Submitted, 2006 International Symposium on Mathematical Theory of Networks and Systems (MTNS)
http://www.cds.caltech.edu/~murray/papers/2005s_sem06-mtns.html

Towards Robust Control Over a Packet Dropping Network

Ling Shi, Michael Epstein and Richard M. Murray

***Abstract*— In this paper, we consider a robust network control problem. We consider linear unstable and uncertain discrete time plants with a network between the sensor and controller and the controller and plant. We investigate the effect of data drop out in the form of packet losses. Four distinct control schemes are explored and sufficient conditions to ensure almost sure stability of the closed loop system are derived for each of them in terms of minimum packet arrival rate and the maximum uncertainty.**

I. INTRODUCTION

In the past decade, networked control systems (NCS) have gained much attention from both the control community and the network and communication community. When compared with classical feedback control system, networked control systems have several advantages. For example, they can reduce the system wiring, make the system easy to operate and maintain and later diagnose in case of malfunctioning, and increase system agility [20]. Although NCS have advantages, inserting a network in between the plant and the controller introduces many problems as well. For instance, zero-delayed sensing and actuation, perfect information and synchronization are no longer guaranteed in the new system architecture as only finite bandwidth is available and data packet drops and delays may occur due to network traffic conditions. These must be revisited and analyzed before networked control systems become prevalent.

Recently, many researchers have spent effort on these issues and some significant results were obtained and many are in progress. Many of the aforementioned issues are studied separately. Tatikonda [19] and Sahai [13] have presented some interesting results in the area of control under communication constraints. Specifically, Tatikonda gave a necessary and sufficient condition on the channel data rate such that a noiseless LTI system in the closed loop is asymptotically stable. He also gave rate results for stabilizing a noisy LTI system over a digital channel. Sahai proposed the notion of anytime capacity to deal with real time estimation and control for a networked control system. In our paper [17], the authors have considered various rate issues under finite bandwidth, packet drops and finite controls. An optimal bit allocation scheme is given in [16] under the networked setting. The effect of pacekt drops on state estimation was studied by Sinopoli, et. al. in [3]. It has further been investigated by many researchers including the present authors in [15] and [6].

Email:{shiling, epstein, murray}@cds.caltech.edu; Tel: (626) 395-2313, Fax: (626) 796-8914.

Work supported in part by NSF ITR grant CCR-0326554.

The authors are equally contributed to this work.

One of the hallmarks of a good control system design is that the closed loop remain stable in the presence of uncertainty [4], [5]. While the researchers in [9] studied the problem of LQG control across packet dropping networks, not many have considered the norm bounded uncertainty investigated in the present paper. In [14], we have examined the impact of a norm bounded uncertainty on the network control system and provide sufficient conditions for stability in terms of the minimum data rates, packet arrival rates for the networks and system uncertainties. The major restriction being we require the system matrices B and C be invertible. In this present paper, we remove the condition and only require that the pair (A, B) is controllable and the pair (C, A) is observable. To do that, we introduce a new matrix norm, called H norm under which if a matrix is stable, the corresponding H norm is strictly less than one. Furthermore, by properly choosing H, a matrix's H norm can converge to its spectrum radius, *i.e.* , the largest absolute value of its eigenvalues.

The paper is organized as follows. In Section II, we present the mathematical model of the system and state our assumptions. Some background mathematics is also provided. In Section III, we state the sufficient conditions for closed loop stability for four distinct control schemes. In Section IV, we provide two algorithms to find the lower bounds on the packet arrival rates and upper bound on the uncertainty. Simulations on an example problem are given in Section V. Conclusions and future work are given in the last section.

II. SYSTEM DESCRIPTION AND SOME MATH PRELIMINARIES

Throughout the paper, the following notations are adopted: if $x \in \mathbb{R}^n$, $||x||$ denotes its Euclidian norm. If $A \in \mathbb{R}^{n\times n}$, $\rho(A)$ is its spectrum radius, $||A||$ denotes its induced two norm and $||A||_H$ denotes its induced H norm induced on the quadratic form $x^T Hx$ for $H > 0$, *i.e.* ,

$$\begin{aligned} ||A||^2 &= \sup_{x\neq 0} \frac{||Ax||^2}{||x||^2} = \sup_{x\neq 0} \frac{x^T A^T Ax}{x^T x} \\ ||A||_H^2 &= \sup_{x\neq 0} \frac{||Ax||_H^2}{||x||_H^2} = \sup_{x\neq 0} \frac{x^T A^T HAx}{x^T Hx}. \end{aligned}$$

Whenever $\log$ appears,it is of base 2.

We consider the networked control systems as seen in Figure 1 with linear discrete time uncertain plants of the form:

$$x_{k+1} = (A+\Delta_k)x_k + Bu_k \quad (1)$$
$$y_k = Cx_k \quad (2)$$

where $x_k \in \mathbb{R}^n$ is the state of the plant, $u_k \in \mathbb{R}^m$ is the control input and $y_k \in \mathbb{R}^q$ is the output of the plant. It is assumed, without loss of generality, that A is unstable. The pair (A, B) is assumed to be controllable and (C, A) is assumed to be observable. The plant uncertainty is given by Δ_k which is unknown but satisfies $||\Delta_k|| < \beta$. The initial condition $x_0 \in \mathbb{R}^n$ is unknown but bounded.

As seen in Figure 1 the output of the sensor and the output of the controller are to be sent through the networks. We analyze the effect of packet drops in these networks (ignoring packet delays and reordering issues) and assume infinite bandwidth. The packet drops are indicated by λ_k and γ_k, in Network 1 and Network 2 respectively. Where λ_k and γ_k are Bernoulli independent and identically distribute (i.i.d) random variables with parameters λ and γ, *i.e.* , $E[\lambda_k] = \lambda$ and $E[\gamma_k] = \gamma$ for all k. It is also assumed λ_k and γ_k are independent of each other for all k.

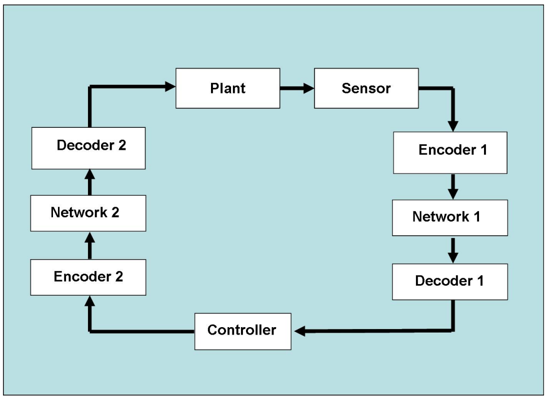

Fig. 1. Networked Control System

We assume the use of a linear observer and a state feedback controller that are designed ignoring the effect of the networks. There are several ways to implement the observer and controller in the NCS as described below. We seek to answer the question: what are the conditions on the network and system parameters $(\lambda, \gamma, \beta, A, B, C)$ such that the closed loop system is stable in some sense?

Before we give our sufficient conditions, we introduce a few lemmas on which our results rely.

Lemma 1: Assume $A \in \mathbb{R}^{n\times n}$ is stable. Then there exists an induced matrix norm $||\cdot||_H$ such that $||A||_H < 1$.

Proof: As A is stable, the following Lyapunov equation has a unique solution $H > 0$ for any $Q > 0$.

$$A^T HA - H = -Q.$$

Let $P = H^{\frac{1}{2}}$ and multiply P^{-T} from the left and P^{-1} from the right to get

$$P^{-T}A^T P^T PAP^{-1} = I - P^{-T}QP^{-1} < I,$$

hence

$$||PAP^{-1}|| = \sqrt{\lambda_{\max}(P^{-T}A^T P^T PAP^{-1})} < 1.$$

Now define $z = Px$, then

$$\begin{aligned} ||A||_H^2 &= \sup_{x\neq 0} \frac{||Ax||_H^2}{||x||_H^2} \\ &= \sup_{x\neq 0} \frac{x^T A^T HAx}{x^T Hx} \\ &= \sup_{z\neq 0} \frac{z^T P^{-T}A^T P^T PAP^{-1}z}{z^T z} \\ &= ||PAP^{-1}||^2 \\ &< 1. \end{aligned}$$

■

Corollary 2: For any matrix $A \in \mathbb{R}^{n\times n}$ and any $H > 0$, the following identity holds

$$||A||_H = ||PAP^{-1}||,$$

where $P = H^{\frac{1}{2}}$.

As packet drops introduce unavoidable randomness into the system, the classical notion of stability for deterministic systems in the sense of Lyapunov [18] is not adequate. The definition of stability in a probabilistic setting is not new. It is usually considered when there is inherent randomness in the system, for example, in the jump linear systems [8] or in stochastic hybrid systems [1]. In [8], the authors give the most frequently seen definitions of stochastic stability. We use almost sure stability in our problem formulation which is defined below.

Definition 3: System (1) is called almost sure stable if

$$P\{\lim_{k\to\infty} |x_k(x_0, \omega)| = 0\} = 1,$$

where ω is the underlying randomness for the closed loop system.

Stability in the sense of Lyapunov requires that for any $\varepsilon > 0$, there exists a time T, such that for all $k \geq T$, $|x_k| \leq \varepsilon$. For almost sure stability, however, it is allowed that $x_k > \varepsilon$ for any $k > 0$ and for any $\varepsilon > 0$ which may occur with arbitrary low probability.

Let us now consider a discrete-time jump linear system (JLS) given by

$$x_{k+1} = H(\sigma_k)x_k, k \geq 0 \quad (3)$$

where σ_k is either an i.i.d process in the state space $\underline{N} = \{1, 2, \cdots, N\}$ with probability distribution $P = \{\sigma_0 = j\} = p_j$ for $j \in \underline{N}$ or a finite-state and time homogeneous Markov chain with state space $\underline{N}$, transition probability matrix $P = (p_{ij})_{N\times N}$, and initial distribution $p = (p_1, \cdots, p_N)$. We will model the closed loop NCS as such a jump linear system. The following lemma provides a sufficient condition on the almost sure stability for such a system.

Lemma 4: [7] Suppose the form process $\{\sigma_k\}$ is a finite-state ergodic Markov process with the unique invariant distribution $\pi = \{\pi_1, \cdots, \pi_N\}$. Then the system in Eqn. (3) is almost sure stable if there exist a matrix norm $||\cdot||$ such that

$$\prod_{i=1}^{N} ||H(i)||^{\pi_i} < 1.$$

III. Main Results

We have several different options in designing the control loop in the NCS. We can choose where to place the observer: at the sensor, in which case the estimate $\hat{x}_k$ is sent across the network; or at the controller, in which case the raw sensor measurement y_k is transmitted. We consider only state feedback controllers, but we also have a choice of what to apply at the plant when the control packet is not received: apply zero control, *i.e.* the plant will evolve open-loop; or use anticipatory control which predicts what the control signal would have been and applies that value. We will analyze all four of these situations below.

1) Zero control, observer at the sensor
2) Zero control, observer at the controller
3) Anticipatory control, observer at the sensor
4) Anticipatory control, observer at the controller

Although the control packets may not be received by the plant, we make the assumption that the observer, regardless of where it is placed, will always know without delay whether or not the control packet was received.

A. Zero Control and Observer at the Sensor

The state estimator is placed at the sensor and receives the output from the plant directly. The observer is of the form

$$\hat{x}_{k+1} = A\hat{x}_k + Bu_k + L(y_k - C\hat{x}_k)\ . \tag{4}$$

The state estimate $\hat{x}_k$ is then sent across the first network, where it is either received ($\lambda_k = 1$) or dropped ($\lambda_k = 0$) by the controller.

The controller then calculates the state feedback

$$\mathbf{u}_k = \lambda_k F \hat{x}_k\ . \tag{5}$$

As seen from Eqn. (5) the state feedback term will be zero when the state estimate is not received by the controller, *i.e.* $\lambda_k = 0$. The control packet $\mathbf{u}_k$ is then sent across the second network to the plant where it is either received ($\gamma_k = 1$) or dropped ($\gamma_k = 0$). As mentioned above, the zero control implementation will evolve open-loop when the control packet is not received, hence the control applied to the plant is given by

$$u_k = \gamma_k \lambda_k F \hat{x}_k\ . \tag{6}$$

We can now write the evolution of the closed loop NCS. Define $e_k = x_k - \hat{x}_k$, then we have

$$\begin{aligned} x_{k+1} &= (A + \lambda_k\gamma_k BF)x_k - \lambda_k\gamma_k BFe_k + \Delta_k x_k \\ e_{k+1} &= (A - LC)e_k + \Delta_k x_k\ . \end{aligned}$$

Or in compact form as a Jump Linear System,

$$\begin{bmatrix} x_{k+1} \\ e_{k+1} \end{bmatrix} = A_{\theta(k)} \begin{bmatrix} x_k \\ e_k \end{bmatrix} \tag{7}$$

where $A_\theta(k)$ is a random matrix taking the following forms:

$$A_{\theta(k)} = \begin{cases} M_1 + T_k\ , & \text{if } \lambda_k\gamma_k = 1 \\ M_2 + T_k\ , & \text{if } \lambda_k\gamma_k = 0 \end{cases} \tag{8}$$

with

$$M_1 = \begin{bmatrix} A+BF & -BF \\ 0 & A-LC \end{bmatrix} \tag{9}$$

$$M_2 = \begin{bmatrix} A & 0 \\ 0 & A-LC \end{bmatrix} \tag{10}$$

$$T_k = \begin{bmatrix} \Delta_k & 0 \\ \Delta_k & 0 \end{bmatrix}. \tag{11}$$

Recall $||\Delta_k|| < \beta$, so we have

$$\begin{aligned} ||T_k|| &= \sqrt{\lambda_{\max}(T_k^T T_k)} \\ &= \sqrt{2\lambda_{\max}(\Delta_k^T \Delta_k)} \\ &= \sqrt{2}\ ||\Delta_k|| \\ &< \sqrt{2}\,\beta\ . \end{aligned}$$

Since (A, B) is assumed to be controllable, we can properly design F such that $A + BF$ is stable. Similarly, we can properly design L such that $A+LC$ is stable as (C, A) is observable. Hence M_1 is stable. By Lemma 1, we know there exists $H = PP > 0$ such that $||M_1||_H < 1$. Hence we have

$$||T_k||_H = ||PT_kP^{-1}|| \leq \sqrt{2}\,\beta||P|| \cdot ||P^{-1}||.$$

Let

$$\beta < \beta_{max} = \frac{1 - ||M_1||_H}{\sqrt{2}\ ||P|| \cdot ||P^{-1}||}, \tag{12}$$

and define the following quantities

$$N_1 = ||M_1||_H + \sqrt{2}\,\beta||P|| \cdot ||P^{-1}|| \tag{13}$$

$$N_2 = ||M_2||_H + \sqrt{2}\,\beta||P|| \cdot ||P^{-1}||\ . \tag{14}$$

Note $N_i \geq ||M_i + T_k||_H$ for all $i \in \{1, 2\}$ and all k, and we also have $N_1 < 1$.

Theorem 5: Given the zero control scheme with the observer placed at the sensor, and the quantities defined in Eqn. (9) - (14), then if λ and γ are such that

$$N_1^{\lambda\gamma} N_2^{1-\lambda\gamma} < 1, \tag{15}$$

the closed loop NCS converges almost surely.

Proof: This is a direct consequence from Lemma 4 by considering the networked control system as a jump linear Markov system, in which case the underlying Markov chain is ergodic with invariant distribution $\{\lambda\gamma, 1 - \lambda\gamma\}$. Notice that as $N_1 < 1$, inequality (15) can always be satisfied for sufficiently large λ and γ. ∎

Corollary 6: If λ and γ are such that

$$\lambda\gamma > R_{\min} = \frac{\log N_2}{\log N_2 - \log N_1}, \tag{16}$$

the closed loop system converges almost surely.

B. Zero Control and Observer at the Controller

We consider the same setup as above, but place the observer at the controller rather than at the sensor. This means the raw sensor measurement y_k is sent across the first network to the controller. The observer updates the estimate of the state according to

$$\hat{x}_k = A\hat{x}_k + Bu_k + \lambda_k L(y_k - C\hat{x}_k) \ , \tag{17}$$

which corresponds to doing the correction step only when the measurement packet is received. The control packet sent to the plant is then

$$\mathbf{u}_k = F\hat{x}_k \tag{18}$$

and the control applied to the plant

$$u_k = \gamma_k F\hat{x}_k \ . \tag{19}$$

The corresponding closed loop JLS is Eqn. (7), but with

$$A_{\theta(k)} = \begin{cases} M_1 + T_k \ , & \text{if } \lambda_k = 1 \text{ and } \gamma_k = 1 \\ M_2 + T_k \ , & \text{if } \lambda_k = 1 \text{ and } \gamma_k = 0 \\ M_3 + T_k \ , & \text{if } \lambda_k = 0 \text{ and } \gamma_k = 1 \\ M_4 + T_k \ , & \text{if } \lambda_k = 0 \text{ and } \gamma_k = 0 \end{cases} \tag{20}$$

where

$$M_1 = \begin{bmatrix} A+BF & -BF \\ 0 & A-LC \end{bmatrix} \tag{21}$$

$$M_2 = \begin{bmatrix} A & 0 \\ 0 & A-LC \end{bmatrix} \tag{22}$$

$$M_3 = \begin{bmatrix} A+BF & -BF \\ 0 & A \end{bmatrix} \tag{23}$$

$$M_4 = \begin{bmatrix} A & 0 \\ 0 & A \end{bmatrix} \tag{24}$$

$$T_k = \begin{bmatrix} \Delta_k & 0 \\ \Delta_k & 0 \end{bmatrix}. \tag{25}$$

Once again defining

$$N_i = ||M_i||_H + \sqrt{2}\,\beta ||P|| \cdot ||P^{-1}|| \ , \tag{26}$$

for $i = \{1, 2, 3, 4\}$, we have the following sufficient condition for stability.

Theorem 7: Given the zero control scheme with the observer placed at the controller, and the quantities defined in Eqn. (12) and Eqn. (21) - (26), then if λ and γ are such that

$$N_1^{\lambda\gamma} N_2^{\lambda(1-\gamma)} N_3^{(1-\lambda)\gamma} N_4^{(1-\lambda)(1-\gamma)} < 1, \tag{27}$$

the closed loop system converges almost surely.

Proof: The proof follows exactly the same as in Theorem 5. ■

C. Anticipatory Control and Observer at the Sensor

In the previous sections, when the control packet was not received at the plant the control was set to zero. As an alternative one could use a scheme similar to [12], whereby if the plant does not receive a control packet it applies the predicted control value. In this scenario, the controller transmits not only the current control value u_k, but also the predicted control values for the future, *i.e.* $\{\overline{u}_{k+1}, \overline{u}_{k+2}, \cdots\}$. We will denote this control sequence that is transmitted to the plant at time k as $\mathbf{u}_k = \{u_k, \overline{u}_{k+1}, \overline{u}_{k+2}, \cdots\}$.

Comparing this to the zero control implementation we see that the control packet will contain much more data with anticipatory control. In fact, the way it is written it will contain the predicted control input for all future time. Since we have assumed infinite bandwidth in this network this is not an issue.

The predicted control values are found by simulating the known plant dynamics. That is we use the model

$$\overline{x}_{n+1} = A\overline{x}_n + B\overline{u}_n \ , \forall n = k, k+1, \cdots \tag{28}$$

and let $\overline{u}_n = F\overline{x}_n$. Thus we can see the control sequence sent to the plant at time k is given by

$$\mathbf{u}_k = \{F\overline{x}_k, F(A+BF)\overline{x}_k, F(A+BF)^2\overline{x}_k, \cdots\} \ . \tag{29}$$

The model will need to capture the effect of packet drops in the networks. The key will be how the model state $\overline{x}_k$ is initialized. This depends on where the observer is located.

For the observer placed at the sensor, the observer will be the same as in Eqn. (4). The state estimate $\hat{x}_k$ is once again sent across the first network to the controller, where the model Eqn. (28) resides. At every timestep that the estimate is received at the controller, the value of the model state is set to the observer $\overline{x}_k = \hat{x}_k$. When the estimate is not received, the model state is initialized by setting $\overline{x}_k = A\overline{x}_{k-1} + Bu_{k-1}$, recall the assumption that knowledge of what control was previously applied allows the use of this expression.

Since the control packets $\mathbf{u}_k$ are randomly dropped, the plant applies the control value corresponding to the current timestep from the last received control sequence. In other words, $u_k = \mathbf{u}_r(k - r + 1)$ where $r = \max\{m : \gamma_m = 1, m \le k\}$, *i.e.* the last timestep the control packet was successfully received at the plant, and $\mathbf{u}_k(i)$ is the i^{th} entry in the control packet $\mathbf{u}_k$. So once a new control packet is received the previous sequences are all disregarded. Thus we see the control applied to the plant at time k can be written in terms of the state estimate according to

$$\begin{gathered} u_k = F(A+BF)^{k-p}\hat{x}_p \\ \text{where } p = \max\{m : \lambda_m\gamma_m = 1, m \le k\} \ , \end{gathered} \tag{30}$$

that is p is the last timestep both the estimator and control packets were successfully received. Note that if $p = k$, *i.e.* the current estimator and control packets are received, we recover the non-networked control signal $u_k = F\hat{x}_k$.

Now define $e_x = x_k - \hat{x}_k$ and $\overline{e}_k = x_k - \overline{x}_k$, then we see

$$\begin{aligned} x_{k+1} &= (A+BF)x_k - \lambda_k\gamma_k BF\hat{x}_k \\ &\quad -(1-\lambda_k\gamma_k)BF\overline{x}_k + \Delta_k x_k \\ e_{k+1} &= (A-LC)e_k + \Delta_k x_k \\ \overline{e}_{k+1} &= \lambda_k\gamma_k A e_k + (1-\lambda_k\gamma_k)A\overline{e}_k + \Delta_k x_k \end{aligned}$$

We can now write the closed loop system as a JLS,

$$\begin{bmatrix} x_{k+1} \\ e_{k+1} \\ \overline{e}_{k+1} \end{bmatrix} = A_{\theta(k)} \begin{bmatrix} x_k \\ e_k \\ \overline{e}_k \end{bmatrix} \tag{31}$$

where

$$A_{\theta(k)} = \begin{cases} M_1 + T_k\ , & \text{if } \lambda_k\gamma_k = 1 \\ M_2 + T_k\ , & \text{if } \lambda_k\gamma_k = 0 \end{cases} \tag{32}$$

and

$$M_1 = \begin{bmatrix} A+BF & -BF & 0 \\ 0 & A-LC & 0 \\ 0 & A & 0 \end{bmatrix} \tag{33}$$

$$M_2 = \begin{bmatrix} A+BF & 0 & -BF \\ 0 & A-LC & 0 \\ 0 & 0 & A \end{bmatrix} \tag{34}$$

$$T_k = \begin{bmatrix} \Delta_k & 0 & 0 \\ \Delta_k & 0 & 0 \\ \Delta_k & 0 & 0 \end{bmatrix} . \tag{35}$$

where now $||T_k|| < \sqrt{3}\,\beta$. Similarly to the previous analysis, M_1 will be stable (eigenvalues are those of $A+BF$, $A-LC$ and 0), so there exists an $H = PP$ such that $||M_1||_H < 1$. We still need $M_1 + T_k$ to be stable, so we will require

$$\beta < \frac{1 - ||M_1||_H}{\sqrt{3}\,||P|| \cdot ||P^{-1}||} , \tag{36}$$

and once again define the following

$$\begin{aligned} N_1 &= ||M_1||_H + \sqrt{3}\,\beta||P|| \cdot ||P^{-1}|| \quad (37) \\ N_2 &= ||M_2||_H + \sqrt{3}\,\beta||P|| \cdot ||P^{-1}|| \,. \quad (38) \end{aligned}$$

Now we state the sufficient condition for stability.

Theorem 8: Given the anticipatory control scheme described above with the observer placed at the sensor, and the quantities defined in Eqn. (33) - (38), then if λ and γ are such that

$$N_1^{\lambda\gamma} N_2^{(1-\lambda\gamma)} < 1, \tag{39}$$

the closed loop system converges almost surely.

Proof: The proof follows exactly the same as in Theorem 5. ■

D. Anticipatory Control, Observer at Controller

The control strategy is the same as described above, but now we move the observer to the controller. This means that the raw sensor measurement y_k is broadcasted across the first network and the observer takes the form as in Eqn. (17). The predicted control values are calculated using the same model as in Eqn. (28) The difference is how the model is initialized. Since the observer is now located at the controller, the model always has access to the state estimate; hence the model is initialized to $\overline{x}_k = \hat{x}_k$ at every timestep.

The control packet sent at time k is the same as Eqn. (29). Likewise the plant implements the control in the same manner, by keeping only the most recently received packet and choosing the signal corresponding to the current time. It is not too hard to see that in terms of the state estimate, the control signal applied to the plant at time k is given by

$$\begin{gathered} u_k = F(A+BF)^{k-p}\hat{x}_p \\ \text{where } p = \max\{m : \gamma_m = 1, m \le k\} \ , \end{gathered} \tag{40}$$

that is p is the last timestep the control packet was successfully received.

We can now write the closed loop system as the JLS in Eqn. (31), with

$$A_{\theta(k)} = \begin{cases} M_1 + T_k\ , & \text{if } \lambda_k = 1 \text{ and } \gamma_k = 1 \\ M_2 + T_k\ , & \text{if } \lambda_k = 1 \text{ and } \gamma_k = 0 \\ M_3 + T_k\ , & \text{if } \lambda_k = 0 \text{ and } \gamma_k = 1 \\ M_4 + T_k\ , & \text{if } \lambda_k = 0 \text{ and } \gamma_k = 0 \end{cases} \tag{41}$$

and

$$M_1 = \begin{bmatrix} A+BF & -BF & 0 \\ 0 & A-LC & 0 \\ 0 & A & 0 \end{bmatrix} \tag{42}$$

$$M_2 = \begin{bmatrix} A+BF & 0 & -BF \\ 0 & A-LC & 0 \\ 0 & 0 & A \end{bmatrix} \tag{43}$$

$$M_3 = \begin{bmatrix} A+BF & -BF & 0 \\ 0 & A & 0 \\ 0 & A & 0 \end{bmatrix} \tag{44}$$

$$M_4 = \begin{bmatrix} A+BF & 0 & -BF \\ 0 & A & 0 \\ 0 & 0 & A \end{bmatrix} \tag{45}$$

$$T_k = \begin{bmatrix} \Delta_k & 0 & 0 \\ \Delta_k & 0 & 0 \\ \Delta_k & 0 & 0 \end{bmatrix} . \tag{46}$$

Once again defining

$$N_i = ||M_i||_H + \sqrt{3}\,\beta||P|| \cdot ||P^{-1}|| \ , \tag{47}$$

for $i = \{1, 2, 3, 4\}$, we have the following sufficient condition for stability.

Theorem 9: Given the anticipatory control scheme with the observer placed at the controller, and the quantities defined in Eqn. (36) and Eqn. (42) - (47), then if λ and γ are such that

$$N_1^{\lambda\gamma} N_2^{\lambda(1-\gamma)} N_3^{(1-\lambda)\gamma} N_4^{(1-\lambda)(1-\gamma)} < 1, \qquad (48)$$

the closed loop system converges almost surely.

Proof: The proof follows exactly the same as in Theorem 5. ∎

IV. Finding $||\cdot||_H$

In Section III, we provide sufficient conditions on the network and system parameters such that the closed loop system is almost sure stable. The theorems give a condition on λ and γ for a particular H such that $||M_1||_H < 1$ and upper bound on β. The question is how to chose the H such that we can get a tight lower bound on λ and γ and a tight upper bound on β. Clearly we want to make $||M_1||_H$ as small as possible to allow for higher percentages of packet drop, but the tradeoff is that the corresponding H could make the other $||M_i||_H$ and $||P|| \cdot ||P^{-1}||$ become large, which can increase the lower bounds on λ and γ and decrease the upper bound on β. In general we want to solve the following problems.

- Find suitable H such that

$$\beta_{\max} = \frac{1 - ||M_1||_H}{||P|| \cdot ||P^{-1}||}$$

is maximum.

- Find suitable H such that the λ and γ satisfying the sufficient stability condition for the corresponding NCS implementation are minimized.

Notice that they have contradicting objectives. If we allow more uncertainty in the system, we need larger packet arrival rates. If the uncertainty is small, the arrival rates are certainly reduced.

We provide two algorithms below to find the smallest lower bounds on λ and γ and largest upper bound on β. Before we introduce them, we state a few theorems that state there indeed exists $H > 0$ such that we can push $||M_1||_H$ to its lower limit, *i.e.* , make $||M_1||_H$ as close to $\rho(M_1)$ as possible. Again the tradeoff is that the corresponding H may make $||M_2||_H$ and $||P|| \cdot ||P^{-1}||$ large.

Theorem 10: (Schur) Given $A \in \mathbb{R}^{n\times n}$ with real eigenvalues $\lambda_1, \cdots, \lambda_n$ in any prescribed order, there is an orthogonal matrix $U \in \mathbb{R}^{n\times n}$ such that

$$U^T A U = T = [t_{ij}]$$

is upper triangular, with diagonal entries $t_{ii} = \lambda_i, i = 1, \cdots, n$.

Proof: See [2], page 79. ∎

Theorem 11: Assume $A \in \mathbb{R}^{n\times n}$ is stable. Then there exists a sequence of $H_i > 0$ such that

$$\lim_{i\to\infty} ||A||_{H_i} = \rho(A)\ .$$

Proof: Let $D_i = \mathrm{diag}(i, i^2, \cdots, i^n)$. By Schur's theorem, there is an orthogonal matrix U and an upper triangular matrix T such that $A = U^T T U$. As

$$D_i T D_i^{-1} = \begin{bmatrix} \lambda_1 & i^{-1}t_{12} & i^{-2}t_{13} & \cdots & i^{-n+1}t_{1n} \\ 0 & \lambda_2 & i^{-1}t_{23} & \cdots & i^{-n+2}t_{2n} \\ 0 & 0 & \lambda_3 & \cdots & i^{-n+3}t_{3n} \\ . & . & . & \cdots & . \\ 0 & 0 & 0 & \cdots & i^{-1}t_{n-1,n} \\ 0 & 0 & 0 & 0 & \lambda_n \end{bmatrix}$$

for any given $\epsilon > 0$, for sufficiently large i, we can be sure that

$$||D_i T D_i^{-1}|| \le \rho(A) + \epsilon.$$

Define $H_i = U^T D_i^T D_i U > 0$ and $z = P_i x$ where $P_i = D_i U$. Then

$$\begin{aligned} ||A||_{H_i}^2 &= \sup_{x\neq 0} \frac{x^T A^T H A x}{x^T H x} \\ &= \sup_{z\neq 0} \frac{z^T P_i^{-T} A^T P_i^T P_i A P_i^{-1} z}{z^T z} \\ &= ||P_i A P_i^{-1}||^2 \\ &= ||D_i U U^T T U U^T D_i^{-1}|| \\ &= ||D_i T D_i^{-1}|| \\ &\le \rho(A) + \epsilon. \end{aligned}$$

On the other hand, any matrix norm is bounded below by its spectral radius, by letting i go to infinity, we hence prove the theorem. ∎

Remark 12: If $A \in \mathbb{R}^{n\times n}$ does not have real eigenvalues, then U becomes unitary instead of orthogonal. The transpose operator is replaced by the Hermitian transpose. Everything else is unchanged.

We can use the proof to Theorem 11 to form the algorithm given in Table I.

TABLE I
ALGORITHM I

Given (A, B, C, F, L) and corresponding control scheme
1. Form the matrices M_j as in Section III, where $j \in \{1,2\}$ or $\{1,2,3,4\}$ depending on different control schemes.
2. Find U, T such that $M_1 = U^T T U$ via standard Schur's algorithm. Set $i = 1$.
3. Form $H_i = P_i^T P_i$ where $P_i = D_i U$ and $D_i = \mathrm{diag}(i, i^2, \cdots, i^n)$.
4. Find maximum sufficient uncertainty according to (12) or (36) depending on different control schemes.
5. Find minimum sufficient packet arrival rates λ and γ for some portion of the uncertainty found in the last step according to Theorems 5, 7, 8 or 9.
6. $i = i + 1$
7. Repeat steps 3 to 6 until the incremental increase or decrease of these numbers are within a certain threshold.

This algorithm works fine to get a lower bound for λ and γ, however it works terrible to find a upper bound for β as zero is almost obtained in each iteration. The reason relies on the way we form the H_i matrices. In each iteration D_i is increased dramatically and hence make $||P||\cdot||P^{-1}||$ very big. Therefore we seek another algorithm to compensate for this.

The second algorithm iteratively solves the discrete time Lyapunov equation to find H. Details are given in Table II.

TABLE II

ALGORITHM II

Given (A, B, C, F, L) and corresponding control scheme
1. Form the matrices M_j as in Section III, where $j \in \{1, 2\}$ or $\{1, 2, 3, 4\}$ depending on different control schemes.
2. Set $i = 1$ and $Q_i = I$
3. Solve $A^T H_i A - H_i = -Q_i$ via standard Lyapunov equation solvers to get H_i. Set $Q_i = H_i$.
4. Decompose H_i into $H_i = P_i P_i$ via standard algorithms.
5. Find maximum sufficient uncertainty according to (12) or (36) depending on different control schemes.
6. Find minimum sufficient packet arrival rates λ and γ for some portion of the uncertainty found in the last step according to Theorems 5, 7, 8 or 9.
7. $i = i + 1$
8. Repeat steps 3 to 7 until the incremental increase or decrease of these numbers are within a certain threshold.

Notice that the difference between the two algorithms lies in the way we form H. In the first algorithm, we decompose H into $H = P^T P$, while in the second $H = PP$.

It turns out that the second algorithm almost produces the same result for lower bound of λ and γ as in the first algorithm, but gives a much better upper bound of β.

When this algorithm stops, the results we get may not be the best ones we are interested in. For example, as the algorithm starts, the lower bounds on λ and γ may begin decreasing but later increase (those bounds are equal for the case only M_1 and M_2 are present), while the upper bound on β may keep decreasing. Hence an intelligent way to make use of this algorithm is to look at the history of the results obtained and determine which values are the best. For systems having less uncertainty and more packet drops, more iterations are better. While for systems having large uncertainty and less packet drops, less iterations are enough.

V. Examples and Discussion

Consider the system described by Eqn. (1) - (2) with the following parameters,

$$A = \begin{bmatrix} 1.04779 & 0.09672 \\ 0.947898 & 0.951068 \end{bmatrix}$$
$$B = \begin{bmatrix} 0.0048768 \\ 0.096734 \end{bmatrix}$$
$$C = [1, 0]$$

so the eigenvalues of the A matrix are $[1.306, 0.6927]$. The gains F and L were chosen so that the eigenvalues of $A + BF$ and $A - LC$ were placed at $[0.025, 0.026]$ and $[0.015, 0.016]$ respectively. In all the simulations described below the maximum norm of the uncertainty was set to 30% of it's theoretical limit.

Using the second algorithm above, we can plot how the β_{max}, λ_{min}, and γ_{min} evolve under Algorithm 2. The results are shown in Fig. 2 for the anticipatory control with the observer at the controller. As seen in the plot, for roughly the first 50 iterations the minimum packet acceptance rates are decreasing but then the λ_{min} value begins to increase. The β_{max} appears to be monotonically decreasing.

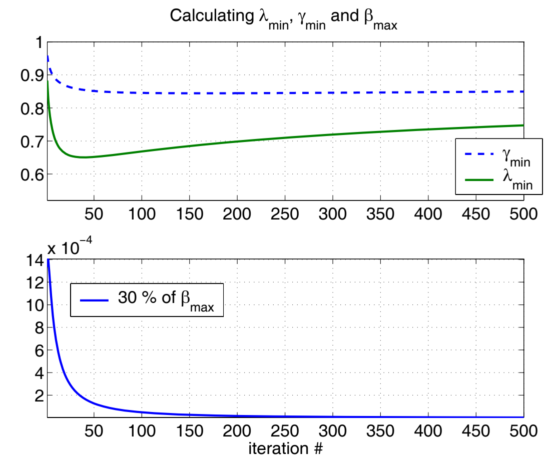

Fig. 2. Plotting the evolution of β_{max}, λ_{min}, and γ_{min} under Algorithm 2 for anticipatory control with observer at the controller.

The tradeoff in choosing H should now be apparent. For the remainder of the results, H will be chosen with Algorithm 2 terminating after 50 iterations. In Fig. 3 we see the stability regions for each of the NCS implementations. From this plot it would seem the zero control scheme would be best as it gives the larger region for stability. When running a simulation, however, we will see that in fact the anticipatory control gives more desirable behavior.

In the simulation results shown below the initial state was bounded so that $|x_i(0)| \leq 1$ and the observer state $|\hat{x}_i(0)| \leq 1$. In Fig. 4 the norm of the state is plotted for $\lambda = 0.9$ and $\gamma = 0.95$ and each of the NCS implementations. As seen they all converge to the origin as $t \to \infty$, and indeed this (λ, γ) pair is inside the stability region for all four cases. The peak of the norms are slightly smaller using the anticipatory control, indicating this might a better choice. Next, the rates were dropped to $\lambda = 0.8$ and $\gamma = 0.65$ which makes it outside the guaranteed stability region for all the control schemes. Nonetheless, as seen in Fig. 5 the states converge to the origin as well. This demonstrates that our conditions are only sufficient and not necessary. As in the previous simulation the peaks of anticipatory control scheme are significantly smaller than their zero control counterparts.

The reason the stability region for the anticipatory

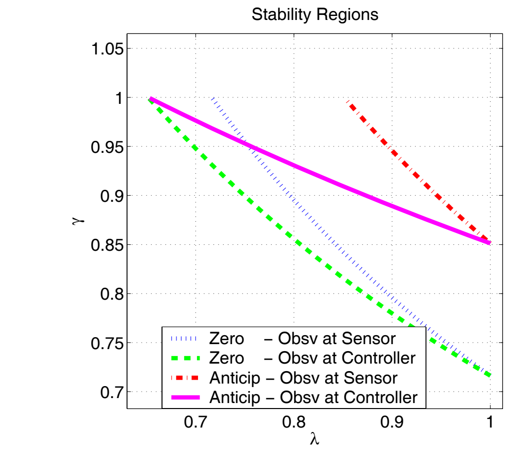

Fig. 3. Stability regions for the four different NCS implementations.

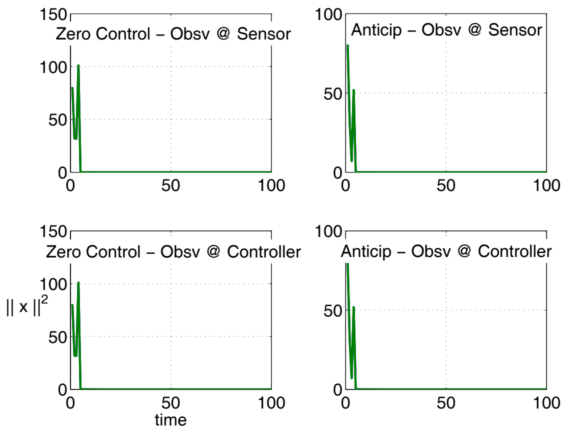

Fig. 4. Simulation results for $\lambda = 0.9$ and $\gamma = 0.95$, plots are the norm of the state.

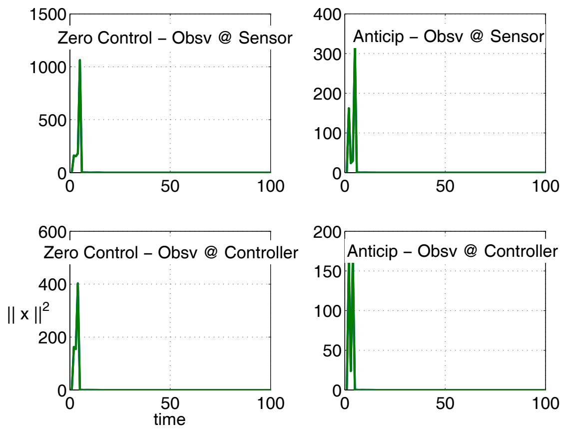

Fig. 5. Simulation results for $\lambda = 0.8$ and $\gamma = 0.65$, plots are the norm of the state.

control scheme is smaller than the zero control scheme has to do with the fact that we introduce an extra state variable in the anticipatory case. This can have the effect of increasing the associated H norms of the JLS matrices. Intuitively, the anticipatory scheme should provide better performance (provided the uncertainty is small) since the plant receives more information. This is indeed verified from the example above.

VI. Conclusion and Future Work

In this paper we analyzed controlling linear discrete time systems with norm bounded uncertainty in the plant matrix over packet dropping networks. We considered the effect of packet losses on closed loop stability. Sufficient conditions for stability are given in terms of the packet arrival probability as well as the norm of the uncertainty for different control schemes. As most networks experience not only packet drops but delays as well [10], we would like to include this effect in the stability analysis.

Although the norm bounded uncertainty is frequently used, there are other types of uncertain models that might be more applicable in certain cases. For example, when the uncertainty is described by a convex set [11]. We would like to obtain similar results for other types of uncertainty as well.

This paper has given sufficient conditions for stability and certainly some of them are not necessary. It is interesting to find necessary conditions for stability as well. Lastly, this paper has dealt with guarantees for stability but has not addressed the issue of performance. Investigating how the embedded networks degrade performance is a interesting area for future work.

References

[1] A.Abate, Ling Shi, S.Simic, and S.Sastry. A stability criterion on stochastic hybrid systems. International Symposium of Mathematical Theory of Networks and Systems, Leuven, 2004.

[2] Roger A.Horn and Charles R.Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[3] B.Sinopoli, L.Schenato, M.Franceschetti, K.Poolla, M.Jordan, and S.Sastry. Kalman filtering with intermittent observations. *IEEE Transactions on Automatic Control*, 49(9):1453–1464, 2004.

[4] John Doyle, Bruce Francis, and Allen Tannenbaum. *Feedback Control Theory*. Macmillan Publishing, 1990.

[5] Geir Dullerud and Fernando Paganini. *A Course in Robust Control Theory*. Springer, 1999.

[6] Michael Epstein, Ling Shi, Abhishek Tiwari, and Richard M. Murray. Estimation of linear stochastic systems over a queueing network. International Conference on Sensor Networks, 2005.

[7] Yuguang Fang. A new general sufficient condition for almost sure stability of jump linear systems. *IEEE Trans. Automat. Contr.*, 42:378–382, March 1997.

[8] Yuguang Fang and Kenneth A. Loparo. Stochastic stability of jump linear systems. *IEEE Transactions on Automatic Control*, 47(7):1204–1208, 2002.

[9] Vijay Gupta, Demetri Spanos, Babak Hassibi, and Richard M. Murray. On lqg control across a stochastic packet-dropping link. American Control Conference, 2005.

[10] J.Nilsson. *Real Time Control Systems with Delays*. PhD thesis, Lund Institute of Technology, Lund, Sweden, 1998.

[11] Valter Leite and Pedro Peres. Robust control through piecewise lyapunov functions for discrete time-varying uncertain systems. *International Journal of Control*, 77(3):230–238, 2004.

[12] Luis Montestruque and Panos Antsaklis. On the model based control of networked systems. *Automatica*, 39:1837–1843, 2003.

[13] A. Sahai. *Anytime Information Theory*. PhD thesis, Massachusetts Institute of Technology, 2001.

[14] Ling Shi, Michael Epstein, and Richard.M.Murray. Networked control with norm bounded uncertainty: A stability analysis. American Control Conference, Minneapolis, Minnesota, Submitted, 2006.

[15] Ling Shi, Michael Epstein, Ahbishek Tiwari, and Richard.M.Murray. Estimation with information loss: Asymptotic analysis and error bounds. CDC-ECC05, 2005.

[16] Ling Shi and Richard.M.Murray. Towards a packet-based control theory - part one: Stabilization over a packet-based network. American Control Conference, Portland, Oregon, 2005.

[17] Ling Shi and Richard.M.Murray. Towards a packet-based control theory - part two: Rate issues. American Control Conference, Minneapolis, Minnesota, Submitted, 2006.

[18] S.S.Sastry. *Nonlinear Systems; Analysis, Stability, and Control*. Springer Verlag, 1999.

[19] Sekhar Tatikonda and Sanjoy Mitter. Control under communication constraints. *IEEE Trans. Automat. Contr.*, 49:1056 – 1068, July 2004.

[20] Wei Zhang. *Stability Analysis of Networked Control Systems*. PhD thesis, Case Western Reserve University, 2001.