Analysis of Communication Losses in Vehicle Control Problems

Pete Seiler* and Raja Sengupta
Department of Mechanical Engineering
University of California at Berkeley
Email: pseiler@vehicle.me.berkeley.edu
raja@PATH.berkeley.edu

Abstract

In this paper we study the effect of communication packet losses in the feedback loop of a control system. Our motivation is derived from vehicle control problems where information is communicated via a wireless local area network. For such problems, we consider a simple packet-loss model for the communicated information and note that results for discrete-time linear systems with Markovian jumping parameters can be applied. The goal is to find a controller (if one exists) such that the closed loop is mean square stable for a given packet loss rate. A linear matrix inequality (LMI) condition is developed for the existence of a stabilizing dynamic output feedback controller. This LMI condition is used to study the effect of communication losses on a vehicle following problem. In summary, these results can be used not only to design controllers but also give a 'worst-case' performance specification (in terms of packet-loss rate) for an acceptable communications system.

1 Introduction

The purpose of this paper is to analyze the effect of random losses in the feedback loop due to a non-ideal communication network. These losses will deteriorate the performance and may even cause the system to go unstable. Our goal is two-fold. First, we would like to find a controller such that the closed loop is mean square stable for a given level of network performance. This is a problem for a control system designer who must deal with a fixed level of network performance. This goal assumes that such a controller exists. The network must satisfy some level of performance otherwise no stabilizing controller may exist. Thus the dual problem is to find limits that the network performance must satisfy for a stabilizing controller to exist.

We will examine systems in the form of Figure 1. The approach taken in this paper is motivated by problems arising in Automated Highway Systems (AHS) and coordinated control of unmanned aerial vehicles (See [9, 8] and references therein). We briefly describe some of these problems. The vehicles used in Automated Highway Systems typically use a radar to sense the relative spacing from their predecessor. During a radar failure, the fault diagnostic system may reconfigure the controller to rely on the communicated position information. In Section 5.2, we demonstrate how this simple vehicle following problem fits into the form of Figure 1 with sensor delays due to the wireless network, but no network between the controller and the plant actuator. For the coordination of unmanned aerial vehicles (UAVs), we are concerned with acceptable limits of network performance. The vision is to coordinate large numbers of UAVs to accomplish a complex task. These UAVs will communicate on an ad hoc network formed by the UAVs themselves. If we sense that the current network configuration performance is too poor for control, then we can reconfigure the network. Knowledge of the bounds on acceptable network performance is key to making this distributed agent system robust in hostile environments.

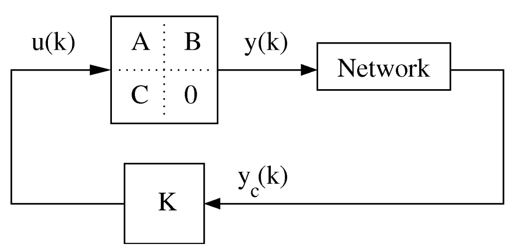

Figure 1: Feedback with communicated information

A system with communication delays in the feedback loop can be modeled as a discrete-time linear system with Markovian jumping parameters. This approach has previously been applied under the assumption of bounded delays [11, 14]. Conditions for the mean square stability of Markovian jump linear systems (MJLS) have been established in [3]. It has been shown that mean square stability is equivalent to stochastic stability and exponential mean square stability for MJLS and any one implies almost sure

*2162 Etcheverry Hall, University of California at Berkeley, Berkeley, CA 94704 stability [10]. It should also be noted that state feedback controllers for MJLS can be found by solving a set of linear matrix inequalitites (LMIs) [7, 4, 2]. However, as noted in [14], the structure of the communication delay problem results in an output feedback even if the problem was originally state feedback, i.e. the output of the combined plant/network model viewed by the controller is not the full state. The output feedback problem for MJLS is quite complex when placed in the optimization framework. This problem has been attacked as a nonconvex optimization problem in [13] for the continuous time case and in [14] for the discrete time case. Unfortunately, these routines may converge to local extrema and do not guarantee convergence to the global optimum. Another approach is to use a congruence transformation to convert the problem into an LMI optimization problem. This approach was used in [5] to find mode-dependent dynamic output feedback controllers for continuous time MJLS. The benefit of this approach is that the problem can be efficiently solved by interior-point methods [1, 6] with a guarantee that the global optimum will be found. On the other hand, the problem can only be converted into an LMI when the controller has full knowledge of the plant mode. In this paper, we develop a simple network model which is justified by the wireless networks used in vehicle control. We then use a congruence transformation motivated by [5] to find an LMI condition for a restricted case of the discrete-time dynamic output feedback problem. This LMI condition is used to find controllers and network performance constraints for the configuration in Figure 1.

2 Problem Formulation

In this paper, we will consider discrete linear time invariant systems with a communication network in the feedback loop (see Figure 1). The systems we consider are of the form:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) \end{aligned} \tag{1}$$

where $x(k) \in \mathbb{R}^n$ is the state, $u(k) \in \mathbb{R}^m$ is the control input and $y(k) \in \mathbb{R}^p$ is the measurement vector. We will assume that the output, $y(k)$, is communicated across a network and the controller has access to the communicated data, $y_c(k)$. If we assume a simple packet-loss model for the networks, then $y(k)$ and $y_c(k)$ are related by:

$$y_c(k) = \theta(k)y(k) + (1 - \theta(k))y_c(k-1) \tag{2}$$

where $\theta(k)$ is a Bernoulli process given by $Pr[\theta(k) = 0] = p$ and $Pr[\theta(k) = 1] = 1 - p$. Thus p represents the probability that any given packet will be lost. The communication model given by Equation 2 simply states that if a packet received ($\theta(k) = 1$) then it is used, but if a packet is is lost ($\theta(k) = 0$) then the most recent information should be used.

At this point, we note that this very simple network model is only applicable due to our focus on vehicle control problems. For example, the vehicles in an automated highway system communicate across a wireless LAN using a token bus architecture (See [12] for a description of some wireless technologies used in AHS). Each vehicle in a platoon has an opportunity to broadcast a packet of data once in every 20msec token cycle. The corruption of individual bits in a packet is likely to be correlated with the other bits. However, our model makes the weaker assumption that a packet loss in one token cycle is not correlated to packet receipt/loss in other token cycles; it is a Bernoulli process. Furthermore, we assume that network jitter is negligible, so we do not have to model the effect of delayed packets. Finally, the packets of data are each time stamped. Therefore, it is easy for vehicles to determine if they have lost a packet. Thus we can design a controller which uses knowledge of packet losses as they occur.

Using this simple network model, we can consider the augmented state vector, $\bar{x}(k) \in \mathbb{R}^{n+p}$:

$$\bar{x}(k) = [x^T(k) \; y_c^T(k-1)]^T \tag{3}$$

and derive a time-varying open loop plant which contains the network model:

$$\begin{aligned} \bar{x}(k+1) &= \bar{A}_{\theta(k)}\bar{x}(k) + \bar{B}u(k) \\ y_c(k) &= \bar{C}_{\theta(k)}\bar{x}(k) \end{aligned} \tag{4}$$

where:

$$\bar{A}_{\theta(k)} = \begin{bmatrix} A & 0 \\ \theta(k)C & (1-\theta(k))I \end{bmatrix} \tag{5}$$

$$\bar{B} = \begin{bmatrix} B \\ 0 \end{bmatrix} \tag{6}$$

$$\bar{C}_{\theta(k)} = \begin{bmatrix} \theta(k)C & (1-\theta(k))I \end{bmatrix} \tag{7}$$

The $\theta(k)$ subscript denotes the time varying dependence of the state matrices via the network packet loss parameters. We note that the open loop system has two modes: $\theta = 0$ when the packet from sensor is dropped and $\theta = 1$ when the packet is received. The state matrices are stationary processes where the probability of being in a given mode at time k depends on the underlying probability for $\theta(k)$. For example, the probability of being in mode 0 at any given time is equal to p. Our goal is to design a dynamic output feedback controller, K, which has access to the communicated data, $y_c(k)$. Under our network assumptions, the controller also has knowledge of $\theta(k)$, i.e. the controller knows when a packet has been dropped from the sensor. In the next section, we will present some results on discrete time Markovian Jump Linear Systems (MJLS) which can be applied to achieve this goal.

3 Markovian Jump Linear Systems

We consider the following stochastic system:

$$\begin{aligned} x(k+1) &= A_{\theta(k)}x(k) + B_{\theta(k)}u(k) \\ y(k) &= C_{\theta(k)}x(k) \\ x(0) = x_0, & \qquad \theta(0) = \theta_0 \end{aligned} \tag{8}$$

where $x(k) \in \mathbb{R}^n$ is the state, $u(k) \in \mathbb{R}^m$ is the control input and $y(k) \in \mathbb{R}^p$ is the output. The state matrices are functions of a discrete-time Markov chain taking values in a finite set $\mathcal{N} = \{1, \ldots, N\}$. The Markov chain has a transition probability matrix $\mathcal{P} = [p_{ij}]$ where $p_{ij} = Pr(\theta(k+1) = j \mid \theta(k) = i)$ subject to the restrictions $p_{ij} \geq 0$ and $\sum_{j=1}^{N} p_{ij} = 1$ for any $i \in \mathcal{N}$. These restrictions just say that the probability of jumping into a mode must be positive and that the Markov chain must jump from mode i into some state with probability one. When $\theta(k) = i$, the plant is in mode $i \in \mathcal{N}$ and we will use the following notation: $A_{\theta(k)} = A_i$, $B_{\theta(k)} = B_i$ and $C_{\theta(k)} = C_i$.

First we define several forms of stability for discrete-time jump linear systems [10].

Definition 1. *The system given by (8) with $u \equiv 0$ is:*

1. *mean-square stable (MSS) if for every initial state (x_0, θ_0), $\lim_{k\to\infty} E[\|x(k)\|^2] = 0$.*
2. *stochastically stable (SS) if for every initial state (x_0, θ_0), $E\left[\sum_{k=0}^{\infty} \|x(k)\|^2\right] < \infty$.*
3. *exponentially mean square stable (EMSS) if for every initial state (x_0, θ_0), there exists constants $0 < \alpha < 1$ and $\beta > 0$ such that $\forall k \geq 0$, $E\left[\|x(k)\|^2\right] < \beta\alpha^k \|x_0\|^2$.*
4. *almost surely stable if for every initial state (x_0, θ_0), $P\left[\lim_{k\to\infty} \|x(k)\| = 0\right] = 1$.*

It is shown in [10] that for System (8), the first three definitions of stability are actually equivalent and any one implies almost-sure stability. Furthermore, the authors of [10] refer to the equivalent notions of MSS, SS, and EMS as second-moment stability. We will subsequently refer to MSS with the meaning given by these equivalent notions of stability. Below we present matrix inequality conditions for MSS of the MJLS. Theorem 1, which is proved in [3], gives necessary and sufficient matrix inequality conditions for MSS of the system.

Theorem 1. *System (8) is MSS iff there exists matrices $G_i > 0$ for $i = 1, \ldots, N$ that satisfy any of the following conditions:*

1. $G_i - A_i^T \left(\sum_{j=1}^{N} p_{ij} G_j\right) A_i > 0$ *for* $i = 1, \ldots, N$
2. $G_j - A_j \left(\sum_{i=1}^{N} p_{ij} G_i\right) A_j^T > 0$ *for* $j = 1, \ldots, N$
3. $G_i - \sum_{j=1}^{N} p_{ij} A_j^T G_j A_j > 0$ *for* $i = 1, \ldots, N$
4. $G_j - \sum_{i=1}^{N} p_{ij} A_i G_i A_i^T > 0$ *for* $j = 1, \ldots, N$

Our specific application has the structure that $p_{ij} = p_j$ for all $i, j \in \{1, \ldots, N\}$. In words, the probability of the plant being in mode i at time $k+1$ is independent of the plant mode at time k. Thus we can actually apply the following simplified version of Theorem 1 which is presented as a Corollary in [3].

Theorem 2. *If $p_{ij} = p_j$ for all $i, j = 1, \ldots, N$ then System (8) is MSS iff there exists a matrix $G > 0$ such that $G - \sum_{j=1}^{N} p_j A_j^T G A_j > 0$.*

4 Stochastic Stabilizability

In this section we will apply Theorem 2 to derive an LMI condition for controller synthesis. We assume that the controller has access to $\theta(k)$ and the output of the augmented system, $y_c(k)$, but not the system state. The goal is to find a dynamic output feedback controller (or show that one fails to exist) of the form:

$$\begin{aligned} x_c(k+1) &= A_{c,\theta(k)}x_c(k) + B_{c,\theta(k)}y_c(k) \\ u(k) &= C_{c,\theta(k)}x_c(k) \end{aligned} \tag{9}$$

where $x_c(k) \in \mathbb{R}^n$ is the controller state and the subscript c is used to denote the controller matrices/states. Again, for $\theta(k) = i \in \{0, 1\}$, we will use A_{ci}, B_{ci}, and C_{ci} to denote the state space matrices of this two mode controller. We should note that this is not necessarily the optimal use of the measurements and knowledge of past packet loss parameters. However, this formulation leads to computationally tractable results.

With the controller structure above and the plant mode definitions given in Section 2, the closed loop matrices are given by:

$$A_{cl,i} = \begin{bmatrix} \bar{A}_i & \bar{B}_i C_{ci} \\ B_{ci}\bar{C}_i & A_{ci} \end{bmatrix} \text{ for } i \in \{0, 1\} \tag{10}$$

where the subscript 'cl' denotes the closed loop matrices. For the closed loop system, the transition probabilities are given by: $p_{cl,0} = p$ and $p_{cl,1} = (1-p)$. Apply Theorem 2 to conclude that the closed loop system is mean-square stable if and only if there exists a matrix $G > 0$ such that $G - \sum_{j=0}^{1} p_{cl,j} A_{cl,j}^T G A_{cl,j} > 0$. Let $Z = G^{-1}$. By pre- and post-multiplying this condition by Z and using Schur complements, we obtain a more useful form of this condition:

$$\begin{bmatrix} Z & (\bullet)^T & (\bullet)^T \\ \sqrt{p}A_{cl,0}Z & Z & 0 \\ \sqrt{1-p}A_{cl,1}Z & 0 & Z \end{bmatrix} > 0 \tag{11}$$

where $(\bullet)^T$ denote matrix entries which can be inferred from the symmetry of Z. Equation 11 gives a necessary and sufficient condition for the existence of a dynamic output feedback controller which gives closed loop mean-square stability. This is a bilinear matrix inequality (BMI) since it is linear in the controller parameters (for a fixed scaling matrix Z) or in Z (for fixed controller matrices). The following theorem gives an equivalent linear matrix inequality condition.

Theorem 3. *There exists $Z > 0$, A_{ci}, B_{ci}, and C_{ci} for $i = 0, 1$ such that Equation 11 holds iff there exists matrices $Y = Y^T$, $X = X^T$, L_i, F_i, and W_i for $i = 0, 1$ such that:*

$$\begin{bmatrix} \left[\begin{smallmatrix} Y & I \\ I & X \end{smallmatrix}\right] & (\bullet)^T & (\bullet)^T \\ \sqrt{p}\left[\begin{smallmatrix} Y\bar{A}_0+L_0\bar{C}_0 & W_0 \\ \bar{A}_0 & \bar{A}_0X+\bar{B}F_0 \end{smallmatrix}\right] & \left[\begin{smallmatrix} Y & I \\ I & X \end{smallmatrix}\right] & (\bullet)^T \\ \sqrt{1-p}\left[\begin{smallmatrix} Y\bar{A}_1+L_1\bar{C}_1 & \bar{W}_1 \\ \bar{A}_1 & \bar{A}_1X+\bar{B}F_1 \end{smallmatrix}\right] & \left[\begin{smallmatrix} 0 & 0 \\ 0 & 0 \end{smallmatrix}\right] & \left[\begin{smallmatrix} Y & I \\ I & X \end{smallmatrix}\right] \end{bmatrix} > 0 \tag{12}$$

Proof. ($\Rightarrow$) The proof uses a transformation motivated by the proof for the continuous time output feedback MJLS problem [5]. Assume Equation 11 holds and partition Z as:

$$Z = \begin{bmatrix} Z_1 & Z_2 \\ Z_2^T & Z_3 \end{bmatrix} \tag{13}$$

Without loss of generality, we can assume that Z_2 is non-singular. This follows since the set of full rank $n \times n$ matrices is dense in the set of all $n \times n$ matrices. Thus, without loss of generality, we assume that Z_2 is non-singular. Define the matrix $Y = (Z_1 - Z_2Z_3^{-1}Z_2^T)^{-1}$ and note that $Y > 0$ since $Z > 0$. Next, define transformation:

$$T = \begin{bmatrix} Y & I \\ -Z_3^{-1}Z_2^TY & 0 \end{bmatrix} \tag{14}$$

If we multiply Equation 11 on the left by $diag(T^T, T^T, T^T)$ and on the right by $diag(T, T, T)$, we find that Equation 12 is satisfied with the following matrix definitions (for $i = 0, 1$):

$$\begin{aligned} Y &= (Z_1 - Z_2Z_3^{-1}Z_2^T)^{-1} \\ X &= Z_1 \\ F_i &= C_{ci}Z_2^T \\ L_i &= -YZ_2Z_3^{-1}B_{ci} \\ W_i &= Y\bar{A}_iZ_1 + Y\bar{B}_iF_i + L_i\bar{C}_iZ_1 - YZ_2Z_3^{-1}A_{ci}Z_2^T \end{aligned}$$

Note that the nonsingularity of Z_2 was used to ensure that the transformation matrix was invertible.

($\Leftarrow$) Assume that we have found $Y = Y^T$, $X = X^T$, L_i, F_i, and W_i for $i = 0, 1$ such that Equation 12 holds. Then define, for $i = 0, 1$ the scaling and controller matrices as:

$$\begin{aligned} Z &= \begin{bmatrix} X & Y^{-1} - X \\ Y^{-1} - X & X - Y^{-1} \end{bmatrix} \\ B_{ci} &= Y^{-1}L_i \\ C_{ci} &= F_i\left(Y^{-1} - X\right)^{-1} \\ A_{ci} &= -Y^{-1}\left[Y\bar{A}_iX + Y\bar{B}_iF_i + L_i\bar{C}_iX - W_i\right]\left(Y^{-1} - X\right)^{-1} \end{aligned} \tag{15}$$

Schur complements can be used to show that Condition 12 implies that $X - Y^{-1}$ is positive definite and hence $Z > 0$. Next, define the transformation $T = \left[\begin{smallmatrix} Y & I \\ Y & 0 \end{smallmatrix}\right]$. If we plug the matrices given in Equation 15 into Equation 11 and multiply on the left and right by $diag(T^T, \dots, T^T)$ and $diag(T, \dots, T)$, respectively, we recover Equation 12. Hence Equation 11 holds with the matrices defined in Equation 15. □

Before proceeding, we make several comments about this theorem. If Equation 12 has a feasible solution, the proof gives a procedure for constructing a mean-square stabilizing controller. If no feasible solution exist, then it is known that no mean-square stabilizing controller of the given form exists for the particular packet loss rate, p. Fast algorithms exist to solve LMIs, making this feasibility problem computationally tractable. Also note that the proof given above can be extended to derive an LMI condition for the existence of a mode-dependent dynamic output feedback controller for any discrete time MJLS satisfying the constraint $p_{ij} = p_j$ for all i.

5 Numerical Examples

In this section, we present two examples which exploit the LMI condition given in the previous section. The first example is a simple second order system. This example is not physically motivated, but it will display the structure of the controller produced by the LMI condition. Next, we put a simple vehicle following problem in this framework and examine the effect of communication losses.

5.1 Simple 2nd Order System

The following (randomly generated) second order example will be used to study the LMI condition given in the preceding section: $A = \left[\begin{smallmatrix} -0.2656 & -2.2023 \\ -1.1878 & 0.9863 \end{smallmatrix}\right]$, $B = \left[\begin{smallmatrix} -0.5186 \\ 0.3274 \end{smallmatrix}\right]$, and $C = \left[\begin{smallmatrix} 0.2341 & 0.0215 \end{smallmatrix}\right]$. The eigenvalues of this system are -1.3739 and 2.0946. We try to estimate an upper bound on p such that stabilizing controllers exist. Using bisection, we found that stabilizing controllers exist when $p < .121$ and the LMI condition is infeasible for higher packet loss rates.

For $p = .12$, the following MSS controller was found:

$$\begin{aligned} A_{c0} &= \left[\begin{smallmatrix} -0.2153 & -0.0709 & 0.0171 \\ -1.2196 & -0.3590 & -0.0190 \\ 0.0000 & 0.0000 & 0.0012 \end{smallmatrix}\right] \quad B_{c0} = \left[\begin{smallmatrix} 0.0193 \\ -0.0203 \\ -0.9989 \end{smallmatrix}\right] \\ C_{c0} &= \left[\begin{smallmatrix} 0.0968 & 4.1097 & -0.0042 \end{smallmatrix}\right] \end{aligned}$$

$$
\begin{aligned}
A_{c1} &= \begin{bmatrix} -0.8702 & -0.1311 & -0.0011 \\ 0.3994 & -0.2105 & -0.0007 \\ 0.0001 & 0.0000 & 0.0000 \end{bmatrix} \quad B_{c1} = \begin{bmatrix} -2.8003 \\ 6.9182 \\ -0.9997 \end{bmatrix} \\
C_{c1} &= \begin{bmatrix} .0979 & 4.1095 & -.0021 \end{bmatrix}
\end{aligned}
$$

Controller 0 is used when the packet is dropped and controller 1 is used when the packet is received. We note that the plant modes are $\bar{A}_0 = \left[\begin{smallmatrix} A & 0 \\ 0 & 1 \end{smallmatrix}\right]$, $\bar{C}_0 = \left[\begin{smallmatrix} 0_{2\times 1} & 1 \end{smallmatrix}\right]$, $\bar{A}_1 = \left[\begin{smallmatrix} A & 0 \\ C & 0 \end{smallmatrix}\right]$, $\bar{C}_1 = \left[\begin{smallmatrix} C & 0 \end{smallmatrix}\right]$. The controller structure satisfies $A_{c0} \approx \bar{A}_0 + B_{c0}\bar{C}_0 - \bar{B}_0 C_{c0}$ and $A_{c1} \approx \bar{A}_1 + B_{c1}\bar{C}_1 - \bar{B}_1 C_{c1}$. In other words, the controller possesses an observer based structure with with $x_c(k)$ being an estimate of $-\bar{x}(k)$, the state of the augmented plant. Furthermore, B_{ci} are the observer gains and C_{ci} are the feedback gains. We note that $B_{c0} \approx [0\ 0\ -1]^T$ and $B_{c1} \approx [L\ -1]^T$ where $H \in \mathbb{R}^2$ is an observer gain making $A+HC$ stable. This controller structure is very intuitive. When a packet is received the estimates of the original plant states are updated with the plant A and corrected with the observer gain H. When a packet is dropped, the observer gain H is set to zero and the estimates are updated using A, which is the best we can do when no new sensor information arrives. The -1 term in both feedback gains causes the augmented network state to be estimated perfectly (the second block row of $A_i + B_{ci}C_i$ is zeroed out in both modes). This is not surprising since the controller has knowledge of $\theta(k)$ which completely determines the evolution of this state. Finally, $C_{c0} \approx C_{c1}$. Thus, once our estimate of the state is obtained, there is no advantage to varying the feedback gain based on the loss or arrival of sensor information.

5.2 Vehicle Following

Let x_1 and x_2 denote the longitudinal positions of a leader and a follower vehicle, respectively. The goal is to have vehicle 2 follow a distance δ_{des} behind vehicle 1. In other words, the controller should regulate the spacing error, $\epsilon = \delta_{des} - \delta = \delta_{des} - (x_1 - x_2)$, to zero. We assume that feedback linearization has been applied to the nonlinear vehicle model. The goal is to design a controller for the following linearized plant:

$$
\begin{aligned}
\ddot{x}_i &= a_i \qquad (16) \\
\tau \dot{a}_i + a_i &= u_i
\end{aligned}
$$

where a_i is the acceleration of vehicle i and u_i is the desired acceleration. The first order dynamics between u_i and a_i are due to the throttle/brake actuator dynamics. It is easy to show that the spacing error dynamics are given by:

$$
\tau \dddot{\epsilon} + \ddot{\epsilon} = u_2 - u_1 \qquad (17)
$$

Typically a radar is used to measure the vehicle spacing, δ. During a radar failure, the fault diagnostic system may reconfigure the controller to use position and velocity information communicated from the leading vehicle. The spacing error can then be viewed as communicated information since the leading vehicle position is required for its computation. In this scenario, we desire a controller, K, which uses the communicated spacing error to regulate the error to zero (Figure 2). The continuous time system is replaced with an equivalent discrete time system assuming a sample and hold at the plant input and an ideal sampler at the plant output. It is clear that this problem is in the form of Figure 1 with no controller to actuator delays. We would like to apply Condition 12 find an upper limit on the acceptable packet loss rate p. If a feasible controller is found, then the system is MSS when $u_1 = 0$ (lead vehicle desires a constant velocity trajectory).

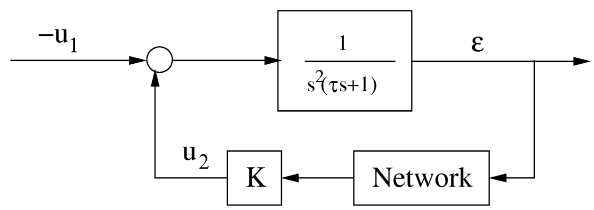

Figure 2: Vehicle Problem: Spacing Error Dynamics

A normalized version of this problem was solved with the following data: $T = \tau = 1$. Since the plant is marginally stable, it is actually possible to find a MSS controller for any $p \in [0, 1)$. For example, Figure 3 shows a simulation of the closed loop when $p = 0.9$. The upper subplot shows the spacing error starting from an initial condition of $1m$ and assuming that the lead car is traveling at a constant velocity. The solid line shows the actual spacing error and the dashed line shows the communicated spacing error used by the controller. The lowest subplot shows $\theta(k)$; the spikes are time instances where packets are received. Even at this very high packet loss rate, the controller is able to mean-square stabilize the system. The control effort (middle subplot) shows how the controller stabilizes the system. When the first packet arrives, the controller decelerates the vehicle to reduce the spacing error and then accelerates to bring the vehicle back to the speed of the preceding vehicle. Every time a packet arrives, the controller is able to tap the spacing error a bit closer to zero and then bring $\dot{e}$ close to zero. The controller is able to wait long periods of time between packets since the plant is not strictly unstable.

This example shows that MSS is a rather weak condition for the vehicle following problem. In particular, it is possible to find stabilizing controllers even for very high packet loss rates, but this does not guarantee that the performance will be adequate. In fact, the H_∞ gain from u_1 to ϵ will be quite large for high packet loss rates. This example shows that we need to extend these results to find a performance metric as a function of packet loss rate.

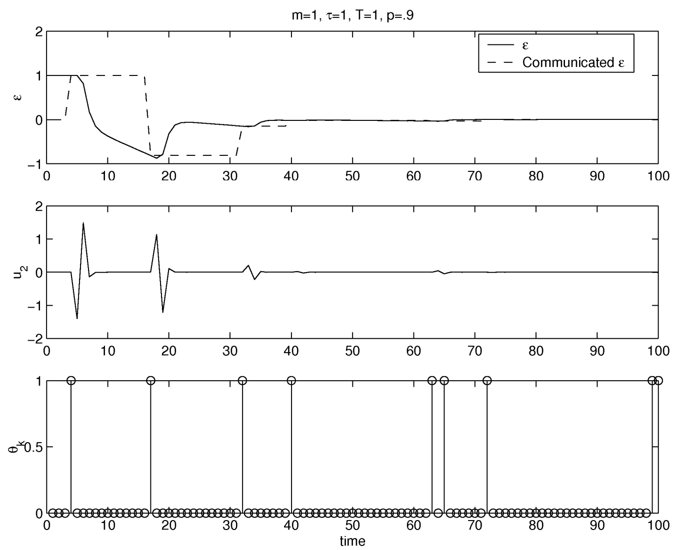

Figure 3: Vehicle Simulation: Spacing Error, Control Effort, and $\theta_1(k)$

6 Conclusions

In this paper we studied the effect of communication packet losses which occur in various vehicle control problems. By considering a simple network model, we were able to apply results for discrete-time MJLS. An LMI condition was developed for the existence of a stabilizing dynamic output feedback controller. Simple numerical examples show several areas for future research. The vehicle following example shows that these results need to be extended to measure performance versus packet loss rate. Furthermore, our vehicle following problem has the rather naive set up that each vehicle uses only information from its predecessor. Future research should focus on a large scale version of this problem where each vehicle can receive and transit information from many vehicles.

Acknowledgment

This work was supported in part by Office of Naval Research (ONR) under grant N00014-99-10756.

References

[1] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *Studies in Applied Mathematics*. SIAM, Philadelphia, PA, 1994.

[2] Yong-Yan Cao and James Lam. Stochastic stabilizability and H_∞ control for discrete-time jump linear systems with time delay. *Journal of the Franklin Institute*, 336:1263–1281, 1999.

[3] O.L.V. Costa and M. D. Fragoso. Stability results for discrete-time linear systems with markovian jumping parameters. *Journal of Mathematical Analysis and Applications*, 179:154–178, 1993.

[4] Oswaldo L.V. Costa and Ricardo P. Marques. Mixed H_2/H_∞-control of discrete-time markovian jump linear systems. *IEEE Transactions on Automatic Control*, 43(1):95–100, January 1998.

[5] D.P. de Farias, J.C. Geromel, J.B.R. do Val, and O.L.V. Costa. Output feedback control of markov jump linear systems in continous-time. *IEEE Transactions on Automatic Control*, To be published.

[6] L. El Ghaoui, R. Nikoukhah, and F. Delebecque. *LMItool: A front-end for LMI optimization, user's guide*, February 1995. Available via anonymous ftp to `ftp.ensta.fr` under `/pub/elghaoui/lmitool`.

[7] Laurent El Ghaoui and Mustapha Ait Rami. Robust state-feedback stabilization of jump linear systems via lmis. *International Journal of Robust and Nonlinear Control*, 6(9):1015–1022, 1996.

[8] Datta Godbole. Control and coordination in uninhabited combat air vehicles. In *Proceedings of the American Control Conference*, pages 1487–1490, June 1999.

[9] J.K. Hedrick, M. Tomizuka, and P.Varaiya. Control issues in automated highway systems. *IEEE Control Systems Magazine*, 14(6):21–32, December 1994.

[10] Y. Ji, H.J. Chizeck, X. Feng, and K.A. Loparo. Stability and control of discrete-time jump linear systems. *Control Theory and Advanced Technology*, 7(2):247–270, 1991.

[11] R. Krtolica, Ü. Özgüner, H. Chan, H. Göktas, J. Winkelman, and M. Liubakka. Stability of linear feedback systems with random communication delays. *International Journal of Control*, 59(4):925–953, 1994.

[12] Sonia Mahal. Effects of communication delays on string stability in an AHS environment. Master's thesis, University of California at Berkeley, March 2000.

[13] Mustapha Ait Rami, Jean-Pierre Folcher, Laurent El Ghaoui, Pascale Bendotti, and Clement-Marc Falinower. Control of jump linear systems: Application to the steam generator water level. In *Proceedings of the 38th IEEE Conference on Decision and Control*, pages 4923–4928, December 1999.

[14] Lin Xiao, Arash Hassibi, and Jonathan How. Control with random communication delays via a discrete-time jump system approach. In *Proceedings of the ACC*, pages 2199–2204, Chicago, Illinois, June 2000.