Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Fault detection and isolation for a kind of NCSs with Markov Delays

Yue Cheng*, Hao Ye, Yongqiang Wang
Guizeng Wang, Chuanhu Ge

Dept. of Automation, Tsinghua University, Beijing, 100084, P.R. China
cheng-y@mails.tsinghua.edu.cn

Abstract: A new approach for fault detection and isolation (FDI) of Networked Control Systems (NCSs) is proposed in this paper. The paper first regards NCSs with unknown network-induced delay as a Jump Markov linear system (JMLS) with unknown input, then gives a Bayesian estimation of the JMLS based on Unknown Input Kalman filter (UIKF), and finally uses the estimation results to detect and isolate faults based on Log-likelihood ratio (LLR) approach. Simulation example shows that the method is robust to random networked induced-delay and unknown input.

1. INTRODUCTION

In recent years, more and more attention has been paid to the study of Networked Control Systems (NCSs). Networked Control System is a feed back control system wherein the control loop is closed via a real-time network (Zhang et al. 2001). In spite of many advantages of NCS, the introduction of network also brings new problems and challenges, such as network-induced delay, packet drop, and quantization problems.

Although a great many of literatures have discussed the control of NCSs (Lian et al. 2003; Zhang et al. 2005), we can find only a limited number of contributions about FDI of NCSs. As shown in (Nilsson 1998), there is no essential difference between state filter for NCSs and the classical standard Kalman filter if we can use some means, such as time stamp, to get the delay time. Therefore, it is more challenging to deal with the FDI of NCSs with unknown delay. (Ye and Ding 2004) proposed to use Taylor Expansion to approximate the influence of unknown delay and introduced a Parity Space based FD approach to generate residuals, in which the controller and actuator are event-driven, and the time delay is smaller than one sample period of the sensor. (Zheng et al. 2006) proposed a FD approach based on Takagi-Sugeno Fuzzy-Model, in which the actuator is time-driven, and the delay is an integer multiple of the sampling period.

In this paper, a new method for fault detection and isolation (FDI) of NCS with random and unknown network-induced delay and unknown input, which is supposed to be integer multiple of the sampling period, is proposed.

Since as shown in (Lin et al. 2000; Zhang et al. 2005), an NCS can be modeled as a Markov Jump Linear System (JMLS), and the state estimation problem of Markov jump linear system (JMLS) has been intensively discussed since 1970s (Ackerson and Fu, 1970, Jaffer et al. 1971, Yaakov 1978; Kim 1994), we may study the problem of FDI of NCS based on the state estimation of JMLS.

However, since NCS in this paper is assumed to have unknown input, and fault isolation is also achieved by treating some of the faults as unknown input, but all of the existing approaches for state estimation of JMLS are based on Kalman Filter, which can not deal with systems with unknown input, in this paper, Unknown Input Kalman Filter (UIKF) proposed by (Darouach et. al., 1995) will be used to give the Bayesian estimation of JMLS, based on which an approach for FDI of NCS will be further proposed. To our knowledge, so far, there is no literature that introduces the idea of UIKF into JMLS and designs a filter for a JMLS with unknown input.

2. PROBLEM FORMULATION AND PRELIMINARIES

2.1 PROBLEM FORMULATION

Assume that the sensors and the actuators are clock-driven and the controller is event-driven (Zheng et al. 2006). Because of the network-induced delay, in each sample period the actuators may receive more than one control signals from controller, whereas the actuators use the latest one. Since the actuators are clock-driven, the network-induced delay from sensors to actuators through controller is integer multiple of the sampling period. As in many literatures (e.g. (Lin et al. 2000), (Zheng et al. 2006) etc), we assume that the delay constructs a Markov chain.

By introducing system and measurement noises into the NCS model of (Zheng et al. 2006), an NCS can be modeled as follow:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k-\tau_k) + Ed(k) + Ff(k) + w(k) \\ y(k) = Cx(k) + v(k) \end{cases} \quad (1)$$

where $x(k)$ is the state vector, $u(k-\tau_k)$ is the control input vector accepted by actuators at the instant k; $d(k)$ is the unknown input vector; $y(k)$ is the plant output vector; $f(k)$ is the fault vector; $w(k)$ and $v(k)$ are system noise and measurement noise respectively, which are Gaussian white noises with known covariance W and V. Suppose that the distribution of initial state $x(0)$ is Gaussian and known, and $x(0)$ $w(k)$ and $v(k)$ are mutually independent. The network-induced delay at the instant k is τ_k periods. Suppose that τ_k is bounded, i.e. $\max(\tau_k)=s-1$, and τ_k is a discrete-time, time-homogeneous, s-state, first-order Markov chain with transition probabilities: $p_{i,j}=\Pr(\tau_{k+1}=j\mid\tau_k=j)$.

Let:

$$U(k)=\left[u(k)^T \quad \cdots \quad u(k-n)^T\right]^T; B(0)=\left[B \quad 0 \quad \cdots \quad 0\right]$$
$$B(1)=\left[0 \quad B \quad 0 \quad \cdots\right]; \cdots B(n)=\left[0 \quad \cdots \quad 0 \quad B\right]$$
$$\tau_k=r_{k+1}$$

Then the NCS model (1) can be written as

$$\begin{cases} x(k+1)=Ax(k)+B(r_{k+1})U(k)+Ed(k)+Ff(k)+w(k) \\ y(k)=Cx(k)+v(k) \end{cases} \quad (2)$$

which is a JMLS model with unknown input and fault. Our aim is to obtain the approximate a posteriori density of $x(k)$ under a fault free hypothesis, and then achieve the target of FDI.

2.2 Brief introduction to State Estimation of Jump Markov Linear System

JMLSs are linear systems whose parameters evolve with time according to a finite state Markov chain (Doucet et al. 2001). In some literatures, JMLS is also called switching linear dynamic system（SLDS）. Let r_k, $k=1,2,\cdots$, denote a discrete time s-state Markov chain with known transition probabilities matrix $\pi=[p_{ij}]$, where $p_{ij}=\Pr(r_{k+1}=j\mid r_k=j), i,j=1,\cdots s$. A JMLS can be modeled as:

$$\begin{cases} x(k+1)=A(r_{k+1})x(k)+B(r_{k+1})u(k)+F(r_{k+1})w(k) \\ y(k)=C(r_k)x(k)+G(r_k)v(k) \end{cases} \quad (3)$$

where the noise $w(k)$ and $v(k)$ are Gaussian white noise with known covariance W and V respectively. This model is a generalized form. In different applications, the system matrixes that are switching are different, for example, $F(r_k)$ and $G(r_k)$ are switching in (Ackerson and Fu, 1970); $C(r_k)$ is switching in (Jaffer et al. 1971) and (Yaakov 1978). As shown in (2), the input matrix $B(r_{k+1})$ is switching in our paper,.

The main target of using the Bayesian filter for JMLS is to recursively find the joint a posteriori density $p(x(k), r_k\mid y_{1:k})$, such as in (Ackerson and Fu, 1970; Jaffer et al. 1971; Yaakov 1978; and Kim 1994). Although there are some detailed differences in these literatures, they have the same framework and essential idea.

Let $\mu_{k,i}\triangleq\mu(k\mid I_k=i, y_{1:k})$, $\Gamma_{k,i}\triangleq\Gamma(k\mid I_k=i, y_{1:k})$, and $\alpha(k,i)$, $s_k(i)$, $i=1\cdots n(k)$, denote the known mean, covariance, the weight, and the Markov chain state of the ith Gaussian term at instant k respectively. Then the procedures in the literatures mentioned above to obtain the joint a posteriori density at instant k+1 from the density at instant k can be summarized as follow:

Step 1: Obtain $\mu'_{k+1,i,j}\triangleq\mu'(k+1\mid I_k=i, y_{1:k+1}, r_{k+1}=j)$, $\Gamma'_{k+1,i,j}\triangleq\Gamma'(k+1\mid I_{k+1}=i, y_{1:k+1}, r_{k+1}=j)$, for $i=1\cdots n(k)$, $j=1\cdots s$, at instant k+1, by using $n(k)\times s$ parallel Kalman filters, where s means the Markov chain has s states.

Step 2: Calculate the likelihoods $p(y_{k+1}\mid I_k=i, y_{1:k}, r_{k+1}=j)$, for $i=1\cdots n(k)$, $j=1\cdots s$;

Step 3: Calculate the weights $\alpha'(k+1,i,j)$ for $i=1\cdots n(k)$, $j=1\cdots s$, according to the likelihoods obtained in Step 2 and the transition probability matrix for Markov chain (i.e. π).

Step 4: Let $\Phi(\mu,\Gamma)$ denote a Gaussian function with mean μ and covariance Γ. Although the a posteriori density can be calculated by

$$p(x(k+1)\mid y_{1:k+1})=\sum_{i=1}^{n(k)}\sum_{j=1}^{s}\alpha'(k+1,i,j)\Phi(\mu'_{k+1,i,j},\Gamma'_{k+1,i,j}) \quad (4)$$

to avoid the problem of exponentially increasing computation and memory, an approximate a posteriori density can be calculated by

$$p(x(k+1)\mid y_{1:k+1})\approx\sum_{i=1}^{n(k+1)}\alpha(k+1,i)\Phi(\mu_{k+1,i},\Gamma_{k+1,i})$$

where $\alpha(k+1,i)$, $\mu_{k+1,i},\Gamma_{k+1,i}$ are obtained by combining and/or neglecting some Gaussian terms in (4). There are many different methods for Step 4, such as (Ackerson and Fu, 1970; Jaffer et al. 1971; Yaakov 1978; Yaakov and Marcus, 1980; and Kim, 1994). In this paper, one method given in (Yaakov and Marcus, 1980) is adopted, in which we'll neglect the Gaussian terms with small weights when the number of the Gaussian terms exceeds a prior decided constant integer.

2.3 Brief introduction of Unknown input Kalman Filter

The standard Kalman filter assumes that all the system parameters and inputs are known, and it may fail in the presence of parametric uncertainties or unknown inputs. So (Darouach et al. 1995) proposed a method to design a Kalman filter robust to system's unknown input. Suppose that the system with unknown input is

$$\begin{cases} x(k+1)=Ax(k)+Bu(k)+Fd(k)+w(k) \\ y(k)=Cx(k)+v(k) \end{cases} \quad (5)$$

where $d(k)$ is the unknown input vector, and the meanings of the other variables and matrixes are the same as standard Kalman filter.

The algorithm for state estimation in (Darouach et al. 1995) can be summarized as:

$$
\begin{aligned}
P^{x}_{k+1/k+1} &= \left(\overline{P}^{x\ -1}_{k/k} + C^{T}V^{-1}C - \overline{P}^{x\ -1}_{k/k}F\left(F^{T}\overline{P}^{x\ -1}_{k/k}F\right)^{-1}F^{T}\overline{P}^{x\ -1}_{k/k}\right)^{-1}\\
P^{xd}_{k+1/k+1} &= P^{x}_{k+1/k+1}\overline{P}^{x\ -1}_{k/k}F\left(F^{T}\overline{P}^{x\ -1}_{k/k}F\right)^{-1}\\
P^{d}_{k/k+1} &= \left(F^{T}C^{T}\left(V + C\overline{P}^{x}_{k/k}C^{T}\right)^{-1}HF\right)^{-1}\\
\hat{x}_{k+1/k+1} &= \overline{x}_{k/k} + P^{x}_{k+1/k+1}C^{T}V^{-1}\left(y(k+1) - C\overline{x}_{k/k}\right)\\
\hat{d}_{k/k+1} &= \left(P^{xd\ T}_{k+1/k+1} - P^{d}_{k/k+1}F^{T}\right)\overline{P}^{x\ -1}_{k/k}\overline{x}_{k/k} + P^{xd\ T}_{k+1/k+1}C^{T}V^{-1}y(k+1)\\
\overline{P}^{x}_{k/k} &= AP^{x}_{k/k}A^{T} + W\\
\overline{x}_{k/k} &= A\hat{x}_{k/k} + Bu(k)
\end{aligned}
\qquad (6)
$$

3. FDI for NCSs

3.1 basic idea

As shown in Sec. 2.1, an NCS can be modeled as a JMLS with unknown input and fault, i.e. (2). Since state estimation of JMLS based on Kalman filter can not deal with unknown input, we will use UIKF technique to design filters for (2), which are robust to unknown input and a subset of faults, to achieve the goal of fault detection and fault isolation. And the Log-likelihood Ratio (LLR) method (Li and Kadirkamanathan 2004) is finally adopted to complete the FDI system.

3.2 Bayesian filter for NCS based on UIKF

The filter for JMLS introduced in Sec. 2.2 is designed in the framework of Bayes' rule. It uses Kalman filter to obtain the conditional density $p(x(k+1)|\, y_{1:k+1}, i_k, r_{k+1})$ from $p(x(k)|\, y_{1:k}, i_k)$, because in the Gaussian linear case Kalman filter is equivalent to Bayesian a posteriori estimation (Ho and Lee 1964).

As a result, when we try to use UIKF to perform state estimation of JMLS with unknown input (UIJMLS), we have to answer the question whether the UIKF is also equivalent to Bayesian a posteriori estimation. As shown in (Darouach et al. 1995), the estimation result (6), i.e. $\hat{x}_{k+1/k+1}$, is a minimum mean-square error (MMSE) estimation of $x(k+1)$ in system (5) given y up to k+1, which means that $\hat{x}_{k+1/k+1}$ and $P^{x}_{k+1/k+1}$ are as same as the mean and covariance of $x(k+1)$ respectively, given y up to k+1. Since in the Gaussian linear case, the mean and covariance can uniquely determine the density, the above fact means that the result of UIKF is equivalent to Bayesian a posteriori estimation in the Gaussian linear case.

Similar to the procedure for state estimation of JMLS based on KF in Sec. 2.2, the main target now becomes to recursively find the a posteriori density of the state $p(x(k), r_k \mid y_{1:k})$ for system (2) with unknown input. Again, let $\hat{x}(k \mid I_k = i, y_{1:k})$, $P(k \mid I_k = i, y_{1:k})$, $\alpha(k,i)$ and $s_k(i)$ denote the mean, covariance, weight, and Markov chain state of the ith Gaussian term ($i = 1\cdots n(k)$) at instant k respectively, by using UIKF, the estimation procedure can be summarized as follow:

Step 1: Obtain $\hat{x}(k+1 \mid I_k = i, r_{k+1} = j, y_{1:k+1})$, $P(k+1 \mid I_k = i, r_{k+1} = j, y_{1:k+1})$, for $i = 1\cdots n(k)$, $j = 1\cdots s$, at instant k+1, by using $n(k)\times s$ parallel UIKFs.

Step 2: Calculate the likelihood of part of the new measurement given the measurement up to instant k. This is the main difference between standard JMLS filter and the filter considering unknown input. Because of the existence of unknown input $d(k)$, it becomes impossible to obtain the margin density of $p(y(k+1)|\, y_{1:k})$, and only the margin density of a part of the new measurement, i.e. $Ty(k+1)$, where T is the basis of the left null space of CF, can be obtained now (Keller et al. 1996). In (Keller et al. 1996), the likelihood of $Ty(k+1)$ given the history measurement is:

$$
L(k+1) = \frac{\exp\left\{-\frac{1}{2}\tilde{y}(k+1 \mid I_k = i, r_{k+1} = j)^{T}\tilde{P}_{ty}(k+1 \mid I_k = i)\tilde{y}(k+1 \mid I_k = i, r_{k+1} = j)\right\}}{\left[(2\pi)^{rank(T)}\det P(k+1 \mid I_k = i, r_{k+1} = j, y_{1:k+1})\right]^{1/2}} \qquad (7)
$$

where

$$
\tilde{y}(k+1 \mid I_k = i, r_{k+1} = j) = T\left(y(k+1) - Cx(k \mid I_k = i)\right);
$$
$$
\tilde{P}(k+1 \mid I_k = i, r_{k+1} = j, y_{1:k+1}) = T\left(C\left(AP(k \mid I_k = i)A^{T} + W\right)C^{T} + V\right)T^{T}
$$

Step 3: Calculate the weights $\alpha'(k+1,i,j)$ for $i = 1\cdots n(k)$, $j = 1\cdots s$. Since there is

$$
p(x(k+1), r_{k+1} \mid y_{1:k+1}) = \sum_{j=1,i=1}^{s,n(k)}\left(p(x(k+1)|\, r_{k+1} = j, I_k = i, y_{1:k+1})\Pr(r_{k+1} = j, I_k = i \mid y_{1:k+1})\right)
$$

we can conclude that $\alpha'(k+1,i,j)$ is equal to $\Pr(r_{k+1} = j, I_k = i \mid y_{1:k+1})$, which can be further written as

$$
\begin{aligned}
\alpha'(k+1,i,j) &= \Pr(r_{k+1} = j, I_k = i \mid y_{1:k+1})\\
&= \frac{p(Ty_{k+1} \mid r_{k+1} = j, I_k = i, y_{1:k})\Pr(r_{k+1} = j \mid I_k = i)\alpha(k,i)}{c(k)}
\end{aligned}
\qquad (8)
$$

where

$$
c(k) = \sum_{i,j=1}^{n(k),s} p(Ty_{k+1}, r_{k+1} = j, I_k = i \mid y_{1:k}) \qquad (9)
$$

In (8), the numerator's first part have been obtained in step 2. Its second part is equal to $\Pr(r_{k+1} = j \mid r_k = s_k(i))$, which is the prior knowledge of the Markov chain, and its third part is the weight obtained at instant k. As a result, we can get all the $n(k)\times s$ weights for the Gaussian terms at instant k+1. And the denominator $c(k)$ is the sum of the numerator for $i = 1\cdots n(k)$ and $j = 1\cdots s$.

Step 4: Perform the same neglecting and/or combining operation as that in Step 4 of Sec. 2.2.

After these four steps, $\hat{x}(k+1 \mid I_{k+1} = i, y_{1:k+1})$, $P(k+1 \mid I_{k+1} = i, y_{1:k+1})$, $\alpha(k,i)$, and $s_{k+1}(i)$ for $i = 1\cdots n(k)$, which composed the approximate a posteriori density $p(x(k+1), r_{k+1} \mid y_{1:k+1})$, are obtained, and the procedure is closed.

Up to now, we have completed the task of design a Bayesian filter for NCS, which can approximately estimate the a posteriori density of the system's state vector.

3.3 FDI strategy

Assume that there are n_f faults, and they won't happen simultaneously. For the purpose of fault isolation, we can divide f into two subsets, i.e. $f_i \in R^1$ and $f_{\bar{i}} \in R^{n_f-1}$, which are composed of the ith fault, and the other faults respectively. Then the NCS model (2) becomes:

$$\begin{cases} x(k+1)=Ax(k)+B(r_{k+1})U(k)+Ed(k)+F_i f_i(k)+F_{\bar{i}} f_{\bar{i}}(k)+w(k) \\ y(k)=Cx(k)+v(k) \end{cases} \quad (10)$$

We may define n_f+1 parallel filters for NCS based on the method discussed in Sec. 3.2, i.e. filter i, i=0,1…n_f, by using the method in Sec. 3.2. Among them, filter i, denoted by M_i, is robust to both unknown input d and the ith fault $f_i(k)$, and filter 0, denoted by M_0, is only robust to unknown input d.

According to the The LLR approach (Li and Kadirkamanathan 2004), we may further define n_f likelihood ratio

$$S_r^k(i)=\sum_{j=r}^{k}\ln\frac{p(y(j)\mid y_{1:j-1},M_i)}{p(y(j)\mid y_{1:j-1},M_0)} \quad (11)$$

where the likelihood $p(y(j)\mid y_{1:j-1},M_i)$ is the one step ahead output prediction density based on the NCS filter M_i, i=0,1,…n_f.

According to LLR approach (Li and Kadirkamanathan 2004), in the presence of fault h, the likelihood based on filter M_h won't change obviously, but the likelihoods based on other filters will decrease. As a result, the LLR $S_r^k(h)$ will positively drift away and take the greatest value among the LLRs defined by (11).

However, as explained in Step 2 of Sec. 3.2, since there is unknown input, we can only get the conditional margin density of $Ty(k+1)$. So (11) should be modified as

$$\begin{aligned} S_r^k(i)&=\sum_{j=r}^{k}\ln\frac{p(T_i y(j)\mid y_{1:j-1},M_i)}{p(T_0 y(j)\mid y_{1:j-1},M_0)} \\ &=\sum_{j=r}^{k}\ln\frac{c_i(j)}{c_0(j)} \end{aligned} \quad (12)$$

where $c_i(j)$ is the likelihood of the filter M_i, which can be calculate according to equation (9). The decision function vector is defined as:

$$\begin{aligned} \gamma(k)&=\begin{bmatrix}\gamma_1(k) & \gamma_2(k) & \cdots & \gamma_{n_f}(k)\end{bmatrix} \\ \gamma_i(k)&=\max_{k-N+1\le j\le k} S_j^k(i) \end{aligned} \quad (13)$$

If the maximal element in the decision vector is larger than the threshold, we can judge that the fault corresponding to it occurs.

Remark: Because the dimensions of the n_f+1 NCS filters are different, the mean values of log-likelihoods $\ln(c_i(j))$ are also different. Therefore, we should subtract the mean value of $\ln(c_i(j))$ from it. Then the mean value of LLR will be near zeros in the fault free case.

4. Simulation Example

To illustrate the fault detection algorithm, an example is given in this section. Suppose that the matrixes of system (1) are:

$$A=\begin{bmatrix}0.5 & 2 & 0 & 0 & 0\\ 0 & 0.2 & 1 & 0 & 1\\ 0 & 0 & 0.3 & 0 & 1\\ 0 & 0 & 0 & 0.7 & 1\\ 0 & 0 & 0 & 0 & 0.1\end{bmatrix}, B=\begin{bmatrix}1 & 0 & 0\\ 0 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 0\\ 0 & 0 & 1\end{bmatrix}$$

$$C=I_5, M=\begin{bmatrix}1 & 0\\ 1 & 0\\ 0 & 0\\ 0 & 0\\ 0 & 0\end{bmatrix}, F=\begin{bmatrix}1 & -0.5\\ 0 & 0\\ 1 & 0\\ 0 & 0\\ 0 & 0\end{bmatrix}$$

$$\mathrm{cov}(v(k))=V=10^{-2}\times\begin{bmatrix}1 & & & 0.5 & \\ & 1 & & & 0.3\\ & & 1 & & \\ 0.5 & & & 1 & \\ & 0.3 & & & 1\end{bmatrix}, \mathrm{cov}(w(k))=W=10^{-2}\times\begin{bmatrix}1 & & & & \\ & 1 & 0.5 & & \\ & 0.5 & 1 & & \\ & & & 1 & \\ & & & & 1\end{bmatrix}$$

And the parameters of the Markov chain are:

$$\pi=\begin{bmatrix}0.35 & 0.65 & 0\\ 0.225 & 0.4179 & 0.3571\\ 0.225 & 0.4179 & 0.3571\end{bmatrix}$$
$$\pi(0)=\begin{bmatrix}0.3 & 0.4 & 0.3\end{bmatrix}$$

Figure 1 gives the unknown inputs d1 and d2 and faults f1 and f2. f1 is an abrupt fault while f2 is an incipient fault. The simulation curve of decision element $\gamma_1(k)$ (dashed curve) and decision element $\gamma_2(k)$ (solid curve) are depicted in figure 2 and figure 3. As shown in figure 1, fault 1 happens at instant 310, and fault 2 at instant 700. From Figure 2, it can be seen that $\gamma_1(k)$ and $\gamma_2(k)$ both exceed the threshold at instant 310, but since $\gamma_1(k)$ is greater than $\gamma_2(k)$, we can conclude that fault 1 occurs. Similarly, figure 3 shows that $\gamma_2(k)$ exceeds threshold and is greater than $\gamma_1(k)$ after fault 2 occurs, so we can conclude that fault 2 has occurred. Further, from Fig.2 and Fig.3, it can be seen that neither of the curves of decision element is affected by the unknown input $d(k)$. For comparison, figure 4 gives the result of the decision vector via UIKF without considering the switching, i.e. the unknown network-induced delay. We can see that the UIKF lose its validity.

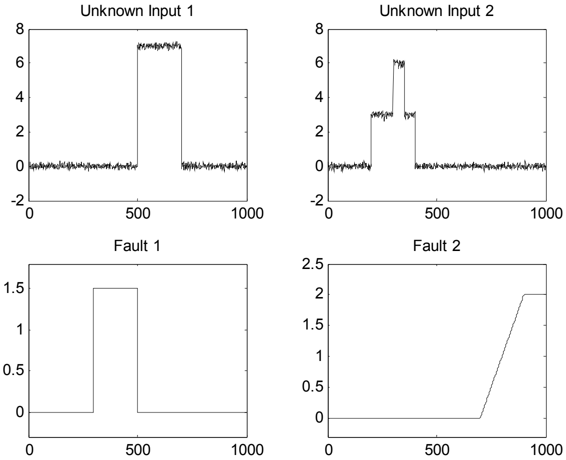

Fig. 1 Unknown input and fault

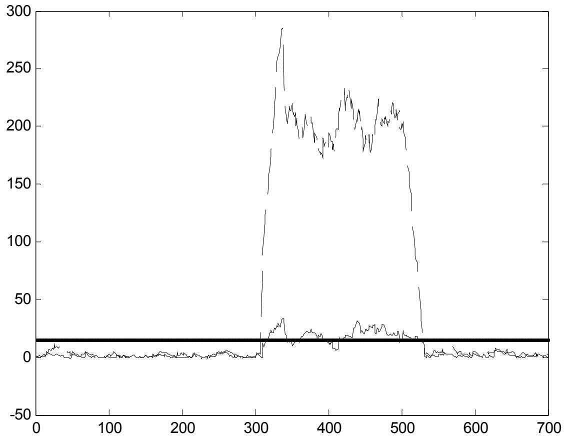

Fig. 2 Fault detection and isolation result by LLRs for the first fault

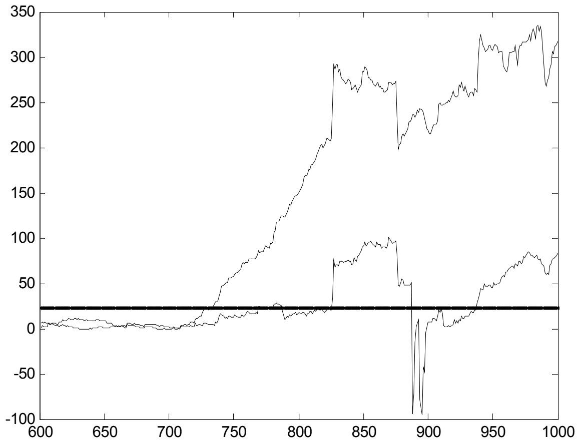

Fig. 3 Fault detection and isolation result by LLRs for the second fault

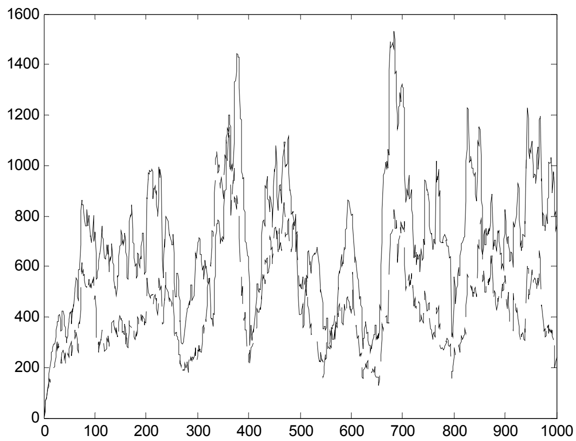

Fig. 4 Fault detection and isolation result by LLRs using UIKF instead of the NCS UIKF filter

5. Conclusion

A new method for fault detection and isolation (FDI) of Networked Control Systems (NCSs) based on Bayesian estimation of JMLS, UIKF and LLR is proposed in this paper. This method is robust to network-induced delay, and traditional unknown input. Simulation example shows the effectiveness of the method.

ACKNOWLEDGEMENT

The work is supported by the National Natural Science Fund of China under Grant 60574085.

REFERENCES

Ackerson Guy A., Fu K.S. (1970). On state estimation in switching environments. IEEE transactions on automatic control, 15, 10-17.

Darouach M., Zasadzinski M., Bassong Anana A., Nowakowski S. (1995). Kalman filtering with unknown inputs via optimal state estimation of singular systems. International journal of systems science, 26(10), 2015-2028.

Doucet A., Neil. J. Gordon, and Vikram Krishnamurthy (2001). Particle filters for state estimation of jump Markov linear systems. IEEE transactions on signal processing, 49(3), 613-624.

Ho Y.C., Lee R.C.Y. (1964). A Bayesian approach to problems in stochastic estimation and control. IEEE transactions on automatic control, 9, 333-339.

Jaffer A. G., Gupta S. C. (1971). Optimal sequential estimation of discrete processes with Markov interrupted observations. IEEE transactions on automatic control, 16, 471-475.

Keller J.Y., Summerer L., Boutayeb M., Darouach M. (1996). Generalized likelihood ratio approach for fault detection in linear dynamic stochastic systems with unknown inputs. International journal of systems science, 27(12), 1231-1241.

Kim Chang-Jin (1994). Dynamic linear model with Markov-switching. Journal of Econometrics, 60, 1-22.

Li Ping, Visakan Kadirkamanathan (2004). Fault detection and isolation in non-linear stochastic systems- A combined adaptive Monte Carlo filtering and likelihood ratio approach. International Journal of Control, 77(12), 1101-1114.

Lian. F. L., J. Moyne and D. Tilbury (2003). Modeling and optimal controller design of networked control systems with multiple delays, International Journal of Control, 76(6), 591-606.

Lin. Xiao, A. Hassibi, and J. P. How (2000). Control with random communication delays via a discrete-time jump linear system approach. Proceedings 2000 American Control Conference,3, 2199–2204.

Nilsson J. (1998). Real-time control systems with delays, PhD thesis, Department of Automatic Control, Lund Institute of Technology, Sweden.

Yaakov Bar-Shalom (1978). Tracking methods in a multitarget environment. IEEE transaction on automatic control, 23(4), 618-626.

Yaakov Bar-Shalom, Marcus G.D. (1980). Tracking with measurements of uncertain origin and random arrival times. IEEE transactions on automatic control, 25, 802-807.

Ye H. and S. X. Ding (2004), Fault detection of networked control systems with network-induced delay. Proceedings of 8th International Conference on Control, Automation Robotics and Vision, Kunming, China, 294-297.

Zhang. L., Shi. Y., and Huang. B. (2005). A New Method for Stabilization of Networked Control Systems With Random Delays. IEEE Transactions on automatic control, 50(8), 1177-1181.

Zhang, W., M.S. Branicky and S.M. Phillips (2001). Stability of networked control systems. IEEE Control Systems Magazine, 21(1), 84-99.

Zheng. Y., Fang. H. and Hua O. Wang (2006). Takagi-Sugeno fuzzy-model-based fault detection for networked control systems with MarKov delays. IEEE Transactions on systems, man, and cybernetics-part B: cybernetics, 36(4), 924-929.