

Model-Based Networked Control System Stability Based on Packet Drop Distributions

Lanzhi Teng, Peng Wen, and Wei Xiang
Faculty of Engineering and Surveying
University of Southern Queensland
West Street, Toowoomba, QLD 4350, Australia
{teng,wen,xiangwei}@usq.edu.au

Abstract—This paper studies the system stability of a Model-Based Networked Control System in the cases where packet losses follow a certain distributions. In this study, the unreliable nature of network links is modelled as a stochastic process. This process provides us two system structures, representing packets dropped and received respectively. This new system with two structures is asymptotically stable, if the plant model is updated with the data from plant within the maximum interval and the packet drop follows discrete distributions with finite expectations such as Uniform Distribution and Bernoulli distribution. If the packet loss follows discrete distributions with infinite expectation such as Poissonian Distribution, the stochastic system is stable when the biggest interval is limited to the maximal update time interval. These results are verified in simulations.

Keywords—Packet Drop Distribution, Model-Based Networked Control System, System Stability

I. INTRODUCTION

Networked control systems (NCSs) have attracted considerable amount of attention in the past decade. Compared with traditional feedback control systems, NCSs reduce the system wiring, make the system easy to operate, maintain and diagnose in case of malfunctioning. In spite of the great advantages that the networked control architecture brings, inserting a network between the plant and the controller introduces many problems as well. Network induced delays are unavoidable because of the scheduling schemes. Packet drops occur sometimes because of network congestions. Unlimited data rate is not possible because of finite bandwidth available. In [1-2], system stability has been studied while network time delays are considered. Vijay Gupta, Babak Hassibi and Richard M. Murray in [3], investigated the system performance with packet drops, and concluded that packet drops degrade a system's performance and possibly cause system instability. In [4], John K. Yook, Dawn M. Tilbury and Nandit R. Soparkar used state estimator techniques to reduce the communication volume in a networked control system.

Packet drop over a network exhibits stochastic behavior. The network can be described as there is some correlation between consecutive packets in term of Markov Chain. In [5-6] H_∞ filtering for a class of uncertain Markovian jump linear systems is investigated. A Markovian jumper linear filter is given in terms of linear matrix inequalities. Optimal Kalman filters are used in Markov jump linear systems as the estimator, and the linear matrix inequality in the bounded real lemma is given as both necessary and sufficient in [7-8].

In NCS, we consider the communication between the sensor and the controller or estimator is subject to unpredictable packet loss. We assume that if a packet dropped, a new observation is taken. In [9], Montestruque proposed a Model-based NCS, and provided the necessary and sufficient conditions for stability in terms of the update time and the parameters of the plant and its model, assuming that the frequency at which the network updates the state in the controller is constant.

In the authors' best knowledge, the packet drop distribution has not been fully investigated. This work studies system stability in the cases where packet drops follow different distributions. We model the unreliable nature of the network links as a stochastic process, and assume that this stochastic process is independent of the system initial condition and the plant model state is updated with the plant state at the time when packet arrives. Then, a model for the model-based NCS is built up and a new system matrix is obtained regarding the intervals between the arrived packets following random distributions. The result of our study shows that the system is stable as long as the system error is reset within the maximum update time. Our further study also shows that the distributions of the packet drops affect the system stability. This conclusion is demonstrated in examples at the end.

This paper is organized as follows. In section 2, system model is set up in the form of packet losses. In section 3, system stability is analyzed in the cases where packet drops follow different distributions. In section 4, example is provided to verify our conclusion. Conclusion is drawn in section 5.

II. SYSTEM DESCRIPTION

A model-based control system in Fig. 1 is considered, where the plant is given as:

$$x(n+1) = Ax(n) + Bu(n) \quad (1)$$

where $x(n)$ is the plant state vector. A , B are system parameter matrices.

A model of the plant is built up to provide the estimated plant state vector. The plant model dynamics is given by:

$$\hat{x}(n+1) = \hat{A}\hat{x}(n) + \hat{B}u(n) \quad (2)$$

where $\hat{x}(n)$ is the estimate of the plant state, \hat{A} , \hat{B} are the model matrices. We define the modelling error matrices

$\tilde{A} = A - \hat{A}$, and $\tilde{B} = B - \hat{B}$, representing the difference between the plant and the model.

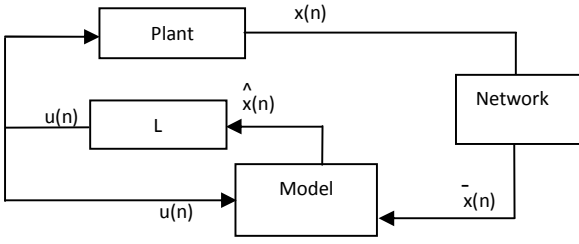


Figure 1 Model-based Networked Control System Model

We consider only state feedback controller, which is given by:

$$u(n) = L\hat{x}(n) \quad (3)$$

where L is the controller feedback gain matrix. We assume that the dynamic model is subject to the same control signal $u(n)$ as the original system.

III. STABILITY ANALYSIS

As the packets are sent through a network from the plant to the controller, the packet drops happen randomly. Periods of packet streams are interrupted by periods without packets received. The intervals between the received packets vary with a distribution. The lengths of interval depend on the QoS of the network.

The stochastic process $\{\gamma_n\}$ models the unreliable nature of the network links. We assume that γ_n is independent of the initial condition, $x(0)$. We reuse the previous packet if a packet is not received. The vector $\bar{x}(n)$ is current state $x(n)$ with probability p if a packet is received. That gives us the following equation:

$$\bar{x}(n) = \gamma_n x(n) \quad (4)$$

Although the control packets may not be received by the plant, we make the assumption that the model will always know without delay whether or not the control packet was received. We define the state error as:

$$e(n) = \bar{x}(n) - \hat{x}(n) \quad (5)$$

We assume that the plant model state $\hat{x}(n)$ be updated with the plant's state $\bar{x}(n)$ at every n_k , where $n_k - n_{k-1} = h_k$, h_k is the interval between the received packets, $k=0, 1, 2, \dots$. Then, $e(n_k) = 0$.

Now we can write the evolution of the closed loop NCS,

$$\begin{pmatrix} x(n+1) \\ e(n+1) \end{pmatrix} = A(\gamma_n) \begin{pmatrix} x(n) \\ e(n) \end{pmatrix} \quad (6)$$

$$\text{where } A(\gamma_n) = \begin{cases} A_0 = \begin{pmatrix} A & -BL \\ 0 & \hat{A} + \hat{B}L \end{pmatrix}, & \gamma_n = 0 \\ A_1 = \begin{pmatrix} A + BL & -BL \\ \tilde{A} + \tilde{B}L & \hat{A} - \tilde{B}L \end{pmatrix}, & \gamma_n = 1 \end{cases} \quad (7)$$

We define $z(n) = \begin{pmatrix} x(n) \\ e(n) \end{pmatrix}$, (6) can be represented by

$$z(n+1) = A(\gamma_n)z(n) \quad (8)$$

We modeled the system as a set of linear systems, in which the system jumps from one mode representing by A_0 to another representing by A_1 . We define matrix Λ as the function of A_0 , A_1 , p and α

$$\Lambda = f(A_0, A_1, p, \alpha) \quad (9)$$

where p is the probability of the interval between the received packets, α is the packet drop rate.

On the interval, $n \in [n_k, n_{k+1})$, the system described by (8) has the following response

$$z(n) = \begin{pmatrix} x(n) \\ e(n) \end{pmatrix} = \Lambda^{n-n_k} \begin{pmatrix} x(n_k) \\ 0 \end{pmatrix} = \Lambda^{n-n_k} z(n_k)$$

Note that at time n_k , $z(n_k) = \begin{pmatrix} x(n_k) \\ 0 \end{pmatrix}$, that is the error $e(n_k)$ is reset to zero. We can represent this by $z(n_k) = \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} z(n_k^-)$, here I is the unit matrix with proper dimensions and $z(n_k^-) = \Lambda^{h_k} z(n_{k-1})$, we have

$$z(n_k) = \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \Lambda^{h_k} z(n_{k-1}).$$

If at $n = n_0$ the initial condition $z(0) = \begin{pmatrix} x(0) \\ 0 \end{pmatrix}$, the system response is:

$$\begin{aligned} z(n) &= \Lambda^{n-n_k} \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \Lambda^{h_k} z(n_{k-1}) \\ &= \Lambda^{n-n_k} \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \Lambda^{h_k} \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \Lambda^{h_{k-1}} z(n_{k-2}) \\ &\dots \\ &= \Lambda^{n-n_k} \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \Lambda^{h_k} \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \dots \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \Lambda^{h_1} \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} z(0) \end{aligned}$$

i.e.

$$z(n) = \Lambda^{n-n_k} \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \Lambda^{h_k} \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \dots \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \Lambda^{h_1} \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} z(0) \quad (10)$$

The update time h_k vary randomly. The frequency at which the network updates the state is not constant. The packet loss may obey different distributed statistical processes. The distribution of intervals, representing by h_k , may follow Bernoulli Distribution process, Uniform Distribution, Poissonian Distribution, etc. We categorize the case update time $h_k > h_0$ as time delay, and $h_k > h_{\max}$ as packet loss, where h_0 is data interval without packet loss, h_{\max} is the maximal update time. $h_k > h_{\max}$ will cause the system instable.

A. Bernoulli Distribution

Basically, random variable, $\gamma_n = 0$, when this link fails, i.e. the packet is lost, $\gamma_n = 1$, otherwise. γ_n takes value 0 with small probability α , representing packets dropped to yield a interval between the received packets, and γ_n takes value 1 with big probability $1-\alpha$, representing no packet dropout. α is a known constant. Fig. 2 shows the probability function of a Bernoulli distribution.

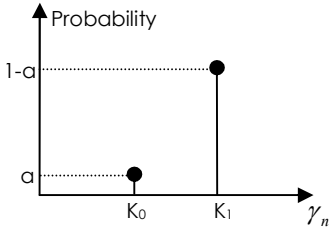


Figure 2 Probability Mass Function in Bernoulli Distribution

The probability mass function of this distribution is

$$p_i = \Pr(\gamma_n = k_i) = \begin{cases} \alpha, & k_0 = 0 \\ 1-\alpha, & k_1 = 1 \end{cases} \quad (11)$$

Fig. 3 shows that a stream of packets is interrupted with a interval of certain length without packets.

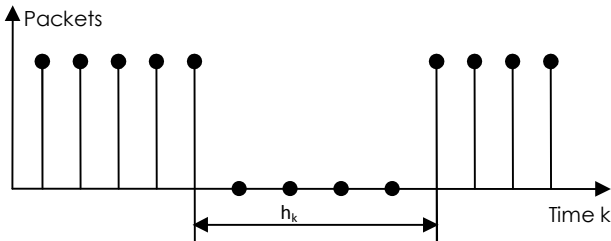


Figure 3 A Packet Stream in Bernoulli Distribution

We define $\Lambda = \alpha A_0 + (1-\alpha)A_1$ to model the jump system in case the packet loss obeys Bernoulli Distribution process. In (10), if the interval between the received packets h_k is less than the maximum update time, h_{\max} , there are packets arrived from the plant to update the model states before the system become instable. Then, the system keeps stable. When $h_k > h_{\max}$, packet loss, the system becomes instable.

B. Uniform Distribution

The adjacent packets may be received in a period, following consecutive packets drop out as shown in Fig. 4.

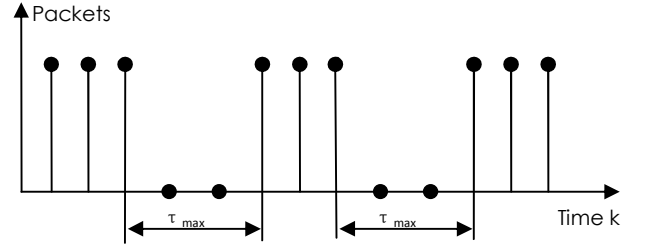


Figure 4 A Packet Stream in Uniform Distribution

The length of intervals between the received packets may vary. We assume that the interval variable X has any of n possible values, k_1, k_2, \dots, k_n that are equally possible. The probability of any outcome k_i , is $1/n$, where $i=1, 2, \dots, n$. The probability function is defined only at integer values of i as following.

$$p_i = \Pr(X = k_i) = \frac{1}{n} \quad (12)$$

Fig. 5 is the probability mass function in Uniform Distribution. The connecting lines are only guides for the eye and do not indicate continuity.

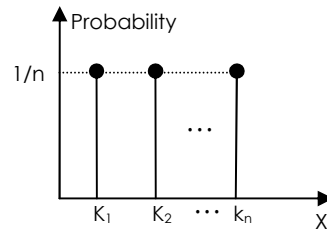


Figure 5 Probability Mass Function in Uniform Distribution

We define

$$\Lambda = \alpha \left(\frac{1}{n} A_0 + \frac{1}{n} A_0 + \dots + \frac{1}{n} A_0 \right) + (1-\alpha)A_1$$

to model the jump system in case the packet loss obeys Uniform Distribution process. We have $\Lambda = \alpha A_0 + (1-\alpha)A_1$, where α is the data dropout rate. In

(10), if the interval between the received packets is less than the maximum update time, h_{\max} , there are packets arrived from the plant to update the model states. The system is stable. To keep the system stable, $\tau_{\max} < h_{\max}$, where τ_{\max} is the biggest interval.

C. Poissonian Distribution

If packet loss obeys a Poisson distributed statistical process, i.e. if successive packet loss is independent and obeys normal counting statistics, in a sequence of n packets the interval variable X has any of n possible values, k_1, k_2, \dots, k_n probabilities p_i , where $i=1, 2, \dots, n$. The probability function is defined only at integer values of i as following

$$p_i = \Pr(X = k_i) = \frac{\lambda^{k_i} e^{-\lambda}}{k_i!} \quad (13)$$

where e is the base of the natural logarithm ($e = 2.71828\dots$), $k_i!$ is the factorial of k_i , λ is the average value of X .

Fig. 6 is the probability mass function in Poissonian Distribution.

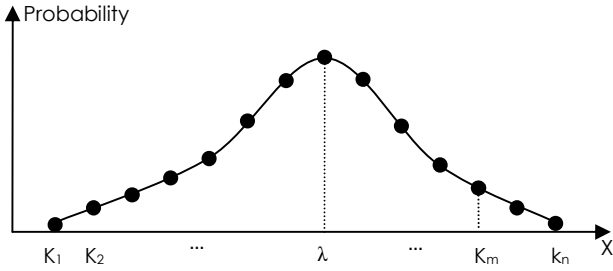


Figure 6 Probability Mass Function in Poisson Distribution

$$\text{We define } \Lambda = \alpha \left(\sum_{i=1}^{k_m} p_i A_0 + \sum_{i=k_m+1}^n p_i A_0 \right) + (1 - \alpha) A_1$$

to model the jump system in case the packet loss obeys Poissonian Distribution process. Then, we have $\Lambda = \alpha A_0 + (1 - \alpha) A_1$, where α is the data dropout rate. In (10), if the interval between the received packets is less than the maximum update time, h_{\max} , the system is stable. To keep the system stable, $\tau_{\max} < h_{\max}$, where τ_{\max} is the biggest interval. However, the interval variable is attributed to a discrete distribution with infinite as shown in Fig. 7, packets drop out at random points in time. At some points the adjacent packets may be received without interval between the packets. At some points packets may be dropped with bigger or smaller interval between the received packets. When $n \rightarrow \infty$, the interval may be infinite with a very small probability. i.e. $\tau_{\max} \rightarrow \infty$. We may find a margin value among the intervals between the received packets, $\tau_m \approx h_{\max}$, when $X = k_m$. When the interval is less than this margin value, τ_m , the

system stays stable. When the interval is greater than the maximum update time, τ_{\max} , the system becomes instable.

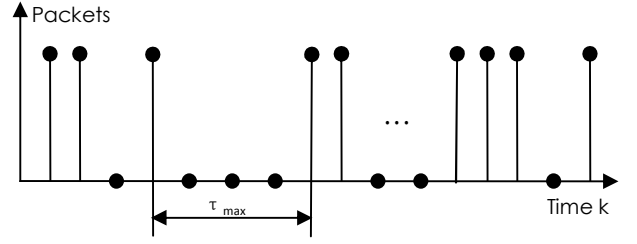


Figure 7 A Packet Stream in Poissonian Distribution

From the distributions we can see that if the random variable X , which represents the intervals between the packets, is attributed to a discrete distribution with finite expectation such as Uniform Distribution and Bernoulli Distribution, the system keeps stable when the biggest interval is less than the update time, $\tau_{\max} < h_{\max}$. If X is attributed to a discrete distribution with infinite expectation such as Poissonian Distribution, the system keeps stable for the periods $\tau < \tau_m$ and becomes instable for the periods $\tau > \tau_m$.

IV. SIMULATIONS

In order to experimentally verify the correctness of the above analysis, we used a simple feedback control system to estimate the system response in case of packet loss. We now present an example of a full state feedback as following:

$$x(n+1) = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} x(n) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u(n)$$

with the state feedback law

$$u(n) = (-1 \quad -2) \hat{x}(n)$$

We design a plant model using a random perturbation of the original plant matrices:

$$\hat{x}(n+1) = \begin{pmatrix} 1.3626 & 1.6636 \\ -0.2410 & 1.0056 \end{pmatrix} \hat{x}(n) + \begin{pmatrix} 0.4189 \\ 0.8578 \end{pmatrix} u(n)$$

We have two matrices,

$$A_0 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 2 \\ 0 & 0 & 0.9437 & 0.8258 \\ 0 & 0 & -1.0988 & -0.7100 \end{pmatrix}$$

and

$$A_1 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ -1 & -1 & 1 & 2 \\ 0.0563 & 0.1742 & 0.9437 & 0.8258 \\ 0.0988 & -0.2900 & -0.0988 & 1.2900 \end{pmatrix}$$

We assume there is no packet dropout and the frequency at which the network updates the state is constant. Fig. 8 is a plot of magnitude of the maximum eigenvalues of $\begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \Lambda^h \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix}$ against update time h . From the graph it can be seen that the maximum value for h is $h_{\max} = 4$. For $h > 4$, the NCS has eigenvalues with magnitude larger than one and therefore will be unstable.

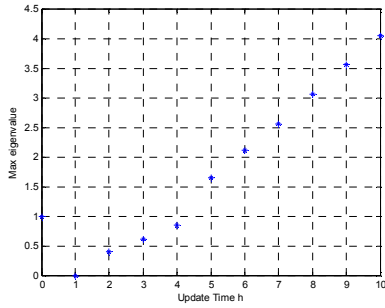


Figure 8 The Plot of Magnitude of the Maximum eigenvalues of the Test Matrix

In the simulation, the system jumps from one mode with $h > 4$, representing by A_0 to another mode with $h \leq 4$, representing by A_1 . Table 1 shows the algorithm.

Table 1 The Algorithm to Verify the Analysis

Given plant matrices, model matrices, and corresponding discrete distribution
1. Form the matrices A_0 and A_1 according to the control scheme
2. Produce the random numbers, simulating the packet intervals, according to the different discrete distribution
3. Apply different matrices to calculate the system response according to different packet intervals
if $h > 4$
$\begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} A_0^h \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix}$
else
$\begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} A_1^h \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix}$
end
4. Plot the system response

Fig. 9-11 show the plots of the system responses with initial condition $z(0) = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}$ in case the packet intervals are attributed to Bernoulli Distribution, Uniform Distribution, and Poissonian Distribution, respectively.

The random numbers, simulating Bernoulli Distribution packet intervals, are as follows, 1, 4, 1, 1, 1, 1, 1, 1, 1, 1, 1, 4, 1, 1, 1, 4, 1, 1, 4, 1, 1, 1, 1, 1.

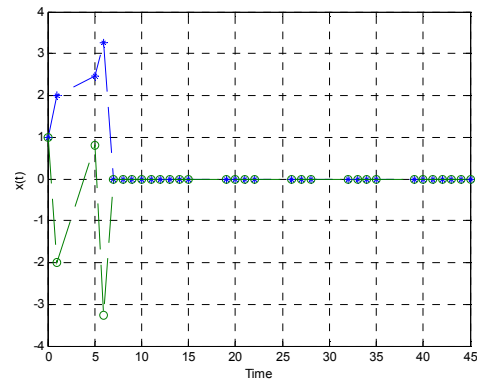


Figure 9 The System Response with Bernoulli Distribution Packet Intervals

The random numbers, simulating Uniform Distribution packet intervals, are as follows, 2, 3, 3, 2, 2, 4, 3, 3, 4, 3, 4, 1, 1, 3, 3, 2, 2, 1, 4, 1, 3, 1, 1, 3, 4, 3, 4, 3, 3, 2.

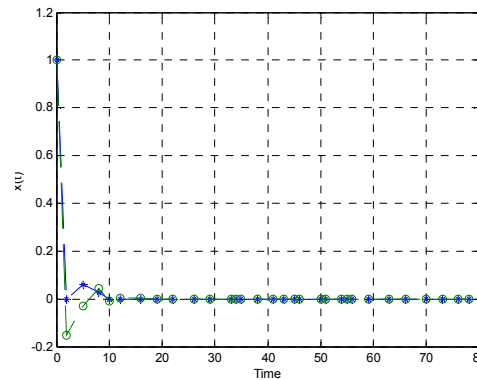
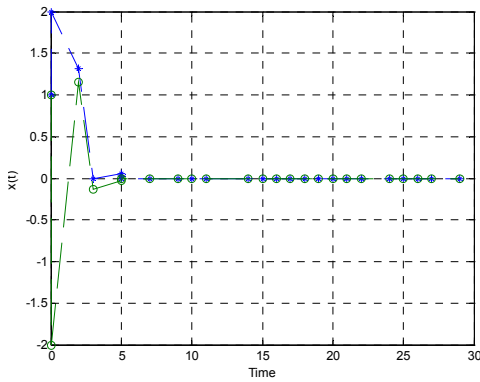
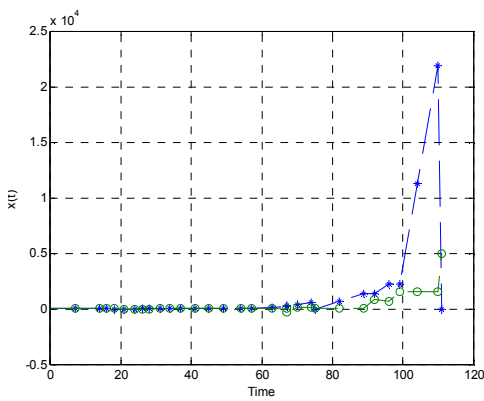


Figure 10 The System Response with Uniform Distribution Packet Intervals

The random numbers, simulating Poisson Distribution packet intervals, are as follows, (a) 1, 3, 2, 3, 1, 1, 3, 3, 2, 1, 1, 2, 1, 4, 2, 2, 2, 2, 1, 2, 1, 1, 2, 2, 2, 3, 2, 2, 2, 3; (b) 8, 8, 3, 3, 4, 4, 3, 3, 4, 4, 4, 5, 5, 5, 6, 4, 7, 5, 1, 4, 5, 2, 8, 8, 4, 5, 4, 6, 7, 2.



(a) The Biggest Packet Interval is less than 4



(b) The Biggest Packet Interval is Greater than 4

Figure 11 The System Response with Poissonian Distribution Packet Intervals

From the above simulation results we can see that if the packet intervals are attributed to a discrete distribution with finite support such as Uniform Distribution and Bernoulli Distribution, the system keeps stable when the biggest interval is less than the update time. If they are attributed to a discrete distribution with infinite support such as Poissonian Distribution, the system keeps stable in finite time. When time increases, the packet interval range increases. Then, the system becomes instable.

V. CONCLUSION

In this paper, the stability problem in NCS with unpredictable packet drops has been investigated. The result of our study shows that the system is stable as long as the system error is reset within the maximum update time. Our further study also shows that the distributions of the packet drops affect the system stability. If the packet drop follows discrete distribution with finite expectations such as Uniform Distribution and Bernoulli distribution, the system is asymptotically stable when the maximum time interval between the received packets is under the maximum update time. If the packet loss follows discrete distribution with infinite expectation such as Poisson Distribution, the stable stochastic system is asymptotically stable, when the maximum time interval between the received packets is limited to the maximal update time. This conclusion is demonstrated in examples at the end.

REFERENCES

- [1] Minrui Fei, Jun Yi, and Huosheng Hu, "Robust Stability Analysis of an Uncertain Nonlinear Networked Control System Category", *International Journal of Control, Automation, and Systems*, Vol. 4, No. 2, pp. 172-177, Apr. 2006.
- [2] Johan Nilsson, "Real-Time Control Systems with Delays", PhD Thesis, University of Toronto, Canada, 2003.
- [3] Vijay Gupta, Babak Hassibi and Richard M. Murray, "Optimal LQG Control across Packet dropping Links", *System & Control Letters*, June 2007, Vol 56, pp. 439-446.
- [4] John K. Yook, Dawn M. Tilbury and Nandit R. Soparkar, "Trading Computation for Bandwidth: Reducing Communication in Distributed Control Systems Using State Estimators", *IEEE Transactions on Control Systems Technology*, July 2002, Vol 10, No. 4, pp. 503-518.
- [5] Carlos E. de Souza and Marcelo D. Fragoso, "Robust H_∞ filtering for uncertain Markovian jump linear systems", *International Journal of Robust and Nonlinear Control*, Vol. 12, pp. 435 - 446, Jan 2002.
- [6] Marcelo D. Fragoso and Carlos E. de Souza, " H_∞ filtering for Markovian jump linear systems", *The 35th conference on Decision and Control*, Kobe, Japan, December 1996.
- [7] A. K. Fletcher, S. Rangan, and V. K. Goyal, "Estimation from Lossy Sensor Data: Jump Linear Modelling and Kalman Filtering", *Third International Symposium on Information Processing in Sensor Networks*, Berkeley, California, USA, 26-27 Apr. 2004, pp. 251-258.
- [8] Alyson K. Fletcher, Sundeep Rangan, Vivek K Goyal, and Kannan Ramchandran, "Robust Predictive Quantization: A New Analysis and Optimization Framework", *ISIT 2004*, Chicago, USA, June 27-July 2, 2004, pp. 429.
- [9] Luis A. Montestruque, "Model-Based Networked Control Systems", PhD Thesis, University of Notre Dame, November 2004.