Tomas Estrada and P.J. Antsaklis, Results on Continuous and Discrete Model-Based Networked Control Systems with Intermittent Feedback, Part I: Stability, ISIS Technical Report, University of Notre Dame, ISIS-2008-001, August 2008.

Technical Report: Results on Continuous and Discrete Model-Based Networked Control Systems with Intermittent Feedback, Part I: Stability

Tomas Estrada and Panos J. Antsaklis

University of Notre Dame

August 2008

[1]

[1]The authors can be reached at testrada@nd.edu and antsaklis.1@nd.edu.

Tomas Estrada and P.J. Antsaklis, Results on Continuous and Discrete Model-Based Networked Control Systems with Intermittent Feedback, Part I: Stability, ISIS Technical Report, University of Notre Dame, ISIS-2008-001, August 2008.

Technical Report: Results on Continuous and Discrete Model-Based Networked Control Systems with Intermittent Feedback, Part I: Stability

Abstract

The aim of this technical report is to provide a thorough compendium of our results in stability of model-based networked control systems with intermittent feedback. The first set of sections deal with continuous time results, while the latter sections focus on discrete time.

We apply the concept of Intermittent Feedback to a class of networked control systems known as Model-Based Networked Control Systems (MB-NCS). Model-Based Networked Control Systems use an explicit model of the plant in order to reduce the network traffic while attempting to prevent excessive performance degradation, while Intermittent Feedback consists of the loop remaining closed for some fixed interval, then open for another interval. We begin by introducing the basic architecture for model-based control with intermittent feedback, then address the case with output feedback (through the use of a state observer), providing a full description of the state response of the system, as well as a necessary and sufficient condition for stability in each case. Examples are provided to complement the theoretical results. Extensions of our results to cases with nonlinear plants are also presented. Next, we investigate the situation where the update times τ and h are time-varying, first addressing the case where they have upper and lower bounds, then moving on to the case where their distributions are i.i.d or driven by a Markov chain.

We then shift our focus to the stability of discrete-time plants in Model-Based Networked Control Systems with Intermittent Feedback. We provide a full description of the output, as well as a necessary and sufficient condition for stability of the system. We also extend our results to the case where the full state of the plant is not known, so that we resort to a state observer. Finally, as for the continuous time case, we investigate the situation where the update times are time-varying, first addressing the case where they have upper and lower bounds, then moving on to the case where their distributions are i.i.d or driven by a Markov chain.

1 Introduction

A networked control system (NCS) is a control system in which a data network is used as feedback media. NCS is an important area in control, see for example recent surveys such as [2] and [13], as well as [24], [27], and [28]. The use of networks as media to interconnect the different components of an industrial system is rapidly increasing. However, the use of NCSs poses some challenges. One of the main problems to be addressed when considering an NCS is the size of the bandwidth required by each subsystem. A particular class of NCSs is model-based networked control systems (MB-NCS), introduced by Montestruque and Antsaklis [19]. The MB-NCS architecture makes explicit use of the knowledge of the plant dynamics to enhance the performance of the system, and it is an efficient way to address the issue of reducing packet rate. Here we extend this work by taking advantage of the novel concept of intermittent feedback. In the previous work done in MB-NCS, the updates given to the model of the plant state were performed in instantaneous fashion, but with intermittent feedback the system remains in closed loop control mode for more extended intervals. This notion makes sense as it is a good representation of what occurs in both nature and industry. For example, when driving a car, when approaching a curve or hilly terrain, we pay attention to the road for a longer time, which is equivalent to staying in closed-loop mode, and we only reduce our attention -switch to open loop control- when the road is once again straight. It is worth noting that while the application of intermittent feedback to MB-NCS, the concept has been studied in different contexts, in fields such as chemical engineering [15], psychology and behavior [25],[26], and robotics [16], [20]. While intermittent control is a very intuitive notion, its combination with the MB-NCS architecture allows for obtaining important results and opening new paths in controlling NCSs effectively.

In our earlier work [6,7,8], we have provided results for the cases where the plant is continuous-time. These results are provided with full proofs in this technical report as well.

While the continuous time results serve well as an initial approach, networked control systems require us to investigate what happens in the case of discrete-time plants as well [9]. The results presented in the latter sections are a natural extension of the corresponding ones in continuous time but have the advantage of more closely capturing what takes place in practice, since in digital communications, packets of information are transmitted at discrete intervals. It is important to note that, in the discrete time case, the parameters τ and h, which correspond to how often the loop is closed and for how long the loop is closed each time, are different from the sampling time of the digital plant, since they are tailored after the demands of use of the network, not by the internal clock of the plant. Note also that even when the loop is closed, information is being sent at discrete intervals, typically at a higher rate determined by the internal clock of the plant.

The rest of the chapter is organized as follows: in Section 2, we introduce our approach for model-based control with intermittent feedback and study it in detail. We focus first on the continuous time cases. We provide a full description of the output, as well as a necessary and sufficient condition for stability of the system. We also provide examples in order to illustrate the behavior indicated by the theory. This section deals with the case where the intervals at which the loop is closed and the intervals for which the loop remains closed are both fixed. In Section 3 we extend our results to the case where full information is of the state is not available, and thus we most resort to output feedback, using a state observer. Once again, we provide a

Tomas Estrada and P.J. Antsaklis, Results on Continuous and Discrete Model-Based Networked Control Systems with Intermittent Feedback, Part I: Stability, ISIS Technical Report, University of Notre Dame, ISIS-2008-001, August 2008.

full description of the output, as well as necessary and sufficient conditions for stability. Cases where delays are present in the network are studied in Section 4. We study the case where the plant is nonlinear in Section 5. The case for time-varying updates is presented in Section 6.

We then shift our attention to the cases with discrete-time plants. In Section 7, we describe the problem formulation in detail. We derive the complete description of the output of such a system. We present a necessary and sufficient condition for the stability of the system as well. In Section 9, we extend our results for the case where full information of the state of the plant is not available, so that we resort to a state observer. We once again provide the full description of the response of the system, as well as a necessary and sufficient condition for stability. In Section 10, we investigate the situations where τ and h are time-varying. Finally, in Section 11, we provide conclusions and propose future work.

2 Basic setup for model-based control with intermittent feedback

Let us start by introducing model-based control with intermittent feedback, in its simplest setup.

Tomas Estrada and P.J. Antsaklis, Results on Continuous and Discrete Model-Based Networked Control Systems with Intermittent Feedback, Part I: Stability, ISIS Technical Report, University of Notre Dame, ISIS-2008-001, August 2008.

2.1 Problem Formulation

The basic setup for MB-NCS with intermittent feedback is essentially the same as that proposed in the literature for traditional MB-NCS; see references [16-20] for more results on MB-NCS.

Consider the control of a continuous linear plant where the state sensor is connected to a linear controller/actuator via a network. In this case, the controller uses an explicit model of the plant that approximates the plant dynamics and makes possible the stabilization of the plant even under slow network conditions.

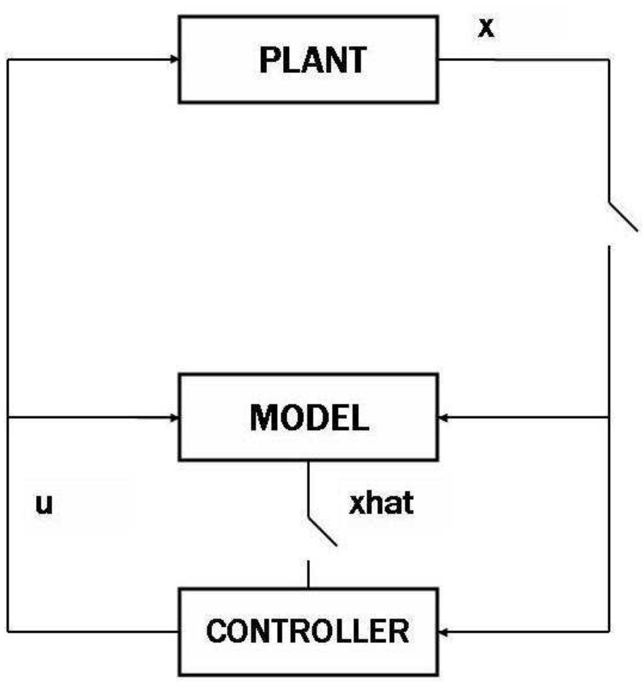

Figure 1: MB-NCS with intermittent feedback - basic architecture

The main idea here is to perform the feedback by updating the model's

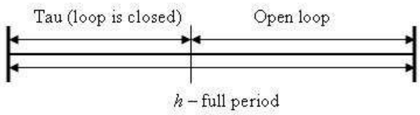

Figure 2: Partition of the time interval into close and open loop intervals

state using the actual state of the plant that is provided by the sensor. The rest of the time the control actions is based on a plant model that is incorporated in the controller/actuator and is running open loop for a period of h seconds.

As mentioned before, the main difference between model-based networked control systems as have been studied previously, and the case with intermittent feedback, which we are here discussing, is that in the literature, the loop is closed instantaneously, and the rest of the time the system is running open loop. Here, we part from the same basic idea, but the loop will remain closed for intervals of time which are different from zero. Intuitively, we should be able to achieve much better results the longer the loop is closed, as the level of degradation of the information increases the longer the system is running open loop, so intermittent feedback should yield better results than those for traditional MB-NCS.

In dealing with intermittent feedback, we have two key time parameters: how frequently we want to close the loop, which we shall denote by h, and how long we wish the loop to remain closed, which we shall denote by τ. Naturally, in the more general cases both h and τ can be time-varying. For the purposes of this section, however, we will deal only with the case where both h and τ are fixed.

We consider then a system such that the loop is closed periodically, every h seconds, and where each time the loop is closed, it remains so for a time of τ seconds. The loop is closed at times t_k, for $k = 1, 2, ...$ Thus, there are two very clear modes of operation: closed loop and open loop. The system will be operating in closed loop mode for the intervals $[t_k, t_k + \tau)$ and in open loop for the intervals $[t_k + \tau, t_{k+1})$. When the loop is closed, the control decision is based directly on the information of the state of the plant, but we will keep track of the error nonetheless.

The plant is given by $\dot{x} = Ax + Bu$, the plant model by $\dot{\hat{x}} = \hat{A}\hat{x} + \hat{B}u$, and the controller by $u = K\hat{x}$. The state error is defined as $e = x - \hat{x}$ and represents the difference between plant state and the model state. The modeling error matrices $\tilde{A} = A - \hat{A}$ and $\tilde{B} = B - \hat{B}$ represent the plant and the model. We also define the vector $z = [x^T \ e^T]^T$.

In the next subsection we will derive a complete description of the response of the system.

2.2 State Response of the System

We will now proceed to derive the response to prove the above proposition in a direct way. To this effect, let us separately investigate what happens when the system is operating under closed and open loop conditions.

During the open loop case, that is, when $t \in [t_k + \tau, t_{k+1})$, we have that

$$u = K\hat{x} \tag{1}$$

so

$$\begin{bmatrix} \dot{x} \\ \dot{\widehat{x}} \end{bmatrix} = \begin{bmatrix} A & BK \\ 0 & \hat{A} + \hat{B}K \end{bmatrix} \begin{bmatrix} x(t) \\ \hat{x}(t) \end{bmatrix} \tag{2}$$

with initial conditions $\hat{x}(t_k + \tau) = x\,(t_k + \tau)$.

Rewriting in terms of x and e, that is, of the vector z :

$$z(t) = \begin{bmatrix} \dot{x}(t) \\ \dot{e}(t) \end{bmatrix} = \begin{bmatrix} A + BK & -BK \\ \tilde{A} + \tilde{B}K & \hat{A} - \tilde{B}K \end{bmatrix} \begin{bmatrix} x(t) \\ e(t) \end{bmatrix} \tag{3}$$

$$z(t_k + \tau) = \begin{bmatrix} x(t_k + \tau) \\ e(t_k + \tau) \end{bmatrix} = \begin{bmatrix} x(t_k + \tau^-) \\ 0 \end{bmatrix}, \forall t \in [t_k + \tau, t_{k+1})$$

Thus, we have

$$\dot{z} = \Lambda_o z, \quad \text{where} \quad \Lambda_o = \begin{bmatrix} A + BK & -BK \\ \tilde{A} + \tilde{B}K & \hat{A} - \tilde{B}K \end{bmatrix}, \forall t \in [t_k + \tau, t_{k+1}) \tag{4}$$

.

The closed loop case is a simplified version of the case above, as the difference resides in the fact that the error is always zero. Thus, for $t \in$

$[t_k, t_k + \tau)$, we have

$$\dot{z} = \Lambda_c z, \quad \text{where} \quad \Lambda_c = \begin{bmatrix} A + BK & -BK \\ 0 & 0 \end{bmatrix}, \; t \in [t_k, t_k + \tau) \tag{5}$$

. This should be clear in that the error is always zero, while the state progresses in the same way as before.

From this, it should be quite clear that given an initial condition $z(t = 0) = z_0$, then after a certain time $t \in [0, \tau)$, the solution of the trajectory of the vector is given by

$$z(t) = e^{\Lambda_c(t)} z_0 \,, \; t \in [0, \tau). \tag{6}$$

In particular, at time τ, $z(\tau) = e^{\Lambda_c(\tau)} z_0$.

Once the loop is opened, the open loop behavior takes over, so that

$$z(t) = e^{\Lambda_o(t-\tau)} z(\tau) = e^{\Lambda_o(t-\tau)} e^{\Lambda_c(\tau)} z_0 \,, \; t \in [\tau, t_1). \tag{7}$$

In particular, when the time comes to close the loop again, that is, after time h, then $z\left(t_1^-\right) = e^{\Lambda_o(h-\tau)} e^{\Lambda_c(\tau)} z_0$.

Notice, however, that at this instant when we close the loop again, we are also resetting the error to zero, so that we must pre-multiply by $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ before we analyze the closed loop trajectory for the next cycle. Because we wish to always start with an error that is set to zero, we should actually multiply by $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ at the beginning.

So then, after k cycles, going through this analysis yields a solution.

$$z(t_k) = \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o(h-\tau)} e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k z_0 = \Sigma^k z_0 \,, \qquad (8)$$

where $\Sigma = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o(h-\tau)} e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$.

The final step is to consider the last (partial) cycle that the system goes through, that is, the time $t \in [t_k, t_{k+1})$. If the system is in closed loop, that is, $t \in [t_k, t_k + \tau)$, then the solution can be achieved merely by pre-multiplying $z(t_k)$ by $e^{\Lambda_c(t-t_k)}$. In the case of the system being in open loop, that is, $t \in [t_k + \tau, t_{k+1})$, then clearly we must pre-multiply by $e^{\Lambda_o(t-(t_k+\tau))} e^{\Lambda_c(\tau)}$.

The results can thus be summarized in the following proposition.

Proposition 1 *The system described by (93) and (94) with initial conditions*

$z(t_0) = \begin{bmatrix} x(t_0) \\ 0 \end{bmatrix} = z_0$ *has the following response:*

$$z(t) = \begin{cases} e^{\Lambda_c(t-t_k)} \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Sigma \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k z_0 \ , \quad t \in [t_k, t_k + \tau) \\ e^{\Lambda_o(t-(t_k+\tau))} e^{\Lambda_c(\tau)} \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Sigma \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k z_0 \ , \quad t \in [t_k + \tau, t_{k+1}) \end{cases} \tag{9}$$

where $\Sigma = e^{\Lambda_o(h-\tau)} e^{\Lambda_c(\tau)}$, $\Lambda_o = \begin{bmatrix} A + BK & -BK \\ \tilde{A} + \tilde{B}K & \hat{A} - \tilde{B}K \end{bmatrix}$, $\Lambda_c = \begin{bmatrix} A + BK & -BK \\ 0 & 0 \end{bmatrix}$,

and $t_{k+1} - t_k = h$.

In the next subsection we will present a necessary and sufficient condition for the stability of the system.

2.3 Stability condition

We now present a necessary and sufficient condition for the stability of the model-based networked control system with intermittent feedback. We use the following definition for global exponential stability. [1]

Definition 2 *The equilibrium $z = 0$ of a system described by $\dot{z} = f(t, z)$ with initial condition $z(t_0) = z_0$ is exponentially stable at large (or globally) if there exists $\alpha > 0$ and for any $\beta > 0$, there exists $k(\beta) > 0$ such that the* *solution*

$$\|\phi\left(t,t_0,z_0\right)\| \leq k\left(\beta\right)\|z_0\| e^{-\alpha(t-t_0)},\ \forall t \geq t_0 \tag{10}$$

whenever $\|z_0\| < \beta$.

With this definition of stability, we state the following theorem characterizing the necessary and sufficient conditions for the system described in the previous section to have globally exponential stability around the solution $z = 0$. The norm used here is the 2-norm, but any other consistent norm can also be used.

Theorem 3 *The system described above is globally exponentially stable around the solution* $z = \begin{bmatrix} x \\ e \end{bmatrix}$ *if and only if the eigenvalues of* $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Sigma \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ *are strictly inside the unit circle, where* $\Sigma = e^{\Lambda_o(h-\tau)}e^{\Lambda_c(\tau)}$.

Proof. Sufficiency. Taking the norm of the solution described as in Proposition #1:

$$\|z(t)\| = \left\| e^{\Lambda_c(t-t_k)} \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o(h-\tau)}e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k z_0 \right\| \leq \tag{11}$$

$$\left\| \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o(h-\tau)}e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k \right\| \|z_0\|$$

Notice we are only doing this part for the case when $t \in [t_k, t_k + \tau)$, but the process is exactly the same for the intervals where $t \in (t_k + \tau, t_k + 1)$.

Analyzing the first term on the right hand side:

$$\left\| e^{\Lambda_c (t-t_k)} \right\| \leq 1 + (t-t_k)\,\bar{\sigma}\left(\Lambda_c\right) + \frac{(t-t_k)^2}{2!} \cdots = e^{\bar{\sigma}(\Lambda_c)(t-t_k)} \leq e^{\bar{\sigma}(\Lambda_c)(\tau)} = K_1 \tag{12}$$

where $\bar{\sigma}\left(\Lambda_c\right)$ is the largest singular value of Λ_c. In general this term can always be bounded as the time difference $t - t_k$ is always smaller than τ. That is, even when Λ_c has eigenvalues with positive real part, $\left\| e^{\Lambda_c (t-t_k)} \right\|$ can only grow a certain amount. This growth is completely independent of k.

We now study the term $\left\| \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o (h-\tau)} e^{\Lambda_c (\tau)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k \right\|$. It is clear that this term will be bounded if and only if the eigenvalues of $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o (h-\tau)} e^{\Lambda_c (\tau)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ lie inside the unit circle:

$$\left\| \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o (h-\tau)} e^{\Lambda_c (\tau)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k \right\| \leq K_2 e^{-\alpha_1 k} \tag{13}$$

with $K_2, \alpha_1 > 0$.

Since k is a function of time we can bounded the right term of the previous inequality in terms of t :

$$K_2 e^{-\alpha_1 k} < K_2 e^{-\alpha_1 \frac{t-1}{h}} = K_2 e^{\frac{\alpha_1}{h}} e^{-\frac{\alpha_1}{h} t} = K_3 e^{-\alpha t} \tag{14}$$

with $K_{3,} \alpha > 0$.

So from (11), using (12) and (14) we conclude that:

$$\|z(t)\| = \left\| e^{\Lambda_c(t-t_k)} \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o(h-\tau)} e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k z_0 \right\| \leq K_1 K_3 e^{-\alpha t} \|z_0\| \ . \tag{15}$$

Necessity. We will now provide the necessity part of the theorem. We will do this by contradiction. Assume the system is stable and that $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o(h-\tau)} e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ has at least one eigenvalue outside the unit circle. Let us define $\Sigma(h) = e^{\Lambda_o(h-\tau)} e^{\Lambda_c(\tau)}$. Since the system is stable, a periodic sample of the response should converge to zero with time. We will take the samples at times t_{k+1}^-, that is, just before the loop is closed again. We will concentrate on a specific term: the state of the plant $x\left(t_{k+1}^-\right)$, which is the first element of $z\left(t_{k+1}^-\right)$. We will call $x\left(t_{k+1}^-\right)$, $\xi\left(k\right)$.

Now assume $\Sigma(\eta)$ has the following form:

$$\Sigma(\eta) = \begin{bmatrix} W(\eta) & X(\eta) \\ Y(\eta) & Z(\eta) \end{bmatrix} .$$

Then we can express the solution $z(t)$ as:

$$e^{\Lambda_c(t-t_k)}\left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}\Sigma(h)\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}\right)^k z_0 \tag{16}$$
$$= \begin{bmatrix} W(t-t_k) & X(t-t_k) \\ Y(t-t_k) & Z(t-t_k) \end{bmatrix}\begin{bmatrix} (W(h))^k & 0 \\ 0 & 0 \end{bmatrix} z_0$$
$$= \begin{bmatrix} W(t-t_k)\,(W(h))^k & 0 \\ Y(t-t_k)\,(W(h))^k & 0 \end{bmatrix} z_0\,.$$

Now, the values of the solution at times t_{k+1}^-, that is, just before the loop is closed again, are

$$z\left(t_{k+1}^-\right) = \begin{bmatrix} W(h)\,(W(h))^k & 0 \\ Y(h)\,(W(h))^k & 0 \end{bmatrix} z_0 = \begin{bmatrix} (W(h))^{k+1} & 0 \\ Y(h)\,(W(h))^k & 0 \end{bmatrix} z_0 \tag{17}$$

We also know that $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o(h-\tau)}e^{\Lambda_c(\tau)}\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ has at least eigenvalue outside the unit circle, which means that those unstable eigenvalues must be in $W(\tau)$. This means that the first element of $z\left(t_{k+1}^-\right)$, which we call $\xi(k)$, will in general grow with k. In other words we cannot ensure $\xi(k)$ will converge to zero for general initial condition x_0.

$$\left\|x\left(t_{k+1}^-\right)\right\| = \|\xi(k)\| = \left\|(W(h))^{k+1} x_0\right\| \to \infty \qquad \text{as} \quad k \to \infty\,, \tag{18}$$

which clearly means the system cannot be stable. Thus, we have a contradiction. ■

2.4 Examples

We ran simulations to verify the results suggested by the theory, which, in itself, is highly intuitive. Naturally, one would think that by using intermittent feedback as opposed to instantaneous closed loop control, there will be many things that will be gained in controlling the system. Indeed, one way to look at this, focusing in particular on the stability conditions derived above, is the following. Consider a control system with a certain plant model, then calculate the eigenvalues of the test matrix as h varies. This curve is very useful in that the stability of the system is determined by the maximum eigenvalue of its corresponding test matrix. So, by observing at which value of h the curve takes a maximum eigenvalue of 1, we are actually determining the range of h for stability.

The following plots are for A =[1 0;0 0] , B =[0;1], and model matrices $\hat{A}$ =[0.5 0; 0 0], $\hat{B}$ =[0; 0.25]. Our choice of controller was K =[-1 -1.5].

Figure 3 plots the maximum eigenvalue for a traditional MB-NCS, without intermittent feedback. Then, in Figure 4, we plot the maximum eigenvalue for various values of τ, as it increases as a percentage of h. Figure 5 shows the case where $\tau = 0.7h$. As we can see, increasing τ yields very significant benefits in terms of expanding the range of stability.

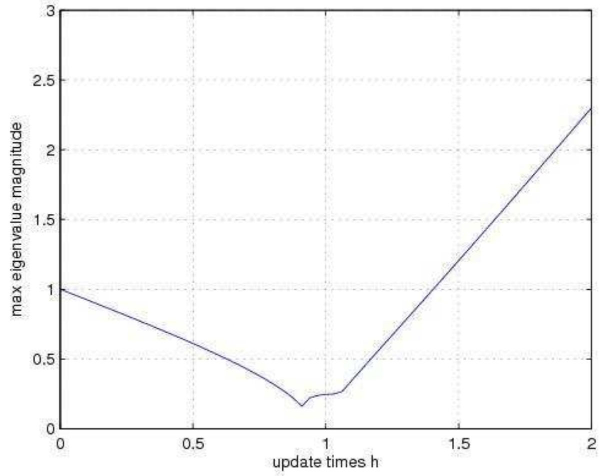

Figure 3: Maximum eigenvalue for traditional MB-NCS

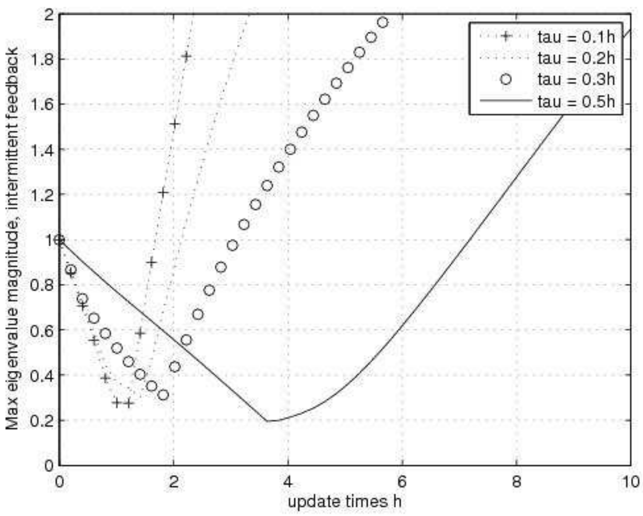

Figure 4: Maximum eigenvalue for MB-NCS with Intermittent Feedback, for various tau.

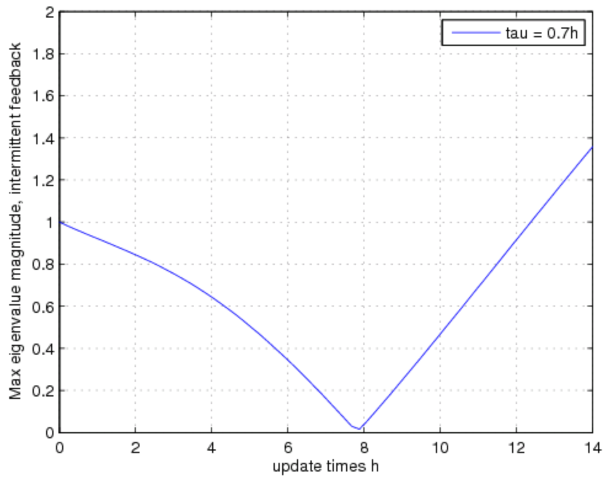

Figure 5: Maximum eigenvalue for MB-NCS with Intermittent Feedback, for tau = 0.7h

Figures 6 plots the model state of this system, for a high value of tau, $h = 0.5$, $\tau = 0.4$. The corresponding control signal is shown in Figure 8. The plots are repeated in Figures 9-11, this time for a low value of tau. As we expected, more intermittent feedback leads to smaller error between the plant state and model state and "softer" control input.

3 Output feedback case

In the previous section we considered plants where the full vector of the state was available at the output. When the state is not directly measurable, we must resort to a state observer. In this section we extend our results to this situation.

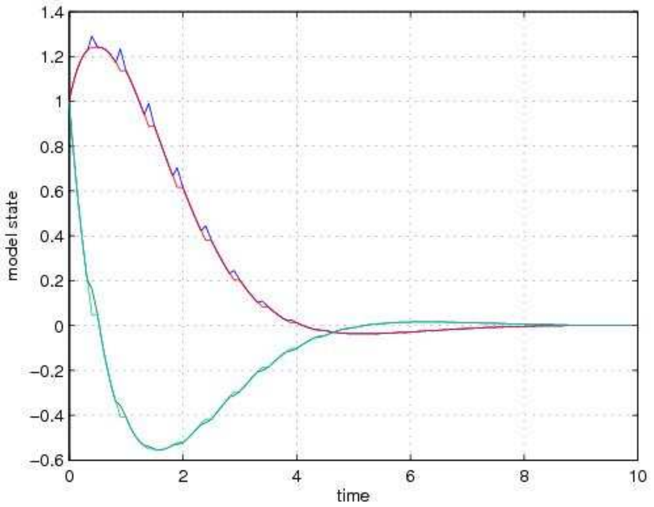

Figure 6: Model state, h=0.5, tau=0.4

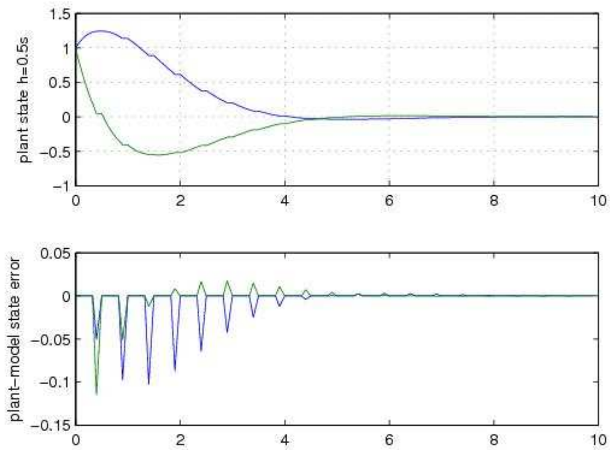

Figure 7: Model state, h=0.5, tau=0.4

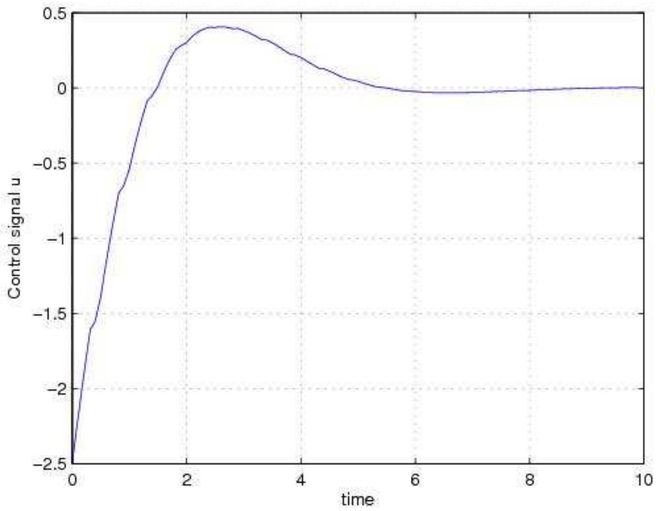

Figure 8: Control signal, h=0.5, tau=0.4

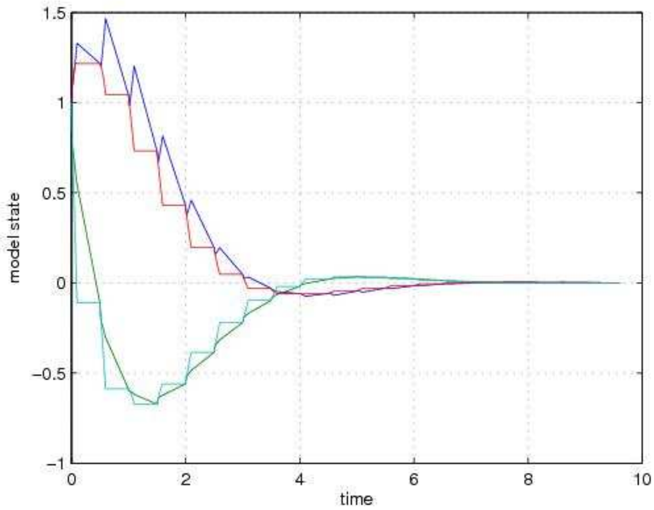

Figure 9: Model state, h=0.5, tau=0.1

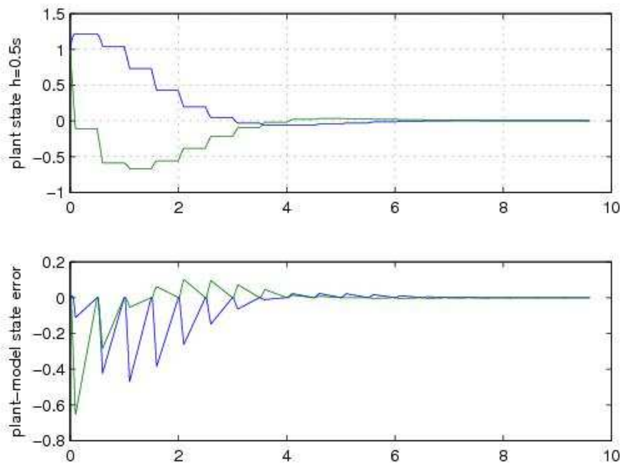

Figure 10: Plant state, h=0.5, tau=0.1

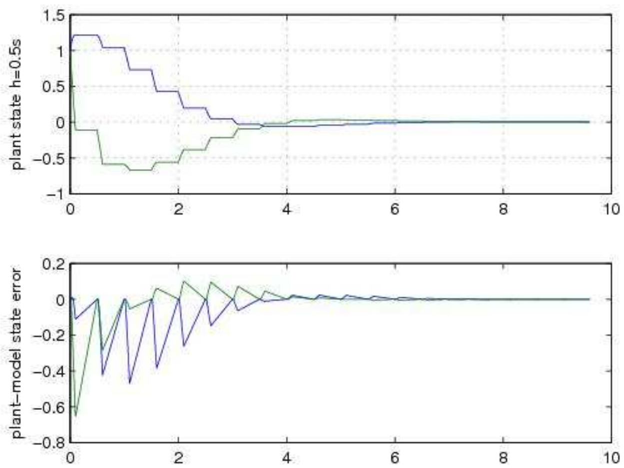

Figure 11: Control signal, h=0.5, tau=0.1

3.1 Problem formulation

As in the architecture used in [?] for instantaneous model-based feedback, we assume that the state observer is collocated with the sensor. We use the plant model to design the state observer. See Figure 12. Our configuration is based on the analogous setup for model-based control with output feedback, proposed by Montestruque.

In [?] it is justified that the sensor carry the computational load of an observer by the fact that, typically, sensors that can be connected to a network have an embedded processor (usually in charge of performing the sampling, filtering, etc.) inside. The observer has as inputs the output and input of the plant. In the implementation, in order to acquire the input, which is at the other side of the communication link, the observer can have a version of the model and controller, and knowledge of the update times τ and h. The controller and the observer are also synchronized.

The observer has the form of a standard state observer with gain L. It makes use of the plant model.

In summary, the system equations are the following:

Plant: $\dot{x} = Ax + Bu,\ y = Cx + Du$

Model: $\widehat{\dot{x}} = \hat{A}\hat{x} + \hat{B}u,\ y = \hat{C}x + \hat{D}u$

Controller: $u = K\hat{x}$

Observer: $\dot{\bar{x}} = (\hat{A} - L\hat{C})\bar{x} + \begin{bmatrix} \hat{B} - L\hat{D} & L \end{bmatrix} \begin{bmatrix} u \\ y \end{bmatrix}$

Controller model state: $\hat{x}$

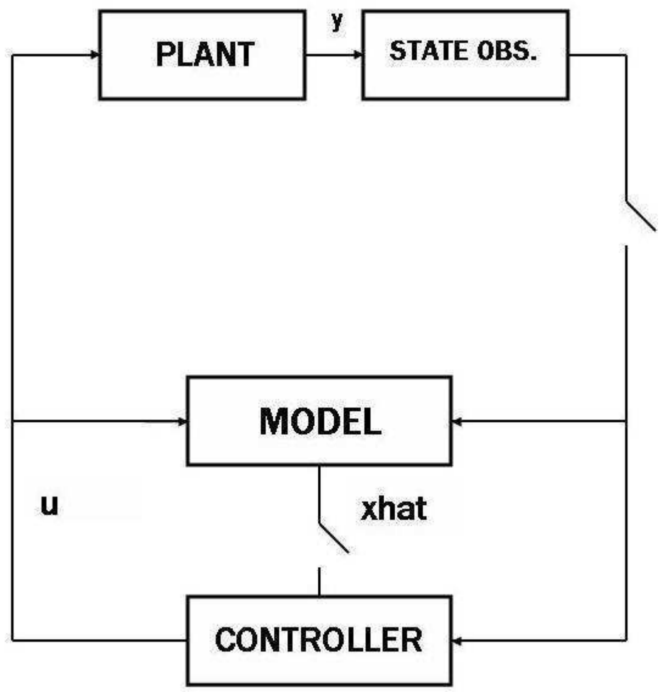

Figure 12: MB-NCS with intermittent feedback - state observer

Observer's estimate: $\bar{x}$

When loop is closed: $e = 0$

Error matrices: $\tilde{A} = A - \hat{A},\ \tilde{B} = B - \hat{B},\ \tilde{C} = C - \hat{C},\ \tilde{D} = D - \hat{D}$

We will derive the state response of the system in the following subsection.

3.2 State response of the system

To find the state response of the system, we proceed in the same fashion as we did before.

During open loop case, that is, when $t \in [t_k + \tau, t_{k+1})$, we have that

$$u = K\hat{x} \tag{19}$$

so

$$\begin{aligned} \dot{x} &= Ax + BK\hat{x} \\ \widehat{\dot{x}} &= (\hat{A} + \hat{B}K)\hat{x} \end{aligned} \tag{20}$$

and

$$\dot{\bar{x}} = (\hat{A} - L\hat{C})\bar{x} + \begin{bmatrix} \hat{B} - L\hat{D} & L \end{bmatrix} \begin{bmatrix} K\hat{x} \\ Cx + DK\hat{x} \end{bmatrix} \tag{21}$$

$$= \begin{bmatrix} LC & \hat{B}K + L\tilde{D}K & \hat{A} - LC \end{bmatrix} \begin{bmatrix} x \\ \hat{x} \\ \bar{x} \end{bmatrix}$$

We define $z = \begin{bmatrix} x \\ \bar{x} \\ e \end{bmatrix}$ with initial condition $\hat{x}\,(t_k) = \bar{x}\,(t_k)\,.$

Thus,

$$\dot{z} = \Lambda_o z \tag{22}$$

where $\Lambda_o = \begin{bmatrix} A & BK & -BK \\ LC & \hat{A} - L\hat{C} + \hat{B}K + L\tilde{D}K & -\hat{B}K - L\tilde{D}K \\ LC & L\tilde{D}K - L\hat{C} & A - L\tilde{D}K \end{bmatrix}$

and

$$z\,(t_k + \tau) = \begin{bmatrix} x(t_k + \tau) \\ \bar{x}(t_k + \tau) \\ e(t_k + \tau) \end{bmatrix} = \begin{bmatrix} x(t_k + \tau)^- \\ \bar{x}(t_k + \tau)^- \\ 0 \end{bmatrix}$$

Similarly, for the closed loop case, that is, when $t \in [t_k, t_k + \tau)$, we have

$$\dot{z} = \Lambda_c z \tag{23}$$

where $\Lambda_c = \begin{bmatrix} A & BK & -BK \\ LC & \hat{A} - L\hat{C} + \hat{B}K + L\tilde{D}K & -\hat{B}K - L\tilde{D}K \\ 0 & 0 & 0 \end{bmatrix}$ because the error is always zero.

From this, it should be quite clear that given an initial condition $z\,(t=0) = z_0$, then after a certain time $t \in [0, \tau)$, the solution of the trajectory of the vector is

$$z\,(t) = e^{\Lambda_c(t)} z_0, \;\; t \in [0, \tau) \tag{24}$$

In particular,

$$z\,(\tau) = e^{\Lambda_c(\tau)} z_0 \tag{25}$$

Once the loop is opened

$$z\,(t) = e^{\Lambda_o(t-\tau)} z\,(\tau) = e^{\Lambda_o(t-\tau)} e^{\Lambda_c(\tau)} z_0 \;, \quad t \in [\tau, t_1) \tag{26}$$

We close the loop again at $t = h$.

$$z\left(t_1^-\right) = e^{\Lambda_o(h-\tau)} e^{\Lambda_c(\tau)} z_0 \tag{27}$$

But we must reset the error to zero, so we pre- and post-multiply by $\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}$.

After going through k cycles, we find that

$$z\left(t_k\right) = \Sigma^k z_0 \tag{28}$$

where $\Sigma = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_o(h-\tau)} e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}$.

Taking into account the last (partial) cycle,

$$z\left(t\right) = \begin{cases} e^{\Lambda_c(t-t_k)} \Sigma^k z_0 \ , \ t \in [t_k, t_k + \tau) \\ e^{\Lambda_o(t-(t_k+\tau))} e^{\Lambda_c(\tau)} \Sigma^k z_0 \ , \ t \in [t_k + \tau, t_{k+1}) \end{cases} \tag{29}$$

where $\Sigma = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_o(h-\tau)} e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}$, and Λ_o, Λ_c as before.

We summarize the result in this proposition.

Proposition 4 *The system described above has a state response:*

$$z\left(t\right) = \begin{cases} e^{\Lambda_c(t-t_k)} \Sigma^k z_0 \ , \ t \in [t_k, t_k + \tau) \\ e^{\Lambda_o(t-(t_k+\tau))} e^{\Lambda_c(\tau)} \Sigma^k z_0 \ , \ t \in [t_k + \tau, t_{k+1}) \end{cases} \tag{30}$$

where $\Sigma = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_o(h-\tau)} e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}$, *and* $\Lambda_o = \begin{bmatrix} A & BK \\ LC & \hat{A} - L\hat{C} + \hat{B}K + L\tilde{D}K \\ LC & L\tilde{D}K - L\hat{C} \end{bmatrix}$

$$\Lambda_c = \begin{bmatrix} A & BK & -BK \\ LC & \hat{A} - L\hat{C} + \hat{B}K + L\tilde{D}K & -\hat{B}K - L\tilde{D}K \\ 0 & 0 & 0 \end{bmatrix}.$$

In the next subsection, we obtain a necessary and sufficient condition for stability.

3.3 Stability condition

As before, we provide a necessary and sufficient condition for stability.

Theorem 5 *The system described above is globally exponentially stable around the solution* $z = \begin{bmatrix} x \\ \bar{x} \\ e \end{bmatrix} = \mathbf{0}$ *if and only if the eigenvalues of* Σ *are strictly inside the unit circle, where where* $\Sigma = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_o(h-\tau)} e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}$, *and* Λ_o, Λ_c *as before.*

Proof. Sufficiency. We will do it for $[t_k, t_k+\tau)$, but it's the same otherwise too.

$$\begin{aligned} \|z(t)\| &= \left\| e^{\Lambda_c (t-t_k)} \Sigma^k z_0 \right\| \qquad (31) \\ &\leq \left\| e^{\Lambda_c (t-t_k)} \right\| \left\| \Sigma^k \right\| \|z_0\| \end{aligned}$$

$$\begin{aligned} \left\| e^{\Lambda_c (t-t_k)} \right\| &\leq 1 + (t-t_k)\,\bar{\sigma}(\Lambda_c) + \frac{(t-t_k)^2}{2!} + \cdots \\ &= e^{\bar{\sigma}(\Lambda_c)(t-t_k)} \leq e^{\bar{\sigma}(\Lambda_c)(\tau)} = K_1 \end{aligned} \tag{32}$$

And $\left\| \left(\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_o (h-\tau)} e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} \right)^k \right\|$ is clearly bounded if and only if the eigenvalues of Σ are within the unit circle.

$$\left\| \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_o (h-\tau)} e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} \right\|^k \leq K_2 e^{-\alpha_1 k} \ , \quad K_2, \alpha_1 > 0 \tag{33}$$

Since k is a function of time, we can bound the right term in terms of t.

$$K_2 e^{-\alpha_1 k} \leq K_2 e^{-\alpha_1 \frac{t-1}{h}} \leq K_2 e^{\alpha_1/h} e^{-\alpha_1 t/h} = K_3 e^{-\alpha t} \ , \quad K_3, \alpha_1 > 0 \tag{34}$$

Thus,

$$\left\| z(t) \right\| = \left\| e^{\Lambda_c (t-t_k)} \Sigma^k z_0 \right\| \leq K_1 K_3 e^{-\alpha t} \left\| z_0 \right\| \tag{35}$$

Necessity. Assume that $\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_o (h-\tau)} e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}$ has at least one eigenvalue outside the unit circle. We will take samples... as we did in the case without the observer. Let's call $\Sigma(h) = e^{\Lambda_o(h-\tau)}e^{\Lambda_c(\tau)}$. We will concentrate on $\xi(k) = \begin{bmatrix} x\left(t_{k+1}^-\right) \\ \bar{x}\left(t_{k+1}^-\right) \end{bmatrix}$.

Assume $\Sigma(\eta) = \begin{bmatrix} W_1(\eta) & W_2(\eta) & X_1(\eta) \\ W_3(\eta) & W_4(\eta) & X_2(\eta) \\ Y_1(\eta) & Y_2(\eta) & Z(\eta) \end{bmatrix}$

For simplicity, let's call

$$W(\eta) = \begin{bmatrix} W_1(\eta) & W_2(\eta) \\ W_3(\eta) & W_4(\eta) \end{bmatrix}, \ X(\eta) = \begin{bmatrix} X_1(\eta) \\ X_2(\eta) \end{bmatrix}, \ Y(\eta) = \begin{bmatrix} Y_1(\eta) & Y_2(\eta) \end{bmatrix} \tag{36}$$

Then we can express $z(t)$ as

$$\begin{aligned} & e^{\Lambda_c(t-t_k)}\left(\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}\Sigma(h)\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}\right)^k z_0 \\ &= \begin{bmatrix} W(t-t_k) & X(t-t_k) \\ Y(t-t_k) & Z(t-t_k) \end{bmatrix}\begin{bmatrix} (W(h))^k & \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 \end{bmatrix} & 0 \end{bmatrix} z_0 \\ &= \begin{bmatrix} W(t-t_k)(W(h))^k & \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ Y(t-t_k)(W(h))^k & 0 \end{bmatrix} z_0 \end{aligned} \tag{37}$$

We know $\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} \Sigma(h) \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}$ has at least one eigenvalue outside the unit circle, thus those unstable eigenvalues must be in $W(h)$. This means that the first two elements of $z\left(t_{k+1}^{-}\right)$, which we call $\xi(k)$, will in general grow with k (if one selects initial condition z_0 along the eigenvector of the corresponding eigenvalue).

Thus, we cannot ensure $\xi(k)$ will converge to zero for a general condition.

$$\left\| \begin{bmatrix} x\left(t_{k+1}^{-}\right) \\ \bar{x}\left(t_{k+1}^{-}\right) \end{bmatrix} \right\| = \left\| \xi(k) \right\| = \left\| (W(h))^k \begin{bmatrix} x_0 \\ \bar{x}_0 \end{bmatrix} \right\| \rightarrow \infty \text{ as } k \rightarrow \infty \tag{38}$$

This means the system is unstable; thus we have a contradiction. ∎

3.4 Examples

We now run simulations to illustrate the above results. Figure 13 displays the model and plant state for a high value of τ, while an analogous plots are displayed in Figure 14 for low values. Finally, in Figure 15 we show the maximum eigenvalue of the system (the system becomes unstable when this value exceeds 1), verifying the added stability range provided by increased intermittent feedback.

For the purpose of these simulations, we used the following values: A =[0 1;0 0.25], B =[0;1], C =[1 0], D =0, $\hat{A}$ =[0.0958 1.0604; -0.0066 -0.0134],

$\hat{B}$ =[-0.0518; 1.0269], $\hat{C}$ =[0.9734 -0.0137], $\hat{D}$ =-.0396, K =[-1 -2], L =[20;100].

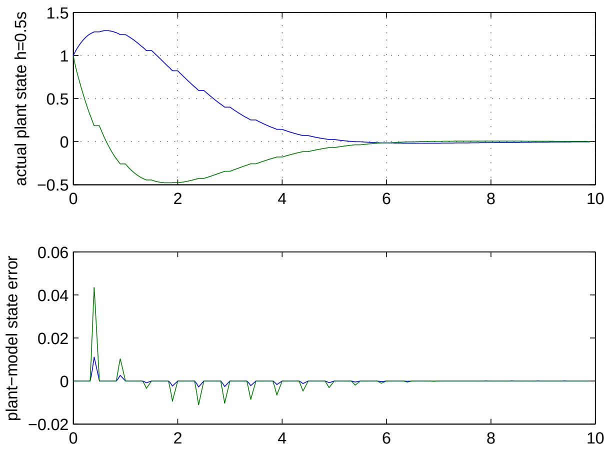

Figure 13: Plant and model state. State observer case, h = 0.5, tau = 0.4

The above results are useful for situations when the full state of the plant is unavailable. An extension of our results to nonlinear plants is presented in the next section.

4 Delays

In the previous sections, we have assumed that the delays in the network are negligible. However, in reality, this is usually not the case. We now consider the case where delays in the network are present. Although in real-life plants delays might be variable, for the sake of analysis we will consider the case where delays are constant and known.

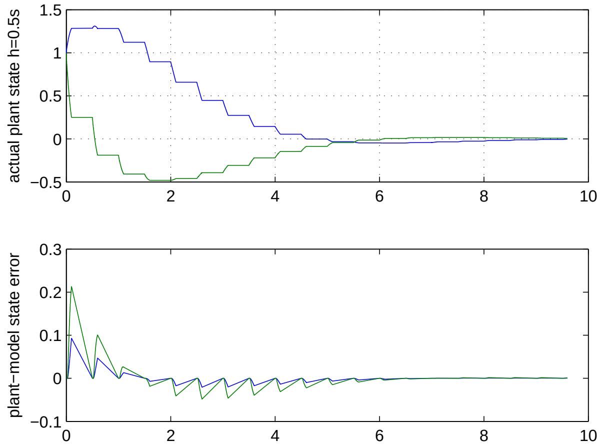

Figure 14: Plant and model state. State observer case, h = 0.5, tau = 0.1

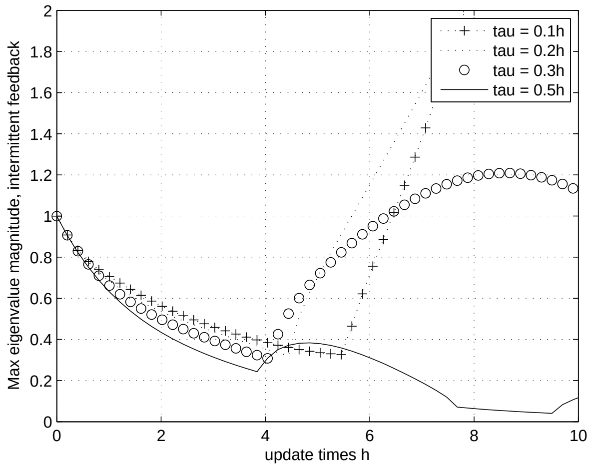

Figure 15: Maximum eigenvalue search. State observer case, intermittent feedback

4.1 Problem formulation

Consider the following setup:

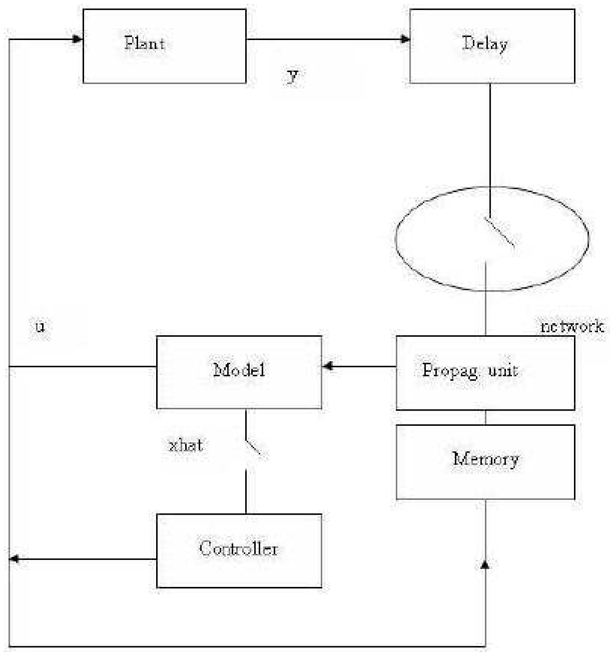

Figure 16: MB-NCS with intermittent feedback - delay case

The corresponding equations are as follows:

Plant: $\dot{x} = Ax + Bu$

Model: $\widehat{\dot{x}} = \hat{A}\hat{x} + \hat{B}u$

Controller: $u = K\hat{x}\ ,\ \ t \in [t_k, t_{k+1})$

Propagation unit: $\breve{x}' = \breve{A}\breve{x} + \breve{B}u\ ,\ \ t \in [t_{k+1} - \tau_d, t_{k+1}]$

Update law: $\breve{x} \longleftarrow x,\ t = t_{k+1} - \tau_d\ \ ;\ \ \hat{x} \longleftarrow \breve{x},\ t = t_k$

This setup follows the original one proposed my Montestruque for the case with instantaneous feedback. The explanation is as follows: "to ease the analysis, we initialize the propagation unit at time $t_{k+1} - \tau_d$ with the state vector the sensor obtains. We then run the plant, model, and propagation unit together until t_{k+1}. At this time, the model is updated with the propagation unit state vector, as described in the update law. This is equivalent to having the propagation unit receive the state vector $x\left(t_{k+1} - \tau_d\right)$ at t_{k+1} and propagate it instantaneously to t_{k+1}." []

To understand this better, let us consider again the setup. As we can see, there are two different times when certain values are reset:

At times t_{k+1}, $\hat{x} \longleftarrow \breve{x}$. Thus, $\hat{e} = 0$ (where $\hat{e} = \breve{x} - \hat{x}$).

At times $t_{k+1} - \tau_d$, $\breve{x} \longleftarrow x$. Thus, $\breve{e} = 0$ (where $e = x - \breve{x}$).

The loop closes at times $t_k, t_{k+1}, t_{k+2},$■ and this happens every h seconds. Now, at this time when the loop closes, information about the plant will pass from the sensor to the propagation unit. Unfortunately, because of the delay τ_d, this is old information. So, instead of $x\left(t_{k+1}\right)$, what the propagation unit receives is $x\left(t_{k+1} - \tau_d\right)$.

Now, the role of the propagation unit is that of a predictor. It uses the old information to predict the actual current state of the plant, that is, it takes old information and predicts current information. We will assume that the processing delay at this stage is negligible, so the propagation unit produces an estimate (or "prediction"), $\breve{x}\left(t_{k+1}\right)$ and this value is fed to the model at time as well. This is why the update rule at t_{k+1} is as stated.

Once the model receives its update, it will run on its own for h seconds, as in the traditional setup, until it receives its next update.

The question arises: if all this is happening at t_{k+1}, why the update $\breve{x} \longleftarrow x$ at time $t_{k+1} - \tau_d$? Furthermore, how is this even possible if, when the loop is closed, the information that the propagation unit receives is delayed? Is the loop closed at any other time? No, what happens is that $\breve{x}$, the state of the propagation unit, is a sort of dummy variable. In fact, if at time $t_{k+1} - \tau_d$ we took a scope and measured the value of $\breve{x}$, the actual value would be different from the $\breve{x}\left(t_{k+1} - \tau_d\right)$ we will be using in the analysis. The setup is such that the analysis is performed *as if* the propagation unit received updated information at $t_{k+1} - \tau_d$ and then used its model to produce the estimate. This is not what is really happening in practice. The propagation unit does not actually receive the update until the loop is closed at time t_{k+1}. However, as we know, this is old information: $x\left(t_{k+1} - \tau_d\right)$.

But since we do not actually care about the actual state of the propagation unit at times when it is not feeding information to the model, we can just say that, from the point of view of the propagation unit, the information was in fact received at time . Thus the update law $\breve{x} \longleftarrow x$ at time $t_{k+1} - \tau_d$.

The next question that may arise is, then, how the propagation unit actually makes its estimate. The propagation unit is set to $\breve{x}\left(t_{k+1} - \tau_d\right)$ and has the same $\hat{A}, \hat{B}$ as the plant model, but notice it also has memory. What is being stored in this memory are the set of control actions $u\left(t\right)$ generated by the model during the previous cycle, in particular, during the last τ_d seconds of the past cycle. The propagation unit uses the set of control actions to predict (or "propagate") the value to a "new," updated value: $\breve{x}(t_{k+1})$. This is the value which the model uses for its updates when the loop is closed. Once again, note that we are assuming the processing delay in the propagation unit is negligible compared to the network delay.

4.2 State response of the system

The following is the development of the system response of for the case of intermittent feedback with delays. We use the setup described previously.

Also, let us recall that, from the analysis Montestruque made for the instantaneous update case,

$$\dot{z}(t) = \Lambda_o z(t), \tag{39}$$

where $\Lambda_o = \begin{bmatrix} A+BK & -BK & -BK \\ \tilde{A}+\tilde{B}K & \hat{A}-\tilde{B}K & -\tilde{B}K \\ 0 & 0 & \hat{A} \end{bmatrix}$ and $z = \begin{bmatrix} x \\ \breve{e} \\ \hat{e} \end{bmatrix}$, with appropriate initial conditions and reset errors.

For the closed loop case, we have dynamics governed by a new matrix, due to the error $\hat{e}$ always being zero,

$$\dot{z}(t) = \Lambda_c z(t) \tag{40}$$

where $\Lambda_c = \begin{bmatrix} A+BK & -BK & -BK \\ \tilde{A}+\tilde{B}K & \hat{A}-\tilde{B}K & -\tilde{B}K \\ 0 & 0 & 0 \end{bmatrix}$.

Let us start out by looking at the interval $t \in [0, \tau)$. During this interval, the system will behave according to closed loop dynamics. So, clearly,

$$z(t) = e^{\Lambda_c(t-t_0)} z_0 \,, \quad t \in [0, \tau) \tag{41}$$

At time $t = \tau^-$ we have

$$z(t) = e^{\Lambda_c \tau} z_0 \;, \; t = \tau^- \tag{42}$$

But when we get to $t = \tau$, we have to make sure $\hat{e}$ is zero. In fact, as this error should have been reset as soon as the loop as closed, we should both pre- and post-multiply by $\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}$.

Thus,

$$z(t) = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_c \tau} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} z_0 \;, \quad t = \tau \tag{43}$$

The next interval, $t \in [t_0 + \tau, t_1 - \tau_d)$, is governed by open loop dynamics, so

$$z(t) = e^{\Lambda_o(t-(t_0+\tau))} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_c \tau} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} z_0 \ , \quad t \in [t_0+\tau, t_1-\tau_d) \tag{44}$$

So at time $t = (t_1 - \tau_d)^-$

$$z(t) = e^{\Lambda_o(h-\tau_d-\tau))} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_c \tau} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} z_0 \ , \quad t = (t_1-\tau_d)^- \tag{45}$$

But here we reset $\breve{e}$ and $\hat{e} = \hat{e} + \breve{e}$

So at $t = t_1 - \tau_d$,

$$z(t) = \begin{bmatrix} I & 0 & 0 \\ 0 & 0 & 0 \\ 0 & I & I \end{bmatrix} e^{\Lambda_o(h-\tau_d-\tau))} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_c \tau} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} z_0 \ , \quad t = t_1 - \tau_d \tag{46}$$

For the next interval, that is $t \in [t_1 - \tau_d, t_1)$

$$z(t) = e^{\Lambda_o(t-(t_1-\tau_d))} \begin{bmatrix} I & 0 & 0 \\ 0 & 0 & 0 \\ 0 & I & I \end{bmatrix} e^{\Lambda_o(h-\tau_d-\tau))} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_c\tau} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} z_0 \ , \quad t \in [t_1-\tau_d, t_1) \tag{47}$$

At $t = t_1^-$,

$$z(t) = e^{\Lambda_o\tau_d} \begin{bmatrix} I & 0 & 0 \\ 0 & 0 & 0 \\ 0 & I & I \end{bmatrix} e^{\Lambda_o(h-\tau_d-\tau))} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_c\tau} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} z_0 \ , \quad t = t_1^- \tag{48}$$

But because of update rule at $t = t_1$

$$z(t_1) = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_o\tau_d} \begin{bmatrix} I & 0 & 0 \\ 0 & 0 & 0 \\ 0 & I & I \end{bmatrix} e^{\Lambda_o(h-\tau_d-\tau))} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_c\tau} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} z_0 \tag{49}$$

So, clearly, we have the following general solution:

For $t \in [t_k, t_k + \tau)$

$$z\,(t) = e^{\Lambda_c(t-t_k)}\Sigma^k z_0 \ , \quad t \in [t_k, t_k + \tau) \tag{50}$$

For $t \in [t_k + \tau, t_{k+1} - \tau_d)$

$$z(t) = e^{\Lambda_o(t-(t_k+\tau))} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_c \tau} \Sigma^k z_0 \ , \quad t \in [t_k + \tau, t_{k+1} - \tau_d) \tag{51}$$

For $t \in [t_{k+1} - \tau_d, t_{k+1})$

$$z(t) = e^{\Lambda_o(t-(t_{k+1}-\tau_d))} \begin{bmatrix} I & 0 & 0 \\ 0 & 0 & 0 \\ 0 & I & I \end{bmatrix} e^{\Lambda_o(h-\tau_d-\tau)} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_c \tau} \Sigma^k z_0 \ , \quad t \in [t_{k+1} - \tau_d, t_{k+1}) \tag{52}$$

where

$$\Sigma = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_o \tau_d} \begin{bmatrix} I & 0 & 0 \\ 0 & 0 & 0 \\ 0 & I & I \end{bmatrix} e^{\Lambda_o(h-\tau_d-\tau))} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_c \tau} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} . \tag{53}$$

4.3 Stability condition

We obtain a stability condition as we did for the cases without delays. It is of the same form, depending on the eigenvalues of a test matrix.

Theorem 6 *The system described above is globally exponentially stable around* *the solution* $z = \begin{bmatrix} x \\ \breve{e} \\ \hat{e} \end{bmatrix} = \mathbf{0}$ *if and only if the eigenvalues of* Σ *are strictly inside the unit circle, where where* $\Sigma = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_o \tau_d} \begin{bmatrix} I & 0 & 0 \\ 0 & 0 & 0 \\ 0 & I & I \end{bmatrix} e^{\Lambda_o(h-\tau_d-\tau))} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} e^{\Lambda_c}$ *and* Λ_o, Λ_c *as before.*

Proof. The proof is performed in the same way as that of the previous cases. ■

5 Nonlinear plants

In the previous sections we have restricted our study to the cases where the plant is linear. Let us now lift this restriction and seek to find the corresponding stability properties for nonlinear plants with intermittent feedback.

The setup and procedure that follows closely mirrors that proposed by Montestruque [17] for traditional MB-NCS. The sufficient conditions obtained relate the stability of the nonlinear MB-NCS with the value of a function that depends on the Lipschitz constants of the plant and model as well as the stability properties of the compensated non-networked model. The results are obtained by studying the worst-case behavior of the norm of the plant state and the error, thus leading to conservative results.

.

5.1 Stability of a class of nonlinear MB-NCS

Let the plant be given by:

$$\dot{x} = f(x) + g\,(u) \tag{54}$$

We use a model on the actuator side of the plant to estimate the actual state of the plant. The controller will be assumed to be a nonlinear state feedback controller. The control signal u is generated by taking into account the plant model state . The plant state sensor will send through the network the real value of the plant state to the model (that is, the loop will be closed) every h seconds, and the loop will remain closed for τ seconds during each cycle. During these times, the state of the model is set to be the same as that of the plant. We will assume the plant model dynamics are given by:

$$\dot{\hat{x}} = \hat{f}(x) + \hat{g}\,(u) \tag{55}$$

And the controller has the following form:

$$u = \hat{h}\,(\hat{x}) \tag{56}$$

We define as the error between the plant state and the plant model state,

$e = x - \hat{x}$. Combining the above, we obtain:

$$\begin{aligned} \dot{x} &= f(x) + g\left(\hat{h}\left(\hat{x}\right)\right) = f\left(x\right) + m(\hat{x}) \\ \dot{\hat{x}} &= \hat{f}(x) + \hat{g}\left(\hat{h}\left(\hat{x}\right)\right) = f\left(x\right) + \hat{m}\left(\hat{x}\right) \end{aligned} \tag{57}$$

Assume also that the plant model dynamics differ from the actual plant dynamics in an additive fashion:

$$\begin{aligned} \hat{f}\left(\zeta\right) &= f\left(\zeta\right) + \delta_f\left(\zeta\right) \\ \hat{m}\left(\zeta\right) &= m\left(\zeta\right) + \delta_m\left(\zeta\right) \end{aligned} \tag{58}$$

Thus:

$$\begin{aligned} \dot{x} &= f\left(x\right) + m\left(\hat{x}\right) \\ \dot{\hat{x}} &= f\left(x\right) + \hat{m}\left(\hat{x}\right) + \delta_f\left(\hat{x}\right) + \delta_m\left(\hat{x}\right) \end{aligned} \tag{59}$$

Assume that f and δ satisfy the following local Lipschitz conditions for with $x, y \in B_L$, a ball centered on the origin:

$$\begin{aligned} \left\|f\left(x\right) - f\left(y\right)\right\| &\leq K_f \left\|x - y\right\| \\ \left\|\delta\left(x\right) - \delta\left(y\right)\right\| &\leq K_\delta \left\|x - y\right\| \end{aligned} \tag{60}$$

It is to be noted that if the plant model is accurate the Lipschitz constant K_δ will be small.

Assume that the non-networked compensated plant model is exponentially stable when $\hat{x}(t_0) \in B_S$, $\hat{x}(t) \in B_\tau$, for $t \in [t_0, t_0 + \tau)$ with B_S and B_τ balls centered on the origin.

$$\|\hat{x}(t)\| \leq \alpha \|\hat{x}(t_0)\| e^{-\beta(t-t_0)} \text{ with } \alpha, \beta > 0\,. \tag{61}$$

Theorem 7 *The non-linear MB-NCS with dynamics described above, and that satisfies the Lipschitz conditions described and with exponentially stable compensated plant model satisfying is asymptotically stable if:*

$$\left(1 - \alpha\left(e^{-\beta(h-\tau)} + \left(e^{K_f(h-\tau)} - e^{-\beta(h-\tau)}\right)\left(\frac{K_\delta}{K_f + \delta}\right)\right)\right) > 0 \tag{62}$$

Proof. We will now analyze the behavior of the plant state norm when the loop is open. The stability of the system can be guaranteed if $\|x(t)\|$ decreases such that $\|x(t_k + \tau)\| > \|x(t_{k+1})\|$, where $t_k+\tau$ is the time the loop is opened and t_{k+1} is the next time the loop is closed, with $t_{k+1} - t_k + \tau = h - \tau$.

In general, we see that in any interval $[t_k + \tau, t_{k+1})$ the following holds true:

$$\begin{aligned} \|x\| = \|\hat{x} + e\| &< \|\hat{x}\| + \|e\| \\ \|e(t_k + \tau)\| &= 0 \\ \|x(t_k + \tau)\| &= \|\hat{x}(t_k + \tau)\| \end{aligned} \tag{63}$$

So, we can guarantee that $\|x\|$ will decrease over the interval $[t_k + \tau, t_{k+1})$

if $\|\hat{x}\| + \|e\|$ decrease.

We know that:

$$\dot{e} = \dot{x} - \dot{\hat{x}} = f(x) - f(\hat{x}) - \delta(\hat{x}) \tag{64}$$

Thus:

$$\begin{aligned} e(t) &= e(t_k + \tau) \int_{t_k+\tau}^{t} \left(f(x(s)) - f(\hat{x}(s)) - \delta(\hat{x}(s)) \right) ds \\ &= \int_{t_k+\tau}^{t} \left(f(x(s)) - f(\hat{x}(s)) - \delta(\hat{x}(s)) \right) ds \,, \quad \forall t \in [t_k + \tau, t_k) \end{aligned} \tag{65}$$

The last equality holds since at $t_k + \tau$ the plant model state is updated and the error is equal to zero. We will now use the Lipschitz condition to bound the norm of the error.

$$\begin{aligned} \|e(t)\| &\leq \int_{t_k+\tau}^{t} \left(\|f(x(s)) - f(\hat{x}(s))\| + \|\delta(\hat{x}(s))\| \right) ds \\ &\leq \int_{t_k+\tau}^{t} \left(K_f \|(x(s)) - (\hat{x}(s))\| + K_\delta \|\hat{x}(s)\| \right) ds \\ &= K_f \int_{t_k+\tau}^{t} \|(x(s)) - (\hat{x}(s))\| \, ds + K_\delta \int_{t_k+\tau}^{t} \|\hat{x}(s)\| \, ds \\ &= K_f \int_{t_k+\tau}^{t} \|e(s)\| \, ds + K_\delta \int_{t_k+\tau}^{t} \|\hat{x}(s)\| \, ds \,, \ \forall t \in [t_k + \tau, t_k) \end{aligned} \tag{66}$$

Then:

$$\begin{aligned}
\|e(t)\| &\le K_f \int_{t_k+\tau}^{t} \|e(s)\|\, ds + K_\delta \int_{t_k+\tau}^{t} \|\hat{x}(s)\|\, ds \qquad (67)\\
&= K_f \int_{t_k+\tau}^{t} \|e(s)\|\, ds + K_\delta \int_{t_k+\tau}^{t} \alpha \|\hat{x}(t_k+\tau)\| e^{-\beta(t-t_k+\tau)} ds\\
&= K_f \int_{t_k+\tau}^{t} \|e(s)\|\, ds + K_\delta \frac{\alpha}{\beta} \|\hat{x}(t_k+\tau)\| \left(1 - e^{-\beta(t-t_k+\tau)}\right),\ \forall t \in [t_k+\tau, t_k)
\end{aligned}$$

We now use the Gronwall-Bellman Inequality [] for the following step. This inequality states that if a continuous real-valued function $y(t)$ satisfies $y(t) < \lambda(t) + \int_a^t \mu(s)\, y(s)\, ds$ with $\lambda(t)$ and $\mu(t)$ continuous real-valued functions and $\mu(t)$ non-negative for $t \in [a, b)$, then $y(t) < \lambda(t) + \int_a^t \lambda(s)\, \mu(s)\, e^{\int_s^t u(\psi) d\psi} ds$ over the same interval. So, we assign $y(t) = \|e(t)\|$,

$\lambda(t) = K_\delta \frac{\alpha}{\beta} \|\hat{x}(t_k+\tau)\| \left(1 - e^{-\beta(t-t_k+\tau)}\right)$, and $\mu(t) = K_f$, and thus obtain:

$$\|e(t)\| \leq K_\delta \frac{\alpha}{\beta} \|\hat{x}(t_k+\tau)\| \left(1 - e^{-\beta(t-t_k+\tau)}\right) \tag{68}$$

$$+ \int_{t_k+\tau}^{t} K_\delta \frac{\alpha}{\beta} \|\hat{x}(t_k+\tau)\| \left(1 - e^{-\beta(s-t_k+\tau)}\right) K_f e^{K_f(t-s)} ds \tag{69}$$

$$= K_\delta \frac{\alpha}{\beta} \|\hat{x}(t_k+\tau)\| \left(1 - e^{-\beta(t-t_k+\tau)} + \int_{t_k+\tau}^{t} \left(1 - e^{-\beta(s-t_k+\tau)}\right) K_f e^{K_f(t-s)} ds\right)$$

$$= K_\delta \frac{\alpha}{\beta} \|\hat{x}(t_k+\tau)\| \left(1 - e^{-\beta(t-t_k+\tau)} + K_f \int_{t_k+\tau}^{t} e^{K_f(t-s)} - e^{-\beta(s-t_k+\tau)} e^{K_f(t-s)} ds\right)$$

$$= K_\delta \frac{\alpha}{\beta} \|\hat{x}(t_k+\tau)\| \left(1 - e^{-\beta(t-t_k+\tau)} + K_f \int_{t_k+\tau}^{t} e^{K_f(t-s)} - e^{K_f t - K_f s - \beta s + \beta(t_k+\tau)} ds\right)$$

$$= K_\delta \frac{\alpha}{\beta} \|\hat{x}(t_k+\tau)\| \Bigg(1 - e^{-\beta(t-t_k+\tau)} + K_f \Bigg(\frac{-1}{K_f} \left(1 - e^{K_f(t-(t_k+\tau))}\right) + \frac{1}{K_f+\beta} \big(e^{-\beta(t-(t_k+\tau}$$

$$= K_\delta \frac{\alpha}{\beta} \|\hat{x}(t_k+\tau)\| \Bigg(1 - e^{-\beta(t-t_k+\tau)} - 1 + e^{K_f(t-(t_k+\tau))} + \frac{K_f}{K_f+\beta} \big(e^{-\beta(t-(t_k+\tau))} - e^{K_f(t-(t}$$

$$= K_\delta \frac{\alpha}{\beta} \|\hat{x}(t_k+\tau)\| \left(e^{K_f(t-(t_k+\tau))} - e^{-\beta(t-(t_k+\tau))}\right) \left(1 - \frac{K_f}{K_f+\beta}\right)$$

$$= K_\delta \|\hat{x}(t_k+\tau)\| \left(e^{K_f(t-(t_k+\tau))} - e^{-\beta(t-(t_k+\tau))}\right) \left(\frac{\alpha}{K_f+\beta}\right) \ , \ \forall t \in [t_k+\tau, t_k)$$

Note that he error signal will be zero if the update time $h - \tau = t_{k+1} - (t_k+\tau)$ is zero (or if the model is perfect, that is, same dynamics as the plant). With this bound over the error signal we can proceed to calculate

Tomas Estrada and P.J. Antsaklis, Results on Continuous and Discrete Model-Based Networked Control Systems with Intermittent Feedback, Part I: Stability, ISIS Technical Report, University of Notre Dame, ISIS-2008-001, August 2008.

the bound over the plant state.

$$\begin{aligned}\|x(t)\| &\leq \|\hat{x}(t)\| + \|e(t)\| \\ &\leq \alpha \|\hat{x}(t_k+\tau)\| e^{-\beta(t-(t_k+\tau))} + K_\delta \|\hat{x}(t_k+\tau)\| \left(e^{K_f(t-(t_k+\tau))} - e^{-\beta(t-(t_k+\tau))}\right)\left(\frac{\alpha}{K_f+\beta}\right) \qquad (70)\\ &= \alpha \|\hat{x}(t_k+\tau)\| \left(e^{-\beta(t-(t_k+\tau))} + \left(e^{K_f(t-(t_k+\tau))} - e^{-\beta(t-(t_k+\tau))}\right)\left(\frac{K_\delta}{K_f+\beta}\right)\right) \qquad (71)\end{aligned}$$

$, \forall t \in [t_k+\tau, t_k)$

For stability, we need$\|\hat{x}(t_k+\tau)\| > \|\hat{x}(t_{k+1})\|$. Therefore, we require:

$$\|\hat{x}(t_k+\tau)\| - \alpha \|\hat{x}(t_k+\tau)\| \left(e^{-\beta(h-\tau)} + \left(e^{K_f(h-\tau)} - e^{-\beta(h-\tau)}\right)\left(\frac{K_\delta}{K_f+\beta}\right)\right) > 0 \qquad (72)$$

$$\|\hat{x}(t_k+\tau)\| \left(1 - \left(e^{-\beta(h-\tau)} + \left(e^{K_f(t-(t_k+\tau))} - e^{-\beta(h-\tau)}\right)\left(\frac{K_\delta}{K_f+\beta}\right)\right)\right) > 0$$

$$\left(1 - \left(e^{-\beta(h-\tau)} + \left(e^{K_f(h-\tau)} - e^{-\beta(h-\tau)}\right)\left(\frac{K_\delta}{K_f+\beta}\right)\right)\right) > 0$$

■

5.2 Stability for a more general class of non-linear MB-NCS

We now extend the results to a nonlinear system whose plant dynamics are given by

$$\dot{x} = f(x) + g(x, u). \tag{73}$$

As above, we will follow the procedure used by Montestruque.

The model and controller are given by

$$\begin{aligned} \dot{\hat{x}} &= \hat{f}(\hat{x}) + \hat{g}(\hat{x}, u) \\ u &= k(\hat{x}) \end{aligned} \tag{74}$$

Substituting, we get:

$$\begin{aligned} \dot{x} &= f(x) + g(x, k(\hat{x})) = f(x) + m(x, \hat{x}) \\ \dot{\hat{x}} &= \hat{f}(\hat{x}) + \hat{g}(\hat{x}, k(\hat{x})) = \hat{f}(\hat{x}) + \hat{m}(\hat{x}, \hat{x}) \end{aligned} \tag{75}$$

Again, let us assume that the uncertainty between the plant and the model is of the additive type:

$$\begin{aligned} \hat{f}(\zeta) &= f(\zeta) + \delta_f(\zeta) \\ \hat{m}(\zeta) &= m(\zeta, \zeta) + \delta_m(\zeta) \end{aligned} \tag{76}$$

So, the error dynamics between the plant and the model are:

$$e = f(x) - f(\hat{x}) - \delta_f(\hat{x}) + m(x, \hat{x}) - m(\hat{x}, \hat{x}) - \delta_m(\hat{x}) \tag{77}$$

Assume also that the Lipschitz conditions hold:

$$\|f(x) - f(y)\| \le K_f \|x - y\| \tag{78}$$

$$\|m(x, s) - m(y, s)\| \le K_m(s) \|x - y\| \tag{79}$$

$$\|\delta_f(x) - \delta_f(y)\| \le K_{\delta_f} \|x - y\| \tag{80}$$

$$\|\delta_m(x) - \delta_m(y)\| \le K_{\delta_m} \|x - y\| \tag{81}$$

Define also $K_{m,\max} = \max_{s \in B_S}(K_m(s))$ for B_S, where B_S is a ball centered at the origin. Assume as well that the non-networked compensated plant model is exponentially stable when $\hat{x}(t_0) \in B_S$, $\hat{x}(t) \in B_\tau$, for $t \in [t_0, t_0 + \tau)$ with B_S and B_τ balls centered on the origin.

$$\|\hat{x}(t)\| \le \alpha \|\hat{x}(t_0)\| e^{-\beta(t-t_0)} \text{ with } \alpha, \beta > 0\,. \tag{82}$$

The following theorem states a sufficient condition for stability.

Theorem 8 *The nonlinear system with dynamics described above and that satisfies the Lipschitz conditions described and with exponentially stable com-*

Tomas Estrada and P.J. Antsaklis, Results on Continuous and Discrete Model-Based Networked Control Systems with Intermittent Feedback, Part I: Stability, ISIS Technical Report, University of Notre Dame, ISIS-2008-001, August 2008.

pensated plant model satisfying () is asymptotically stable if:

$$\left(1-\alpha\left(e^{-\beta(h-\tau)}+\left(e^{\left(K_f+K_{m,\max}\right)(h-\tau)}-e^{-\beta(h-\tau)}\right)\left(\frac{K_{\delta_f}+K_{\delta_m}}{K_f+K_{m,\max}+\beta}\right)\right)\right)>0 \tag{83}$$

Proof. Note that the error can be bounded as follows:

$$\|e(t)\| \le \int_{t_k+\tau}^{t}\left(\left(K_f+K_{m,\max}\right)\|x(s)-\hat{x}(s)\|+\left(K_{\delta_f}+K_{\delta_m}\right)\|\hat{x}(s)\|\right)ds\ ,\quad \forall t\in[t_k+\tau,t_{k+1})\ . \tag{84}$$

The rest of the proof is done as in the previous theorem. ■

6 Stability of MB-NCS with Intermittent Feedback and time-varying updates

Until now we have only considered the case where the parameters τ and h are constant. Let us now take a closer look at what happens when these parameters vary with time. The definitions for Lyapunov stability and mean square stability used throughout this section are the same as those in [18].

6.1 Lyapunov stability with bounded intervals

We shall first analyze the case where the parameters are time-varying, but their probability distributions are unknown. Let the plant, model, and controller have the same dynamics as described in Section 2. The following result describes the state response of the system. The derivation of this result is analogous to that for constant τ and h.

Proposition 9 *The system described above with initial conditions* $z = \begin{bmatrix} x(t_0) \\ 0 \end{bmatrix} = z_0$ *has the following response:*

$$z(t) = \begin{cases} e^{\Lambda_o(t-t_k)} \left(\prod_{j=1}^{k} M(j) \right) z_0 \ , \quad t \in [t_k, t_k + \tau) \\ e^{\Lambda_o(t-(t_k+\tau))} e^{\Lambda_c(\tau)} \left(\prod_{j=1}^{k} M(j) \right) z_0, \\ \qquad t \in [t_k + \tau, t_{k+1}) \end{cases}$$

where $M(j) = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o(h(j)-\tau(j))} e^{\Lambda_c(\tau(j))} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$, $\Lambda_o = \begin{bmatrix} A+BK & -BK \\ \tilde{A}+\tilde{B}K & \hat{A}-\tilde{B}K \end{bmatrix}$, $\Lambda_c = \begin{bmatrix} A+BK & -BK \\ 0 & 0 \end{bmatrix}$, $t_{k+1} - t_k = h(k)$, *and* $\tau(j) < h(j)$.

The proof is as follows.

Proof. The proof is similar to the corresponding development for constant h and τ. On the closed loop interval, the system response is:

$$z(t) = \begin{bmatrix} x(t) \\ e(t) \end{bmatrix} = e^{\Lambda_c(t-t_k)} \begin{bmatrix} x(t_k) \\ 0 \end{bmatrix} = e^{\Lambda_c(t-t_k)} z(t_k) \ , \ \forall t \in [t_k, t_k + \tau). \tag{85}$$

And on the open loop interval, the response is:

$$\begin{aligned} z(t) &= \begin{bmatrix} x(t) \\ e(t) \end{bmatrix} = e^{\Lambda_o(t-(t_k+\tau))} e^{\Lambda_c(t-t_k)} \begin{bmatrix} x(t_k) \\ 0 \end{bmatrix} = e^{\Lambda_o(t-(t_k+\tau))} e^{\Lambda_c(t-t_k)} z(t_k), \\ \forall t &\in [t_k+\tau, t_{k+1}) \end{aligned} \tag{86}$$

Now, note that at times $t_{k,}$, the error is reset to zero, which corresponds to pre-multiplying by $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$.

Using the above, we obtain

$$z(t_k) = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o(h(k)-\tau(k))} e^{\Lambda_c \tau(k)} z(t_{k-1}) \ .$$

Then, with initial conditions $t(0) = t_0, \ z(t_0) = z_0 = \begin{bmatrix} x_0 \\ 0 \end{bmatrix}$:

$$\begin{aligned} z(t) &= e^{\Lambda_c(t-t_k)} z(t_k) \\ &= e^{\Lambda_c(t-t_k)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o(h(k)-\tau(k))} e^{\Lambda_c \tau(k)} z(t_{k-1}) \\ &= e^{\Lambda_c(t-t_k)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o(h(k)-\tau(k))} e^{\Lambda_c \tau(k)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o(h(k-1)-\tau(k-1))} e^{\Lambda_c \tau(k-1)} z(t_{k-2}) \\ &= e^{\Lambda_c(t-t_k)} \left(\prod_{j=1}^{k} M(j) \right) z_0 \ , \ t \in [t_k, t_k+\tau), \end{aligned}$$

where

$$M(j) = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o (h(j)-\tau(j))} e^{\Lambda_c \tau(j)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$$

And similarly for the interval $t \in [t_k + \tau, t_{k+1})$. ■

We now present a condition for Lyapunov stability of this system.

Theorem 10 *The system described above is Lyapunov asymptotically stable for $h \in [h_{\min}, h_{\max}]$ and $\tau \in [\tau_{\min}, \tau_{\max}]$ (with $\tau_{\max} < h_{\min}$) if there exists a symmetric positive definite matrix X such that $Q = X - MXM^T$ is positive definite for all $h \in [h_{\min}, h_{\max}]$ and $\tau \in [\tau_{\min}, \tau_{\max}]$, where* $M = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o (h-\tau)} e^{\Lambda_c (\tau)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$.

Proof. Note that the output norm can be bounded by

$$\begin{aligned} & \left\| e^{\Lambda_o (t-(t_k+\tau))} e^{\Lambda_c (\tau)} \left(\prod_{j=1}^{k} M(j) \right) z_0 \right\| \\ & \leq \left\| e^{\Lambda_o (t-(t_k+\tau))} \right\| \left\| e^{\Lambda_c (\tau)} \right\| \left\| \prod_{j=1}^{k} M(j) \right\| \|z_0\| \\ & \leq e^{\bar{\sigma}(\Lambda_o) h_{\max} - \tau_{\min}} \left\| e^{\Lambda_c (\tau)} \right\| \left\| \prod_{j=1}^{k} M(j) \right\| \|z_0\| \end{aligned}$$

That is, since $e^{\Lambda_o (t-(t_k+\tau))}$ has finite growth and will grow for at most from $\tau_{\min}$ to $h_{\max}$, then convergence of the product of matrices $M(j)$ to zero ensures the stability of the system. Such convergence to zero is guaranteed by the existence of a symmetric positive definite matrix X in the Lyapunov equation. ■

6.2 Mean square stability of continuous MB-NCS with IF with i.i.d update times

Now, let us consider the case where τ is constant, but $h(k)$ are independent identically distributed with probability distribution $F(h)$. This corresponds to the situation where we might not know how frequently we can access the network, but when we do obtain access to it, we continue to have access to it for a fixed amount of time, so as to, for example, complete a given task or transmit a certain set of packets. We present a stability condition for this case:

Theorem 11 *The system described above with update times $h(j)$ independent identically distributed random variable with probability distribution $F(h)$ is globally mean square asymptotically stable around the solution $z = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ if $K = E\left[\left(e^{\bar{\sigma}(\Lambda_o)(h-\tau)}\right)^2\right] < \infty$ and the maximum singular value of the expected value $M^T M$, $\left\|E\left[M^T M\right]\right\| = \bar{\sigma}\left(E\left[M^T M\right]\right)$ is strictly less than one, where $M = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda_o(h-\tau)} e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$.*

Proof. Let us begin by evaluating the expectation of the squared norm of the system. Note that we are doing this for the interval $t \in [t_k, t_k + \tau_k)$, but the proof is the same for the interval $t \in [t_k + \tau_k, t_{k+1})$.

$$
\begin{aligned}
& E\left\| e^{\Lambda_o(t-(t_k+\tau_k))} e^{\Lambda_c(\tau_k)} \left(\prod_{j=1}^{k} M(j) \right) z_0 \right\|^2 \qquad (87) \\
= \;& E\left[z_0^T \left(\prod_{j=1}^{k} M(j) \right)^T \left(e^{\Lambda_o(t-(t_k+\tau_k))} e^{\Lambda_c(\tau_k)} \right)^T e^{\Lambda_o(t-(t_k+\tau_k))} e^{\Lambda_c(\tau_k)} \left(\prod_{j=1}^{k} M(j) \right) z_0 \right] \\
\leq \;& E\left[\bar{\sigma}\left(\left(e^{\Lambda_o(t-(t_k+\tau_k))} e^{\Lambda_c(\tau_k)} \right)^T e^{\Lambda_o(t-(t_k+\tau_k))} e^{\Lambda_c(\tau_k)} \right) z_0^T \left(\prod_{j=1}^{k} M(j) \right)^T \left(\prod_{j=1}^{k} M(j) \right) z_0 \right] \\
\leq \;& E\left[\left(e^{\bar{\sigma}(\Lambda_o)(h-\tau)(k+1)} \right)^2 z_0^T \left(\prod_{j=1}^{k} M(j) \right)^T \left(\prod_{j=1}^{k} M(j) \right) z_0 \right]
\end{aligned}
$$

Now that the expectation is all in terms of the update times, we can use the iid property of the update times and the assumption that K is bounded:

$$
\begin{aligned}
& E\left[\left(e^{\bar{\sigma}(\Lambda_o)(h-\tau)(k+1)} \right)^2 z_0^T \left(\prod_{j=1}^{k} M(j) \right)^T \left(\prod_{j=1}^{k} M(j) \right) z_0 \right] \qquad (88) \\
= \;& K\, z_0^T E\left[\left(\prod_{j=1}^{k} M(j) \right)^T M(k)^T M(k) \left(\prod_{j=1}^{k} M(j) \right) \right] z_0 \\
= \;& K\, z_0^T E\left[\left(\prod_{j=1}^{k} M(j) \right)^T E\left[M^T M \right] \left(\prod_{j=1}^{k} M(j) \right) \right] z_0 \\
\leq \;& K\, \bar{\sigma}\left(E\left[M^T M \right] \right) z_0^T E\left[\left(\prod_{j=1}^{k} M(j) \right)^T \left(\prod_{j=1}^{k} M(j) \right) \right] z_0
\end{aligned}
$$

We repeat the last three steps recursively to obtain

$$\begin{aligned} & E\left\| e^{\Lambda_c(t-t_k)}\left(\prod_{j=1}^{k} M(j)\right) z_0\right\|^2 \\ \leq \ & K\left(\bar{\sigma}\left(E\left[M^T M\right]\right)\right)^k z_0^T z_0 \end{aligned}$$

From here, we can see that if $\left\|E\left[M^T M\right]\right\| =,\bar{\sigma}\left(E\left[M^T M\right]\right) < 1$, then the limit of the expectation as time goes to infinity approaches zero. ∎

6.3 Mean square stability of continuous MB-NCS with IF with Markov chain-driven update times

We now consider the situation where the parameter h is driven by a Markov chain and provide a stability condition.

Theorem 12 *The system described above with update times $h(k) = h_{\omega_k} \neq \infty$ driven by a finite state Markov chain $\{\omega_k\}$ with state space $\{1, 2, ..., N\}$ and transition probability matrix Γ with elements $p_{i,j}$ is globally mean square asymptotically stable around the solution $z = \left[x^T\, e^T\right]^T = \mathbf{0}$ if there exist positive definite matrices $P(1), P(2), ..., P(N)$ such that*

$$\left(\sum_{j=1}^{N} p_{i,j}\left(H(i)^T P(j) H(i)\right) - P(i)\right) < 0 \ \forall i, j \in 1, ..., N$$

with $H(i) = e^{\Lambda_o(h_i-\tau)} e^{\Lambda_c(\tau)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}.$

The proof follows that in [18] for the case of instantaneous feedback.

7 Stability of Discrete-Time Plants using Model-Based Control with Intermittent Feedback

7.1 Problem Formulation

The basic setup for discrete-time MB-NCS with intermittent feedback is essentially the same as that for continuous time; see also [6]. We make the same assumptions as in [15] for the instantaneous feedback case, where both the sensor and actuator sides are synchronized and updates occur at the same instants of time.

Consider the control of a discrete linear plant where the state sensor is connected to a linear controller/actuator via a network. In this case, the controller uses an explicit model of the plant that approximates the plant dynamics and makes possible the stabilization of the plant even under slow network conditions.

In dealing with intermittent feedback, we have two key time parameters: how frequently we want to close the loop, which we shall denote by h, and how long we wish the loop to remain closed, which we shall denote by τ. Naturally, in the more general cases both h and τ can be time-varying. Unlike the continuous time formulation, h and τ are both integers here, as they

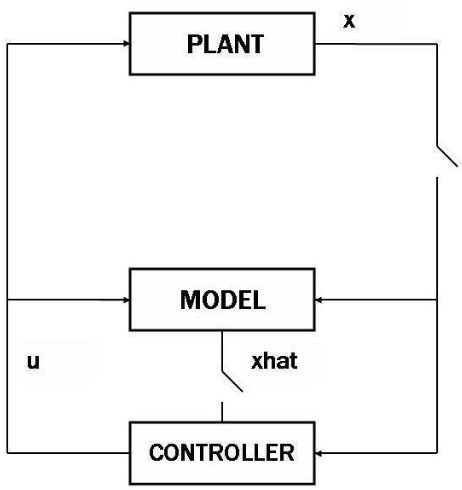

Figure 17: Basic MB-NCS architecture

represent the number of ticks of the clock in the corresponding interval.

We consider then a system such that the loop is closed periodically, every h ticks of the clock, and where each time the loop is closed, it remains so for a time of τ ticks of the clock. The loop is closed at times n_k, for $k = 1, 2, ...$. The system will be operating in closed loop mode for the intervals $[n_k, n_k+\tau)$ and in open loop for the intervals $[n_k + \tau, n_{k+1})$, with $n_{k+1} - n_k = h$. When the loop is closed, the control decision is based directly on the information of the state of the plant, but we will keep track of the error nonetheless.

As mentioned in the introduction, it is important to note that the parameters τ and h are different from the sampling time of the digital plant, since they are tailored after the demands of use of the network, not by the internal clock of the plant. It is also important to keep in mind that even when the loop is "closed", information is being sent at discrete intervals, the duration of which is determined by the internal clock of the plant.

The plant is given by $x(n+1) = Ax(n) + Bu(n)$, the plant model by $\hat{x}(n+1) = \hat{A}\hat{x}(n) + \hat{B}u(n)$, and the controller by $u(n) = K\hat{x}(n)$. The state error is defined as $e(n) = x(n) - \hat{x}(n)$ and represents the difference between plant state and the model state. The modeling error matrices $\tilde{A} = A - \hat{A}$ and $\tilde{B} = B - \hat{B}$ represent the plant and the model. We also define the vector $z = [x^T \ e^T]^T$.

In the next section we will derive a complete description of the response of the system as well as a necessary and sufficient condition for stability.

7.2 State Response of the System and Stability Condition

We will now proceed to derive the response to prove the above proposition. The approach is similar to that we used in [6] for the continuous time case. To this effect, let us separately investigate what happens when the system is operating under closed and open loop conditions.

7.2.1 State response of the system

During the open loop case, that is, when $n \in [n_k + \tau, n_{k+1})$, we have that

$$u(n) = K\hat{x}(n) \tag{89}$$

so

$$\begin{bmatrix} x\left(n+1\right) \\ \hat{x}(n+1) \end{bmatrix} = \begin{bmatrix} A & BK \\ 0 & \hat{A}+\hat{B}K \end{bmatrix} \begin{bmatrix} x(n) \\ \hat{x}(n) \end{bmatrix} \tag{90}$$

with initial conditions $\hat{x}(n_k+\tau) = x\left(n_k+\tau\right)$.

Rewriting in terms of x and e, that is, of the vector z :

$$z(n+1) = \begin{bmatrix} x\left(n+1\right) \\ e(n+1) \end{bmatrix} = \tag{91}$$

$$\begin{bmatrix} A+BK & -BK \\ \tilde{A}+\tilde{B}K & \hat{A}-\tilde{B}K \end{bmatrix} \begin{bmatrix} x(n) \\ e(n) \end{bmatrix}$$

$$z(n_k+\tau) = \begin{bmatrix} x(n_k+\tau) \\ e(n_k+\tau) \end{bmatrix} = \begin{bmatrix} x(n_k+\tau^-) \\ 0 \end{bmatrix},$$

$$\forall n \in [n_k+\tau, n_{k+1}) \tag{92}$$

Thus, we have

$$z(n+1) = \Lambda_{Do} z(n), \quad \text{where} \quad \Lambda_{Do} = \begin{bmatrix} A+BK & -BK \\ \tilde{A}+\tilde{B}K & \hat{A}-\tilde{B}K \end{bmatrix}, \tag{93}$$

$$\forall n \in [n_k+\tau, n_{k+1})$$

.

The closed loop case is a simplified version of the case above, as the difference resides in the fact that the error is always zero. Thus, for $n \in$

Tomas Estrada and P.J. Antsaklis, Results on Continuous and Discrete Model-Based Networked Control Systems with Intermittent Feedback, Part I: Stability, ISIS Technical Report, University of Notre Dame, ISIS-2008-001, August 2008.

$[n_k, n_k + \tau)$, we have

$$z(n+1) = \Lambda_{Dc} z(n), \quad \text{where} \quad \Lambda_{Dc} = \begin{bmatrix} A+BK & -BK \\ 0 & 0 \end{bmatrix}, \qquad (94)$$

$$n \in [n_k, n_k + \tau)$$

. This should be clear in that the error is always zero, while the state progresses in the same way as before.

From this, it should be quite clear that given an initial condition $z(n = 0) = z_0$, then after a certain time $n \in [0, \tau)$, the solution of the trajectory of the vector is given by

$$z(n) = \Lambda_{Dc}^{n} z_0 \, , \; n \in [0, \tau). \qquad (95)$$

In particular, at time τ, $z(\tau) = \Lambda_{Dc}^{\tau} z_0$.

Once the loop is opened, the open loop behavior takes over, so that

$$z(n) = \Lambda_{Do}^{(n-\tau)} z(\tau) = \Lambda_{Do}^{(n-\tau)} \Lambda_{Dc}^{\tau} z_0 \, , \; n \in [\tau, n_1). \qquad (96)$$

In particular, when the time comes to close the loop again, that is, after time h, then $z\,(n_1) = \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau} z_0$.

Notice, however, that at this instant when we close the loop again, we are also resetting the error to zero, so that we must pre-multiply by $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$

before we analyze the closed loop trajectory for the next cycle. Because we wish to always start with an error that is set to zero, we should actually multiply by $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ at the beginning.

So then, after k cycles, going through this analysis yields a solution.

$$
\begin{aligned}
z\left(t_k\right) &= \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}\right)^k z_0 \\
&= \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Sigma \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}\right)^k z_0 \,, \qquad (97)
\end{aligned}
$$

where $\Sigma = \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau}$.

The final step is to consider the last (partial) cycle that the system goes through, that is, the time $n \in [n_k, n_{k+1})$. If the system is in closed loop, that is, $n \in [n_k, n_k + \tau)$, then the solution can be achieved merely by pre-multiplying $z\left(n_k\right)$ by $\Lambda_{Dc}^{(n-n_k)}$. In the case of the system being in open loop, that is, $n \in [n_k + \tau, n_{k+1})$, then clearly we must pre-multiply by $\Lambda_{Do}^{(n-(n_k+\tau))} \Lambda_{Dc}^{\tau}$.

The results can thus be summarized in the following proposition.

Proposition 13 *The system described by (93) and (94) with initial condi-*

tions $z(n_0) = \begin{bmatrix} x(n_0) \\ 0 \end{bmatrix} = z_0$ *has the following response:*

$$z(n) = \begin{cases} \Lambda_{Dc}^{(n-n_k)} \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Sigma \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k z_0 \ , \\ \qquad n \in [n_k, n_k + \tau) \\ \Lambda_{Do}^{(n-(n_k+\tau))} \Lambda_{Dc}^{\tau} \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Sigma \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k z_0 \ , \\ \qquad n \in [n_k + \tau, n_{k+1}) \end{cases} \tag{98}$$

where $\Sigma = \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau}$, $\Lambda_{Do} = \begin{bmatrix} A + BK & -BK \\ \tilde{A} + \tilde{B}K & \hat{A} - \tilde{B}K \end{bmatrix}$, $\Lambda_{Dc} = \begin{bmatrix} A + BK & -BK \\ 0 & 0 \end{bmatrix}$, *and* $n_{k+1} - n_k = h$.

7.2.2 Stability Condition

We will present a necessary and sufficient condition for the stability of the system.

Theorem 14 *The system described by (93) and (94) is globally exponentially stable around the solution* $z = \begin{bmatrix} x \\ e \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ *if and only if the eigenvalues of* $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Sigma \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ *are strictly inside the unit circle, where*

$\Sigma = \Lambda_{Do}^{(h-\tau)}\Lambda_{Dc}^{\tau}$.

Proof. Sufficiency. Taking the norm of the solution described as in Proposition #1:

$$\|z(n)\| = \left\| \Lambda_{Dc}^{(n-n_k)} \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(h-\tau)}\Lambda_{Dc}^{\tau} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k z_0 \right\| \tag{99}$$

$$\leq \left\| \Lambda_{Dc}^{(n-n_k)} \right\| \left\| \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(h-\tau)}\Lambda_{Dc}^{\tau} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k \right\|$$

$$\|z_0\|$$

Notice we are only doing this part for the case when $n \in [n_k, n_k + \tau)$, but the process is exactly the same for the intervals where $n \in (n_k + \tau, n_k + 1)$. Analyzing the first term on the right hand side:

$$\left\| \Lambda_{Dc}^{(n-n_k)} \right\| \leq \left(\bar{\sigma}\left(\Lambda_{Dc}\right)\right)^{n-n_k} \leq \left(\bar{\sigma}\left(\Lambda_{Dc}\right)\right)^{\tau} = K_1 \tag{100}$$

where $\bar{\sigma}\left(\Lambda_{Dc}\right)$ is the largest singular value of Λ_{Dc}. In general this term can always be bounded as the time difference $n - n_k$ is always smaller than τ. That is, even when Λ_{Dc} has eigenvalues with positive real part, $\left\| \Lambda_{Dc}^{(n-n_k)} \right\|$ can only grow a certain amount. This growth is completely independent of k.

We now study the term $\left\| \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k \right\|$. It is clear taht this term will be bounded if and only if the eigenvalues of $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ lie inside the unit circle:

$$\left\| \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k \right\| \leq K_2 e^{-\alpha_1 k} \tag{101}$$

with $K_2, \alpha_1 > 0$.

Since k is a function of time we can bounded the right term of the previous inequality in terms of t :

$$K_2 e^{-\alpha_1 k} < K_2 e^{-\alpha_1 \frac{n-1}{h}} = K_2 e^{\frac{\alpha_1}{h}} e^{-\frac{\alpha_1}{h} n} = K_3 e^{-\alpha n} \tag{102}$$

with $K_{3,}\alpha > 0$.

So from the above, we conclude that:

$$\begin{aligned} &\left\| z(n) \right\| \\ &= \left\| \Lambda_{Dc}^{(n-n_k)} \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k z_0 \right\| \\ &\leq K_1 K_3 e^{-\alpha n} \left\| z_0 \right\| . \end{aligned} \tag{103}$$

Necessity. We will now provide the necessity part of the theorem. We will do this by contradiction. Assume the system is stable and that $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ has at least one eigenvalue outside the unit circle. Let us define $\Sigma(h) = \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau}$. Since the system is stable, a periodic sample of the response should converge to zero with time. We will take the samples at times n_{k+1}, that is, just before the loop is closed again. We will concentrate on a specific term: the state of the plant $x(n_{k+1})$, which is the first element of $z(n_{k+1})$. We will call $x(n_{k+1})$, $\xi(k)$.

Now assume $\Sigma(\eta)$ has the following form:

$$\Sigma(\eta) = \begin{bmatrix} W(\eta) & X(\eta) \\ Y(\eta) & Z(\eta) \end{bmatrix}.$$

Then we can express the solution $z(n)$ as:

$$\begin{aligned} &\Lambda_{Dc}^{(n-n_k)} \left(\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Sigma(h) \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right)^k z_0 \qquad (104) \\ &= \begin{bmatrix} W(n-n_k) & X(n-n_k) \\ Y(n-n_k) & Z(n-n_k) \end{bmatrix} \begin{bmatrix} (W(h))^k & 0 \\ 0 & 0 \end{bmatrix} z_0 \\ &= \begin{bmatrix} W(n-n_k)(W(h))^k & 0 \\ Y(n-n_k)(W(h))^k & 0 \end{bmatrix} z_0 . \end{aligned}$$

Now, the values of the solution at times n_{k+1}^{-}, that is, just before the loop is closed again, are

$$z\left(n_{k+1}\right) = \begin{bmatrix} W(h)\left(W(h)\right)^{k} & 0 \\ Y(h)\left(W(h)\right)^{k} & 0 \end{bmatrix} z_0$$
$$= \begin{bmatrix} \left(W(h)\right)^{k+1} & 0 \\ Y(h)\left(W(h)\right)^{k} & 0 \end{bmatrix} z_0 \tag{105}$$

We also know that $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$ has at least one eigenvalue outside the unit circle, which means that those unstable eigenvalues must be in $W(h)$. This means that the first element of $z\left(n_{k+1}\right)$, which we call $\xi\left(k+1\right)$, will in general grow with k. In other words we cannot ensure $\xi\left(k+1\right)$ will converge to zero for general initial condition x_0.

$$\begin{aligned} \left\|x\left(n_{k+1}\right)\right\| &= \left\|\xi\left(k+1\right)\right\| = \left\|\left(W\left(h\right)\right)^{k+1} x_0\right\| \rightarrow \infty \\ \text{as } k &\rightarrow \infty\,, \end{aligned} \tag{106}$$

which clearly means the system cannot be stable. Thus, we have a contradiction. ■

Tomas Estrada and P.J. Antsaklis, Results on Continuous and Discrete Model-Based Networked Control
Systems with Intermittent Feedback, Part I: Stability, ISIS Technical Report, University of Notre Dame,
ISIS-2008-001, August 2008.

8 Stability of Discrete MB-NCS with Intermittent Feedback (State Observer case)

When the full information of the state is not available, we use a state observer to estimate its value. The corresponding architecture is showing in Figure 18 and is the same as that developed for continuous plants in [6].

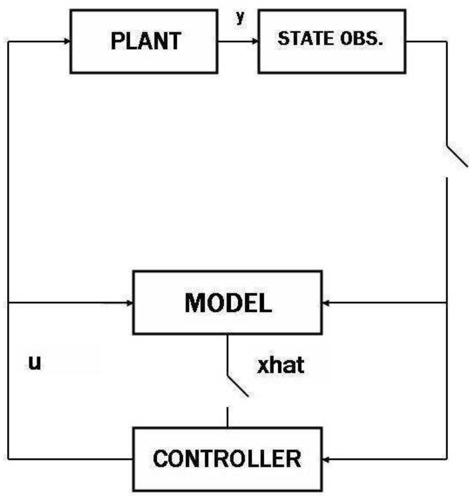

Figure 18: Model-based networked control system with state observer

The equations governing the behavior of the system can be summarized as follows:

Plant: $x(n+1) = Ax(n) + Bu(n)$,

$y(n) = Cx(n) + Du(n)$

Model: $\hat{x}(n+1) = \hat{A}\hat{x}(n) + \hat{B}u(n)$,

$y(n) = \hat{C}\hat{x}(n) + \hat{D}u(n)$

Controller: $u(n) = K\hat{x}(n)$

Observer: $\bar{x}\,(n+1) = (\hat{A} - L\hat{C})\bar{x}\,(n) + \begin{bmatrix} \hat{B} - L\hat{D} & L \end{bmatrix} \begin{bmatrix} u\,(n) \\ y\,(n) \end{bmatrix}$

Controller model state: $\hat{x}$

Observer's estimate: $\bar{x}$

Error matrices: $\tilde{A} = A - \hat{A},\ \tilde{B} = B - \hat{B},$

$\tilde{C} = C - \hat{C},\ \tilde{D} = D - \hat{D}$

We will present the full description of the state response of the system as well as a necessary and sufficient condition for stability. As before, τ and h are integers.

8.1 State Response of the system (State Observer case)

The following proposition details the state response of the system for the case with state observer. The derivation of this result is similar to that of the full information case from the previous section. We will not include it here because of space limitations.

Proposition 15 *The system described above and with initial condition* $z\,(n_0) = \begin{bmatrix} x\,(n_0) \\ \bar{x}(n_0) \\ 0 \end{bmatrix} = z_0$ *has the following state response:*

$$z(n) = \begin{cases} \Lambda_{Dc}^{(n-n_k)} \left(\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} \Sigma \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} \right)^k z_0, \\ n \in [n_k, n_k + \tau) \\ \Lambda_{Do}^{(n-(n_k+\tau))} \Lambda_{Dc}^{\tau} \\ \left(\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} \Sigma \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} \right)^k z_0, \\ n \in [n_k + \tau, n_{k+1}) \end{cases} \tag{107}$$

where $\Sigma = \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau}$, *and*

$$\Lambda_{Do} = \begin{bmatrix} A & BK & -BK \\ LC & \hat{A} - L\hat{C} + \hat{B}K + L\tilde{D}K & -\hat{B}K - L\tilde{D}K \\ LC & L\tilde{D}K - L\hat{C} & A - L\tilde{D}K \end{bmatrix},$$

$$\Lambda_{Dc} = \begin{bmatrix} A & BK & -BK \\ LC & \hat{A} - L\hat{C} + \hat{B}K + L\tilde{D}K & -\hat{B}K - L\tilde{D}K \\ 0 & 0 & 0 \end{bmatrix},$$

and $n_{k+1} - n_k = h$..

8.2 Stability condition (State Observer case)

We now state the following theorem characterizing the necessary and sufficient conditions for the system described in the previous section to have

Tomas Estrada and P.J. Antsaklis, Results on Continuous and Discrete Model-Based Networked Control Systems with Intermittent Feedback, Part I: Stability, ISIS Technical Report, University of Notre Dame, ISIS-2008-001, August 2008.

globally exponential stability around the solution $z = 0$.

Theorem 16 *The system described above is globally exponentially stable around the solution* $z = \begin{bmatrix} x \\ \bar{x} \\ e \end{bmatrix} = \mathbf{0}$ *if and only if the eigenvalues of* $\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix} \Sigma \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}$ *are strictly inside the unit circle, where* $\Sigma = \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau}$, *and* $\Lambda_{Do}, \Lambda_{Dc}$ *as before.*

The proof is similar to that of the case with full information and will be omitted for reasons of space.

9 Stability of discrete time plants with time-varying updates

Until now we have only considered the case where the parameters τ and h are constant. Let us now take a closer look at what happens when these parameters vary with time. The definitions for Lyapunov stability and mean square stability used throughout this section are the same as those in [14].

9.1 Lyapunov stability with bounded intervals

We shall first analyze the case where the parameters are time-varying, but their probability distributions are unknown. The following result describes the state response of the system. The derivation of this result is analogous to that for constant τ and h and is included for the sake of completeness.

Proposition 17 *The system described in (93) and (94) with initial conditions* $z = \begin{bmatrix} x(n_0) \\ 0 \end{bmatrix} = z_0$ *has the following response:*

$$z(n) = \begin{cases} \Lambda_{Dc}^{(n-n_k)} \left(\prod_{j=1}^{k} M(j) \right) z_0 \quad , \quad n \in [n_k, n_k + \tau_k) \\ \Lambda_{Do}^{(n-(n_k+\tau))} \Lambda_{Dc}^{\tau_k} \left(\prod_{j=1}^{k} M(j) \right) z_0 \quad , \quad n \in [n_k + \tau_k, n_{k+1}) \end{cases}$$

where $M(j) = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(h-\tau)(j)} \Lambda_{Dc}^{\tau(j)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$, $\Lambda_{Do} = \begin{bmatrix} A + BK & -BK \\ \tilde{A} + \tilde{B}K & \hat{A} - \tilde{B}K \end{bmatrix}$, $\Lambda_{Dc} = \begin{bmatrix} A + BK & -BK \\ 0 & 0 \end{bmatrix}$, $n_{k+1} - n_k = h(k)$, *and* $\tau(j) < h(j)$.

Proof. The proof is similar to the corresponding development for constant h and τ. On the closed loop interval, the system response is:

$$z(n) = \begin{bmatrix} x(n) \\ e(n) \end{bmatrix} = \Lambda_{Dc}^{(n-n_k)} \begin{bmatrix} x(n_k) \\ 0 \end{bmatrix} = \Lambda_{Dc}^{(n-n_k)} z(n_k) \ , \ \forall n \in [n_k, n_k + \tau). \tag{108}$$

And on the open loop interval, the response is:

$$\begin{aligned} z(n) &= \begin{bmatrix} x(n) \\ e(n) \end{bmatrix} = \Lambda_{Do}^{(n-(n_k+\tau))}\Lambda_{Dc}^{(n-n_k)} \begin{bmatrix} x(n_k) \\ 0 \end{bmatrix} = \Lambda_{Do}^{(n-(n_k+\tau))}\Lambda_{Dc}^{(n-n_k)} z(n_k) \\ \forall n &\in [n_k+\tau, n_{k+1}) \end{aligned} \tag{109}$$

Now, note that at times n_k,, the error is reset to zero, which corresponds to pre-multiplying by $\begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$.

Using the above, we obtain

$$z(t_k) = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(h-\tau)(j)}\Lambda_{Dc}^{\tau(j)} z(n_{k-1}) \ .$$

Then, with initial conditions $n(0) = t_0, \ z(n_0) = z_0 = \begin{bmatrix} x_0 \\ 0 \end{bmatrix}$:

$$\begin{aligned} z(n) &= \Lambda_{Dc}^{(n-n_k)} z(n_k) \\ &= \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(n-(n_k+\tau_k))}\Lambda_{Dc}^{(n-n_k)} z(n_{k-1}) \\ &= \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(n-(n_k+\tau_k))}\Lambda_{Dc}^{(n-n_k)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(n-(n_{k-1}+\tau_{k-1}))}\Lambda_{Dc}^{(n-n_{k-1})} z(n_{k-2}) \\ &= \Lambda_{Dc}^{(n-n_k)} \left(\prod_{j=1}^{k} M(j) \right) z_0 \ , \ n \in [n_k, n_k+\tau_k) \ , \end{aligned}$$

where

$$M(j) = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(h-\tau)(j)} \Lambda_{Dc}^{\tau(j)} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$$

And similarly for the interval $n \in [n_k + \tau, n_{k+1})$. ■

We now present a condition for Lyapunov stability of this system.

Theorem 18 *The system described in (93) and (94) is Lyapunov asymptotically stable for $h \in [h_{\min}, h_{\max}]$ and $\tau \in [\tau_{\min}, \tau_{\max}]$ (with $\tau_{\max} < h_{\min}$) if there exists a symmetric positive definite matrix X such that $Q = X - MXM^T$ is positive definite for all $h \in [h_{\min}, h_{\max}]$ and $\tau \in [\tau_{\min}, \tau_{\max}]$, where* $M = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$.

Proof. Note that the output norm can be bounded by

$$\begin{aligned} &\left\| \Lambda_{Do}^{(n-(n_k+\tau))} \Lambda_{Dc}^{\tau} \left(\prod_{j=1}^{k} M(j) \right) z_0 \right\| \\ &\leq \left\| \Lambda_{Do}^{(n-(n_k+\tau))} \Lambda_{Dc}^{\tau} \right\| \left\| \Lambda_{Dc}^{\tau} \right\| \left\| \prod_{j=1}^{k} M(j) \right\| \|z_0\| \\ &\leq \bar{\sigma}\left(\Lambda_{Do}^{h_{\max}-\tau_{\min}} \right) \left\| \Lambda_{Dc}^{\tau} \right\| \left\| \prod_{j=1}^{k} M(j) \right\| \|z_0\| \end{aligned}$$

That is, since $\Lambda_{Do}^{(n-(n_k+\tau))}$ has finite growth and will grow for at most from $\tau_{\min}$ to $h_{\max}$, then convergence of the product of matrices $M(j)$ to zero ensures the stability of the system. Such convergence to zero is guaranteed by the existence of a symmetric positive definite matrix X in the Lyapunov equation. ■

9.2 Mean square stability of discrete MB-NCS with IF with i.i.d update times

Now, let us consider the case where τ is constant, but $h(k)$ are independent identically distributed with probability distribution $F(h)$. This corresponds to the situation where we might not know how frequently we can access the network, but when we do obtain access to it, we continue to have access to it for a fixed amount of time, so as to, for example, complete a given task or transmit a certain set of packets. We present a stability condition for this case:

Theorem 19 *The system described in (93) and (94) with update times $h(j)$ independent identically distributed random variable with probability distribution $F(h)$ is globally mean square asymptotically stable around the solution $z = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ if $K = E\left[\left(\Lambda_{Do}^{(h-\tau)}\right)^2\right] < \infty$ and the maximum singular value of the expected value $M^T M$, $\left\|E\left[M^T M\right]\right\| = \bar{\sigma}\left(E\left[M^T M\right]\right)$ is strictly less than one, where $M = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Lambda_{Do}^{(h-\tau)} \Lambda_{Dc}^{\tau} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$.*

The proof is similar to that for the continuous case.

Tomas Estrada and P.J. Antsaklis, Results on Continuous and Discrete Model-Based Networked Control Systems with Intermittent Feedback, Part I: Stability, ISIS Technical Report, University of Notre Dame, ISIS-2008-001, August 2008.

9.3 Mean square stability of discrete MB-NCS with IF with Markov chain-driven update times

We now consider the situation where the parameter h is driven by a Markov chain and provide a stability condition.

Theorem 20 *The system described in (93) and (94) with update times* $h(k) = h_{\omega_k} \neq \infty$ *driven by a finite state Markov chain* $\{\omega_k\}$ *with state space* $\{1, 2, ..., N\}$ *and transition probability matrix* Γ *with elements* $p_{i,j}$ *is globally mean square asymptotically stable around the solution* $z = \left[x^T \; e^T\right]^T = \mathbf{0}$ *if there exist positive definite matrices* $P(1), \; P(2), \; ..., \; P(N)$ *such that* $\left(\sum_{j=1}^{N} p_{i,j} \left(H(i)^T P(j) H(i)\right) - P(i)\right)$ $0 \; \forall i, j \in 1, ..., N$ *with* $H(i) = \Lambda_{Do}^{(h_i - \tau)} \Lambda_{Dc}^{\tau} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$

Once again, the proof follows that of the continuous case.

10 Conclusions and future work

We have introduced the concept of model-based control with intermittent feedback. We proposed a basic architecture, focusing first on the continuous time case, and derived a complete description of the output of the system, as well as necessary and sufficient conditions for stability. We have then extended our results to cases with state observers, delays, and nonlinear plants. Finally, we investigated the situation where the update times τ and h are time-varying, first addressing the case where they have upper and lower bounds, then moving on to the case where their distributions are i.i.d or driven by a Markov chain, providing stability conditions in each case. We also obtained an analogous set of results for the discrete-time case.

The focus of the present report was on stability, but the area of performance of networked control systems, both under the model-based architecture and otherwise, remains a relatively unexplored ground for research. In future work, we expect to provide results on performance of model-based networked control systems with intermittent feedback, and will consider other issues, such as robustness, tracking, filtering, and improving control as time elapses (that is, to use intermittent feedback to improve performance, by updating the model during the times when the system is running closed loop, with the aim of enabling the user to run the system closed loop for progressively shorter intervals), as well. Results on some of these issues will be provided in the next part of the technical report.

References

[1] P. Antsaklis and A. Michel, *Linear Systems*, 1st edition, McGraw-Hill, New York, 1997.

[2] J. Baillieul, P. Antsaklis. "Control and communication challenges in networked real-time systems," Proceedings of the IEEE, v 95, n 1, Jan. 2007, p 9-28.

[3] B. Azimi-Sadjadi, "Stabilityof Networked Control Systems in the Presence of Packet Losses," Proceedings of the 42nd Conference of Decision and Control, December 2003.

[4] M.S. Branicky, S. Phillips, and W. Zhang, "Scheduling and feedback co-design for networked control systems," Proceedings of the 41st Conference on Decision and Control, December 2002.

[5] D. Delchamps, "Stabilizing a Linear System with Quantized State Feedback," IEEE Transactions on Automatic Control, vol. 35, no. 8, pp. 916-924.

[6] T. Estrada, H. Lin, and P.J. Antsaklis, "Model-Based Control with Intermittent Feedback", Proceedings of the 14th Mediterranean Conference on Control and Automation, 2006.

[7] T. Estrada, and P.J. Antsaklis, "Control with Intermittent Communication: A New Look at Feedback Control", Workshop on Networked Distributed Systems for Intelligent Sensing and Control, Kalamata, Greece, 2007.

[8] T. Estrada, and P.J. Antsaklis, "Stability of Model-Based Networked Control Systems with Intermittent Feedback", Proceedings of the 15th IFAC World Congress, Seoul, Korea, 2008.

[9] T. Estrada, P.J. Antsaklis, "Stability of Discrete-Time Plants using Model-Based Control with Intermittent Feedback", Proceedings of the

16th Mediterranean Conference on Control and Automation, Ajaccio, France, 2008.

[10] E Fridman, A Seuret, JP Richard, "Robust sampled-data stabilization of linear systems: an input delay approach", Automatica, 2004.

[11] J. Hespanha, P. Naghshtabrizi, Yonggang Xu, "A survey of recent results in networked control systems", Proceedings of the IEEE, v 95, n 1, Jan. 2007, p 138-62.

[12] D. Hristu-Varsakelis, "Feedback Control Systems as Users of a Shared Network: Communication Sequences that Guarantee Stability," Proceedings of the 40th Conference on Decision and Control, December 2001, pp. 3631-36.

[13] M. Kim, et al, "Controlling chemical turbulence by global delayed feedback:Pattern formation in catalytic co oxidation" on pt(110), Science, vol. 292, no. 5520, 2001.

[14] K. Koay and G. Bugmann, "Compensating intermittent delayed visual feedback in robot navigation", Proceedings of the IEEE Conference on Decision and Control Including The Symposium on Adaptive Processes, 2004.

[15] D. Liberzon and R. Brockett, "Quantized Feedback Stabilization of Linear Systems," IEEE Transactions on Automatic Control, Vol 45, no 7, pp 1279-89, July 2000.

[16] L.A. Montestruque, "A Dissertation: Model-Based Networked Control Systems", University of Notre Dame, Notre Dame IN, USA, September 2004.

[17] L.A. Montestruque and P.J. Antsaklis, "Model-Based Networked Control Systems: Necessary and Sufficient Conditions for Stability", 10th Mediterranean Conference on Control and Automation, July 2002.

[18] L.A. Montestruque and P.J. Antsaklis, "State and Output Feedback Control in Model-Based Networked Control Systems", 41st IEEE Conference on Decision and Control, December 2002.

[19] L.A. Montestruque and P.J. Antsaklis, "On the model-based control of networked systems," Automatica, Vol. 39, pp 1837-1843.

[20] L.A. Montestruque and P.J. Antsaklis, "Networked Control Systems: A model-based approach," 12th Mediterranean Conference on Control and Automation, June 2004.

[21] E. Ronco and D. J. Hill, "Open-loop intermittent feedback optimal predictive control:a human movement control model", NIPS99, 1999.

[22] G. Nair and R. Evans, "Communication-Limited Stabilization of Linear Systems," Proceedings of the Conference on Decision and Control, 2000, pp. 1005-1010.

[23] E Fridman, A Seuret, JP Richard, "Robust sampled-data stabilization of linear systems: an input delay approach", Automatica, 2004.

[24] B. H. Salzberg, A. J. Wheeler, L. T. Devar, and B. L. Hopkins, ”The effect of intermittent feedback and intermittent contingent access to play on printing of kindergarten children”, J Anal Behav, vol. 4, no. 3, 1971.

[25] Richard A. Schmidt, *Motor Control and Learning - A Behaviorial Emphasis*. Human Kinetics, 4th ed., 2005.

[26] J. Took, D. Tilbury, and N. Soparkar, ”Trading Computation for Bandwidth: Reducing Computation in Distributed Control Systems using State Estimators,” IEEE Transactions on Control Systems Technology, July 2002, Vol 10, No 4, pp 503-518.

[27] G. Walsh, H. Ye, and L. Bushnell, ”Stability Analysis of Networked Control Systems,” Proceedings of American Control Conference, June 1999.