_k B_{3N}$$

$$A_{kN} = \begin{bmatrix} A+B_1NK & 0 & B_2N \\ I & 0 & 0 \\ K & 0 & 0 \end{bmatrix}, B_{1N} = \begin{bmatrix} -B_1NK & B_1NK & 0 \\ -I & I & 0 \\ -K & K & 0 \end{bmatrix},$$

$$B_{2N} = \begin{bmatrix} -B_1NK & 0 & B_1N \\ 0 & 0 & 0 \\ -K & 0 & I \end{bmatrix}, B_{3N} = \begin{bmatrix} B_1NK & -B_1NK & 0 \\ 0 & 0 & 0 \\ K & -K & 0 \end{bmatrix}$$

In expansion model, all uncertain factors of the systems are taken as State variables. These variables and the plant's state construct the state vector of the system. Obviously, systems (11) and (14) are subsystem of systems (16) and (18) respectively. When systems (16) and (18) are stable, systems (11) and (14) are also stable. So we can design FTC algorithm based on (16) and (18) instead of basing on systems (11) and (14).

4. NETWORKED FAULT TOLERANT CONTOLLER DESIGN

In this section, we work on designing fault tolerant controller for NCS with fixed packet loss rate. That is to construct Stability condition for networked control system. which is usable when sensors or actuators are normal or in failure.

4.1. Packet Loss Rate

For a certain NCS, at any time, the packet loss is random, but when time lasts long enough, the packet loss rate is fixed. The packet loss rate of the sensor to controller channel and controller to actuator channel are represented as p_1 and p_2 respectively, which are defined by equations (19) and (20).

$$p_1 = \lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} \theta_k = p_1 = E(\theta_k) \tag{19}$$

$$p_2 = \lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} \varphi_k = E(\varphi_k) \tag{20}$$

E() represents the mathematical expectation of the variable in parentheses. For a networked control system, p_1 and p_2 can be gotten by detecting enough data samples.

4.2. Stability Analysis of NCS with Fixed Packet Loss Rate

Definition 1 The system (16) or (18) is said to be stabilized in the meaning square sense if the equation (21) or (22) holds for any initial state $z_0 \in R^{2n+m}$ or $\xi_0 \in R^{2n+m}$.

$$\lim_{k\to\infty} E\{\|z_k\|^2\} = 0 \tag{21}$$

$$\lim_{k\to\infty} E\{\|\xi_k\|^2\} = 0 \tag{22}$$

Lemma 1 Assume x and y are both random variables. They are independent of each other, E() represents the Mathematical expectation function, then equation (23) holds.

$$E(xy) = E(x)E(y) \tag{23}$$

Now we analyze the stability station of the NCS with fixed packet loss. Taking the working condition of the sensors into account, we have the conclusion as follows:

Theorem 1 For a networked control system shown as (16) with the packet loss rate of S-C channel is p1 and C-A channel is p2. For any given symmetric positive definite matrix Q, if there is a unique symmetric positive solution, P makes the equation (24) to be right, then the system is stabilized in the meaning square.

$$\begin{aligned} & A_{kM} + p_1 B_{1M} + p_2 B_{2M} + p_1 p_2 B_{3M})^T \\ & P(A_{kM} + p_1 B_{1M} + p_2 B_{2M} + p_1 p_2 B_{3M}) - P = -Q \end{aligned} \tag{24}$$

Proof. The mathematical expectation of State vector z_k is $E(z_k)$, we can get the mathematical expectation of z_{k+1} by equation (25).

$$\begin{aligned} E[z_{k+1}] &= E[(A_{kM} + \theta_k B_{1M} + \varphi_k B_{2M} + \varphi_k \theta_k B_{3M})z_k] \\ &= E[(A_{kM} + \theta_k B_{1M} + \varphi_k B_{2M} + \varphi_k \theta_k B_{3M})]E[z_k] \\ &= (A_{kM} + p_1 B_{1M} + p_2 B_{2M} + p_1 p_2 B_{3M})E[z_k] \end{aligned} \tag{25}$$

Define the function V as (26),

$$V(z_k)=E^T(z_k)PE(z_k) \qquad (26)$$

As P is positive, it is obvious that (27) and (28) will hold.

$$V(z_k)|_{z_k=0}=0 \qquad (27)$$

$$V(z_k)|_{z_k\to\infty}=\infty \qquad (28)$$

Because $V(z_k)$ is the Lyapunov function of the system (16), the value of $V(z_{k+1})$ is shown as (29).

$$V(z_{k+1})=E(z_{k+1})^T PE(z_{k+1})$$
$$=E^T[z_k](A_{kM}+p_1B_{1M}+p_2B_{2M}+p_1p_2B_{3M})^T \qquad (29)$$
$$P(A_{kM}+p_1B_{1M}+p_2B_{2M}+p_1p_2B_{3M})E[z_k]$$

Then we get

$$\Delta V=V(z_{k+1})-V(z_k)$$
$$=E^T[z_k](A_{kM}+p_1B_{1M}+p_2B_{2M}+p_1p_2B_{3M})^T$$
$$P(A_{kM}+p_1B_{1M}+p_2B_{2M}+p_1p_2B_{3M})E[z_k]-E^T[z_k]PE[z_k] \quad (30)$$
$$=E^T[z_k][(A_{kM}+p_1B_{1M}+p_2B_{2M}+p_1p_2B_{3M})^T$$
$$P(A_{kM}+p_1B_{1M}+p_2B_{2M}+p_1p_2B_{3M}-P]E[z_k]$$

If there is positive matrix P that makes the matrix $(A_{kM}+p_1B_{1M}+p_2B_{2M}+p_1p_2B_{3M})^T P(A_{kM}+p_1B_{1M}+p_2B_{2M}+p_1p_2B_{3M}-P$ to be negative, then we can conclude that $\Delta V<0$, that is the system is stable.

Taking the working condition of the actuators into account, we have the conclusion as follows:

Theorem 2 For system shown as (18), with the packet loss rate of *S-C* channel is p_1 and *C-A* channel is p_2. For any given symmetric positive definite matrix *Q*, if there is a unique symmetric positive solution *P* to the equation(31), then the system is stabilized in the meaning square.

$$A_{kN}+p_1B_{1N}+p_2B_{2N}+p_1p_2B_{3N})^T$$
$$P(A_{kN}+p_1B_{1N}+p_2B_{2N}+p_1p_2B_{3N}-P=-Q \qquad (31)$$

The proof of the Theorem **2** is same as the Theorem **1**.

Remark: Equations (24) and (30) can be rewritten as other equations shown as (32) and (33).

$$F_M(A,B,C,\tau,p_1,p_2,M,K,P,Q)=0 \qquad (32)$$

$$F_N(A,B,C,\tau,p_1,p_2,N,K,P,Q)=0 \qquad (33)$$

F_M and F_N are corresponding functions. As *A, B, C,p_1, p_2* and τ are known for a system, for any reasonable values of *P* and *Q*, we can design *K* that can make system keep stable no matter sensors or actuators are normal or in failure.

Actuators are usually at the plant site. The data transported from sensor to controller and transported from controller to actuator *via* the same network, so the packet loss rate of

the two channels is the same in this situation, that is $E(\theta_k)=E(\varphi_k)=p$. Then the stability condition of the systems (16) and (18) will be described as (34) and (35) respectively.

$$(A_{kM}+pB_{1M}+pB_{2M}+p^2B_{3M})^T$$
$$P(A_{kM}+pB_{1M}+pB_{2M}+p^2)-P=-Q \qquad (34)$$

$$(A_{kN}+pB_{1N}+pB_{2N}+p^2B_{3N})^T$$
$$P(A_{kN}+pB_{1N}+pB_{2N}+p^2B_{3N})-P=-Q \qquad (35)$$

5. SIMULATION

Consider a NCS with the plant is represented by (36).

$$\dot{x}=\begin{bmatrix} 0 & 1 \\ 0 & -2 \end{bmatrix}x+\begin{bmatrix} 0 \\ 1000 \end{bmatrix}u$$
$$y=\begin{bmatrix} 1 & 0 \end{bmatrix}x \qquad (36)$$

Assume that the network induced time delay is 15ms, we take *T=30ms*, then the discrete model of the plant is as (37).

$$x(k+1)=\begin{bmatrix} 1 & 0.02912 \\ 0 & 0.9418 \end{bmatrix}x(k)+\begin{bmatrix} 0.1114 \\ 14.78 \end{bmatrix}$$
$$u(k)+\begin{bmatrix} 0.3298 \\ 14.3425 \end{bmatrix}u(k-1) \qquad (37)$$

Assume that the sensor used to detect x_2 is in failure and the packet loss rate of network is 0.2. By Theorem **1**, we can get the feedback matrix *K=[-0.1 -0.012]*. Because packet loss is random, we do several experiments to illustrate the above conclusion. A set of step response curves of *y* with the sensor in normal condition and in fault condition is shown as Fig. (**2**) and Fig. (**3**) respectively. We also draw the step response curves of *y* when the sensors in fault condition with the packet loss rate are 0.1 and 0.3 as shown in Figs. (**4 and 5**) respectively. Any divergence curve can show that the system is unstable, so we only give one curve in Fig. (**5**).

Both the steps' response to the curve of y in Fig. (**2**) and in Fig. (**3**) is convergent. It is shown that the algorithm designed with Theorem **1** can keep the whole system stable whether the sensors is in normal condition or in failure. In other words, the fault tolerant controller is effective. It is also clear that the performance shown in Fig. (**3**) is worse than that in Fig. (**2**). The overshoot becomes bigger and the adjustment time increases. The fault tolerant control can make the control system avoid danger when sensors are in failure but it cannot make the system keep up its performance.

With the same control algorithm, and the different packet loss rate, the step response curves of *y* in Fig. (**4**) are convergent and the curve in Fig. (**5**) is divergent. It is illustrated that the controller designed for the NCS with fixed packet loss rate is also useful when the real packet loss rate is lower than the packet loss rate used in designing. When the real packet loss rate is greater than the packet loss rate used in
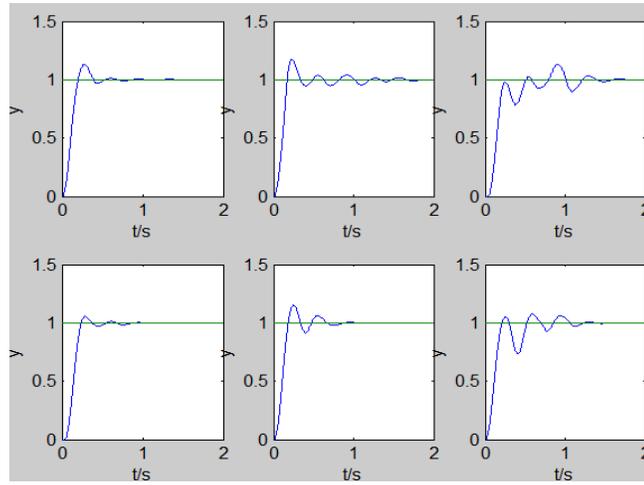
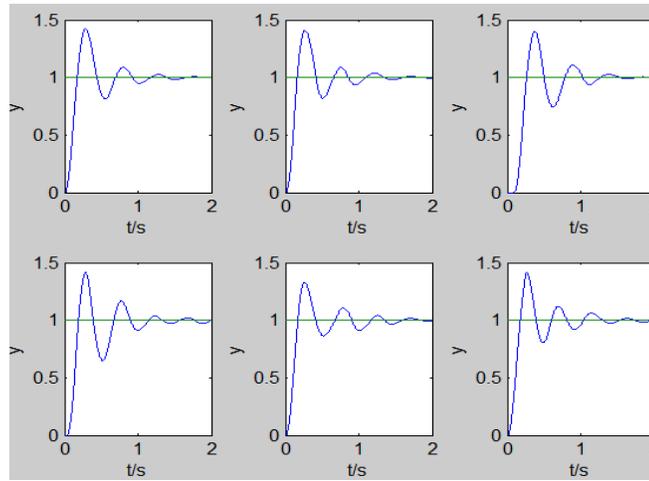**Fig. (2).** Step response curve of y (PLR=0.2, sensors are normal).

Fig. (3). Step response curve of y (PLR=0.2, sensor 2 is in failure).
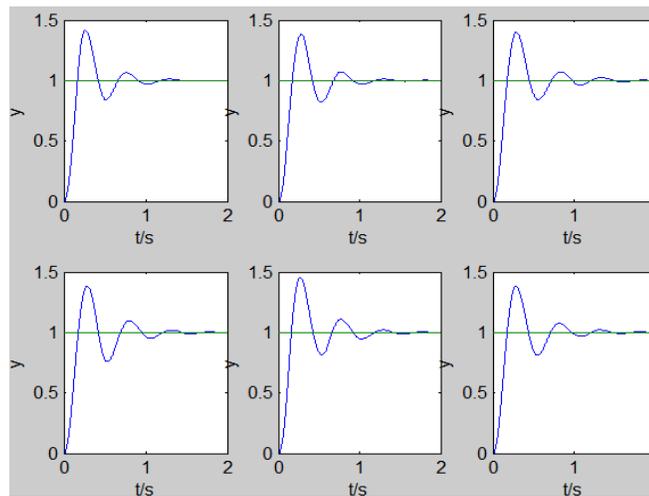

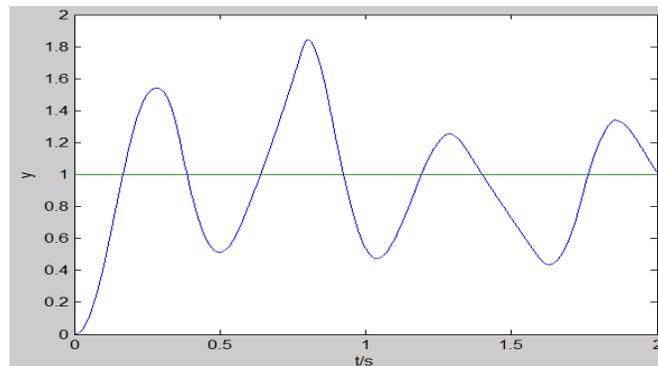**Fig. (4).** Step response curve of y (PLR=0.1, sensor 2 in failure).

Fig. (5). Step response curve of y (PLR=0.3, sensor 2 in failure).

designing, the controller will not be effective anymore, then fault tolerant controller should be redesigned. So the margin of packet loss rate of the NCS should be known when fault tolerant controller is designed.

CONCLUSION

In this paper, we researched on the fault tolerant control of networked control systems with short time delay and fixed data packet loss rate. The packet loss of the two data transmission channels are described by two random variables and the working condition of sensors and actuators are described by two diagonal matrixes. A synthesis model of the closed loop system is presented. Based on this model, we establish the condition that can make the system to be stabilized in the meaning square sense by constituting a Lyapunov function. A simulation example illustrates that the controller designed in this way is feasible and effective. Compared with the previous research, this study has simplified the system model, has solved the problems that algorithm cannot be applied due to its complexity and has very important theoretical value and practical significance in the successful application of networked control system. This research also plays a key role in improving the network fault-tolerant control theory and promotes the development and application of networked control technology. In order to better apply theory in reality, the next work is to research on the fault-tolerant control of nonlinear networked control systems.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

REFERENCES

[1]   M. C. F. Donkers, W. P. M. H. Heemels, D. Bernardini, A. Bemporad, and V. Shneer, "Stability analysis of stochastic networked control systems", *Automatica*, vol. 48, pp. 917-925, 2012.

[2]   M. B. G. Cloosterman, L. Hetel, N.van de Wouw, W. P. M. H. Heemels, J. Daafouz, and H. Nijmeijer, "Controller synthesis for networked control systems", *Automatica,* vol. 46, pp.1584-1594, 2010.

[3]   J. G. Li, J. Q. Yuan, and J. G. Lu, "Observed-based H∞ control for networked nonlinear systems with random packet losses", *ISA Transactions,* vol. 49, pp. 39-46, 2010.

[4]   Y. C. Tian, and D. Levy, "Compensation for control packet dropout in networked control systems", *Information Sciences,* vol. 178, pp. 1263-1278, 2008.

[5]   W. A. Zhang, and L. Yu, "Modelling and control of networked control systems with both network-induced delay and packet-dropout", *Automatica,* vol. 44, pp. 3206-3210, 2008.

[6]   T. Suzuki, M. Kono, N. Takahashi, and O. Sato, "Controllability and stabilizability of a networked control system with periodic communication constraints", *Systems & Control Letters,* vol. 60, pp. 977-984, 2011.

[7]   Y. Ishido, K. Takaba, and D. E. Quevedo, "Stability analysis of networked control systems subject to packet-dropouts and finite-level quantization", *Systems & Control Letters,* vol. 60, pp. 325-332, 2011.

[8]   P. V. Zhivoglyadov, and R. H. Middleton, "Networked control design for linear systems", *Automatica,* vol. 39, pp. 743-750, 2003.

[9]   Y. –L. Wang, and Q. –L. Han, "Modelling and controller design for discrete-time networked control systems with limited channels and data drift", *Information Sciences,* vol. 269, pp. 332-348, 2014.

[10]   Z. Yu, S. Li, and Hongbin Du, "Razumikhin–Nussbaum- lemma-based adaptive neural control for uncertain stochastic pure- feedback nonlinear systems with time-varying delays", *International Journal of Robust and Nonlinear Control,* vol. 23, pp. 1214-1239, 2013.

[11]   H. Niemann, and J. Stoustrup, "Passive fault tolerant control of a double inverted pendulum-a case study", *Control Engineering Practice,* vol. 13, pp. 1047-1059, 2005.

[12]   S. Hu, D. Yue, Z. Du, and J. Liu, "Reliable H1 non-uniform sampling tracking control for continuous-time non-linear systems with stochastic actuator faults", *IET Control Theory and Applications*, vol. 6, no. 1, pp. 120-129, 2012.

[13]   Z. Zuo, D. W. C. Ho, and Y. Wang, "Fault tolerant control for singular systems with actuator saturation and nonlinear perturbation", *Automatica,* vol. 46, pp. 569-576, 2010.

[14]   K. Renganathan, and V. C. Bhaskar, "An observer based approach for achieving fault diagnosis and fault tolerant control of systems modeled as hybrid Petri nets", *ISA Transactions,* vol. 50, pp. 443-453, 2011.

[15]   K. Renganathan, and V. C. Bhaskar, "An observer based approach for achieving fault diagnosis and fault tolerant control of systems modeled as hybrid Petri nets", *ISA Transactions,* vol. 50, pp. 443-453, 2011.

[16]   D. Xie, H. Huang, X. Han, and Z. Wang, "Fault-tolerant control for networked control systems with bounded packet dropout", In: *Proceedings of the 7ᵗʰ World Congress on Intelligent Control and Automation,* 2008, pp. 9297-9301.

[17]   D. Xie, and Y. Wu, "Fault-tolerant control of networked control systems with packet dropouts: switched system approach", In: *IEEE Control and Decision Conference (CCDC '09) Chinese,* 2009, pp. 5577-5581.

[18]   Z. Huo, H. Fang, Y. Zheng, and C. Xu, "Fault-tolerant control research on networked control systems with multiple-packet trans-

mission", In: *IEEE Control and Decision Conference, Chinese,* 2009, pp. 544-548.

[19]   Z. Huo, and Z. Zhang, "Research on fault-tolerant control for ncs with data packet dropout", In: *Systems and Control in Aerospace and Astronautics*, 2008, pp. 1-4.

[20]   J. Ren, C. –W. Li, and D. –Z. Zhao, "Fault-tolerant control of networked control system with time-varying transmission times", In: *IEEE Proceedings of 6th International Conference on Machine Learning Cybernetics,* 2007, vol. 1, pp. 545-549.

[21]   J. P. Georges, D. Theilliol, J. C. Ponsart, and C. Aubrun, "Networked Control System with intermittent observations:FDI/FTC design based on Interacting Multiple Model Approach", In: *17th Mediterranean Conference on Control & Automation,* 2009, pp. 528-533.

[22]   Z. Zhang, Z. Huo, and L. Zhang, "Fault-tolerant control research for networked control systems based on quasi T-S fuzzy models", In: *IEEE International Conference on Industrial Technology,* 2008, pp. 1- 4.

[23]   N. Xie, and B. Xia, "Robust fault tolerant controller design for uncertain networked control systems with fast-varying delay and packet-dropout", *In: 6th International Conference on Wireless Communications Networking and Mobile Computing,* 2010, pp. 1-4.

[24]   W. Li , H. Cao , and E. Li, "Robust guaranteed cost fault-tolerant control for uncertain ncs with fast interval time-varying delay", In:

IEEE International Conference on Mechatronics and Automation, 2010, pp. 22-28.

[25]   J. Yao, X. Liu, G. Zhu, and L. Sha, "NetSimplex: controller fault tolerance architecture in networked control systems", *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 346-356, 2013.

[26]   Z. Yin, Y. Kang, X. Chang, and S. Lun, "Robust H-infinity fault tolerant control for a class of nonlinear networked control systems based on T-S fuzzy model", In: *24th Chinese Control and Decision Conference (CCDC),* 2012, pp. 2292-2296.

[27]   H. Zhihong, Z. Yuan, and X. Chang, "A robust fault-tolerant control strategy for networked control systems", *Journal of Network and Computer Applications*, vol. 34, pp. 708-714, 2011.

[28]   C. Yang, Z. Guan, and J. Huang, "Stochastic fault tolerant control of networked control systems", *Journal of the Franklin Institute,* vol. 346, pp. 1006-1020, 2009.

[29]   L. Xiaoyuan, Z. Junling, Z. Yuyan, and G. Xinping, "Active fault-tolerant control for networked control systems with data packet dropout *via* predictive control", In: *Proceedings of the 30th Chinese Control Conference,* 2011, pp. 4241-4246.

[30]   L. Yao, and B. Peng, "Fault diagnosis and fault tolerant control for the non-Gaussian time-delayed stochastic distribution control system", *Journal of the Franklin Institute,* vol. 351, pp. 1577-1595, 2014.

[31]   A. Paoli, M. Sartini, and S. Lafortune, "Active fault tolerant control of discrete event systems using online diagnostics", *Automatica*, vol. 47, pp. 639-649, 2011.