# Output Feedback Model-Based Control of Uncertain Discrete-Time Systems with Network Induced Delays.

Eloy Garcia and Panos J. Antsaklis, *Fellow, IEEE*

***Abstract*—A new architecture for model-based control of unstable and uncertain systems with network induced delays, and for set-point tracking over networks, is presented in this paper. This setup provides better performance than similar approaches in terms of steady-state tracking error and reduction of network traffic by transmitting measurement updates only when necessary. The results in this paper also extend previous work using the Model-Based Networked Control Systems (MB-NCS) approach to consider the output feedback case and to consider models and uncertain systems which do not necessarily have the same dimension; that is, both the parameters and the order of the system are unknown. The Model-Based Event-Triggered (MB-ET) framework presented in this paper is used for stabilization and tracking of piece-wise constant signals and it is extended in two directions; first, to consider a more general two channel networked system and second, to address the reference input tracking problem in the presence of network induced delays.**

## I. INTRODUCTION

IN Networked Control Systems (NCS) the system to be controlled and the different components such as actuators, controllers, and sensors are spatially distributed and communication between these components is achieved through the use of a digital communication network [1]. NCS offer many advantages such as cost efficiency and improved functionality [2]-[3], but new problems arise by using this control implementation compared to the classical wired control systems.

One of the main problems in NCS which is studied in this paper is the design of control schemes considering lack of feedback measurements for possibly long intervals of time. Model uncertainties are important to be considered under this situation. One of the properties of a classical closed loop system with continuous feedback is that the appropriate design of closed loop controllers reduces sensitivity to model uncertainties. Naturally, this property is lost as feedback measurements are no longer received at the controller node. An important framework that considers model uncertainties and that is able to reduce sampling rate is called Model-Based Networked Control Systems (MB-NCS) [4]-[6]. Results obtained in relation to MB-NCS deal primarily with stability issues in different cases such as continuous-time and discrete-time plants with periodic update intervals [5] and with random update intervals [6].

Both authors are with the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556 USA (e-mail: egarcia7@nd.edu, antsaklis.1@nd.edu). The support of the National Science Foundation under Grant No. CNS-1035655 is gratefully acknowledged.

Similar model-based approaches have been considered by different authors, but many of them assume that the model and system parameters are exactly the same [7]-[9]. Since the absence of feedback measurements produces large errors between the model and the real system response, even small uncertainties may result in undesirable system behavior if they are not considered in the controller design stage.

Previous work in MB-NCS assumes that the model is of the same dimension as the real plant [4]-[6], [10]-[11]. The new MB-NCS architecture described in the present paper not only generalizes to the case of output feedback but it also considers different types of uncertainties that result in the real system being of different dimension than the available model. The work in this paper focuses on Single-Input Single-Output (SISO) systems using output feedback and dynamic controllers. The objective is to minimize network communication and the steady-state plant tracking error for step reference inputs. In addition, we extend these results to consider the case of uncertain systems affected by network induced delays. It is also shown that this approach can be used for stabilization of systems that transmit feedback measurements over uncertain additive Gaussian channels, where the unknown parameters that characterize the channel are treated as the uncertainties in the control system.

The problem of reference input tracking in NCS has been considered by different authors. Gao and Chen [12] proposed a new model based on the original plant and the model reference system. This new model considers a Zero-Order-Hold (ZOH) in the actuator node and it follows a sampled-data approach at the updating instants of the ZOH. Although parameter uncertainties are considered in the controller design step, the nominal model is not used between updates to estimate the state of the plant; reduction of transmission rate is not an objective in that work. Goodwin *et al.* [13] presented different NCS architectures and compared their properties for typical problems such as disturbance rejection and input tracking. One of these architectures considers models of both the plant and the network channel. The objective in [13] is to model the whole NCS at the controller node and generate a nominal output of the system that is compared to the real output and feed the controller using the resulting error. It is shown that under certain conditions this architecture outperforms common ones that do not use a model of the system in the control loop. In contrast to the results presented in the present paper, one significant drawback of the approach in [13] is that it is restricted to stable plants. Other approaches to this problem can be found in [14]-[16].

The paper is organized as follows: section II describes the model-based architecture. Conditions for stability and for bounded steady-state tracking error are presented in Section III. Section IV extends the results for a more general networked architecture. Network delays are considered in Section V. Illustrative examples are offered in section VI and section VII summarizes the results of the paper.

## II. MODEL-BASED NETWORKED ARCHITECTURE

MB-NCS were introduced in [4]; this configuration makes use of an explicit model of the plant (or system) which is added to the actuator/controller node to compute the control input based on the state of the model rather than on the plant state. The state of the model is updated when the controller receives the measured state of the plant. Fig. 1 shows the interconnection of several NCSs. The labeled small blocks correspond to each system's actuator and sensor nodes. We assume that the systems are decoupled, i.e. the dynamics of each system in Fig. 1 depend only on its own state. Without loss of generality we will focus on a particular system/model pair along with the corresponding actuator and sensor nodes.

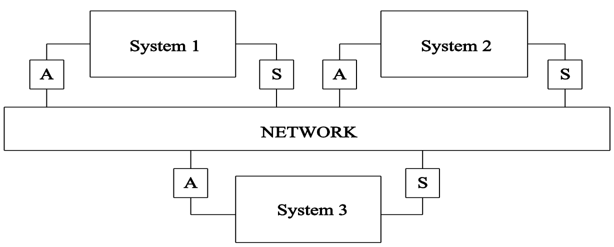


Fig. 1. Representation of NCSs.

In contrast to previous work in MB-NCS, we do not assume that the entire state vector is available but only an output of the system. In order to obtain a better tracking performance and to avoid implementation of state observers using uncertain parameters we use a transfer function representation for the model and the system. We consider discrete-time systems which are modeled by:

$$\hat{T}(z) = \frac{\hat{Y}(z)}{U(z)} = \frac{b_0 z^m + b_1 z^{m-1} + b_2 z^{m-2} ... + b_{m-1} z + b_m}{z^n + a_1 z^{n-1} + a_2 z^{n-2} + ... + a_{n-1} z + a_n}. \quad (1)$$

We consider strictly proper systems, i.e. $n>m$. The model (1) may be unstable i.e. not all poles of the transfer function $\hat{T}(z)$ have magnitude less than one. Typically, the model will represent the unstable dynamics of the real system. We consider uncertain systems that can be represented using stable and proper multiplicative or additive uncertainties:

$$T(z) = \hat{T}(z) \cdot \Delta T_M(z), \quad T(z) = \hat{T}(z) + \Delta T_A(z). \quad (2)$$

Let $T_u(z)$ represent in general either a multiplicative or an additive uncertainty. As a result, the model and the plant are, in general, of different order. If a state-space representation is to be used we will find that the model and plant state vectors have different dimensions and this type of uncertainty has not been considered yet in the MB-NCS setup. In order to deal with this dimensionality problem we implement the discrete-time model as a simple difference equation. The system output measurements are used directly to update the current and past output variables of the model without need of implementing a state observer.

In order to find the time instants that the sensor needs to send a measurement to the controller node we implement an event-triggered strategy. In event-triggered control [17]-[18] the sensor measures the state at every sampling time, it computes the state error and, based on the norm of this error, a decision is made whether the measurement needs to be sent to update the model. In the Model-Based Event-Triggered (MB-ET) framework the state error is defined as the difference between the current state and the state of the model [18]. Here, we use a similar definition for the error but in this case we are only able to measure the output of the system. The output error in this case is given by:

$$e(k) = \hat{y}(k) - y(k) \quad (3)$$

where $y(k)$ is the output of the system and $\hat{y}(k)$ is the output of the model. Node architectures for the set-point tracking model-based problem are shown in Fig. 2 and Fig. 3.

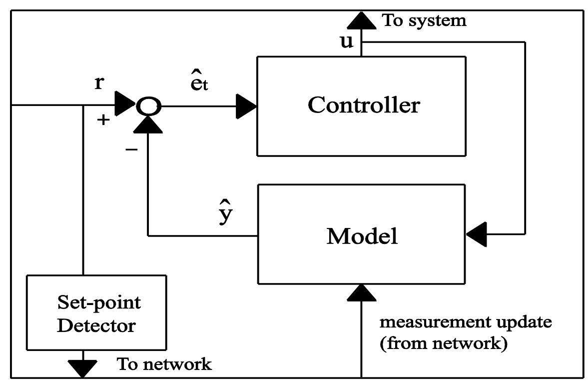


Fig. 2. Model-Based set-point tracking networked system: Actuator/controller node.

Fig. 2 represents the actuator/controller node and it contains the model $\hat{T}(z)$, the controller $C(z)$, which is designed based on the available model, and the set-point detector. The function of the set-point detector is to determine the time instants at which the reference input, which is only available at the controller node, changes its set-point value in order to transmit this information to the sensor node. Fig. 3 represents the sensor node. The sensor needs to compute the output error (3), and compare its absolute value to a fixed threshold. When the error is greater than the threshold an event is triggered and the current and $n$-1 past measurements of the plant are sent to the controller node. In order to calculate the output error the sensor needs the current value of the model output $\hat{y}(k)$ in addition to the plant output measurements. The exact copy of the output of the model can be easily obtained without sending frequent information as follows: copies of the model and controller are implemented in the sensor and at the time instants when the reference input changes values the controller node only needs to transmit the new set-point value. When a reference input value is received at the sensor node it is held until a new value arrives.

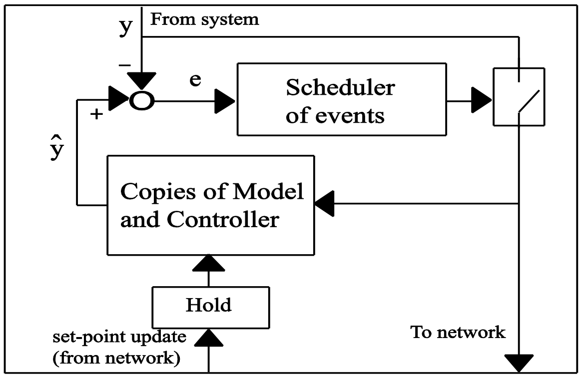


Fig. 3. Model-Based set-point tracking networked system: Sensor node.

The overall approach can be used for tracking of different types of reference input signals but it will be necessary to obtain an accurate estimation of the external input in the sensor node or to increase the communication rate from the controller to the sensor which is undesirable in networked implementations since other systems and applications need to communicate information. For this reason we restrict this framework to track piecewise constant signals which is general enough for many applications [22]-[23].

## III. OUTPUT FEEDBACK MB-ET CONTROL

The output feedback problem in MB-NCS has been studied mostly through the use of state observers as in [4]-[5]. The implementation of a state observer, of course, is obtained using the model parameters. For traditional non-networked systems, that is when continuous feedback is available, and in the absence of model uncertainties the design of observer and controller gains can be done independently from each other, i.e., the separation principle holds. However, in most real life problems there exists a model-plant mismatch and it is not possible to obtain a perfect estimation of the states of a system using uncertain parameters; see [19] for further details.

In the present paper the model in both the controller and the sensor nodes is implemented as a difference equation which represents the time-domain equivalent of the transfer function (1). The absolute value of the output error is compared to fixed positive threshold $\alpha$. When the relation $|e(k)| > \alpha$ holds then the sensor transmits a measurement update. At this point, the output error (3) is set to zero, since the model output is equal to the real output of the system. Therefore, the output error is bounded by:

$$|e(k)| \leq \alpha \tag{4}$$

When the sensor transmits an update according to the current output error, then it sends the current and $n$-1 past output measurements which are used to update the model in the controller. At the same time the sensor uses exactly the same measurements to update its own copy of the model.

Next, we provide conditions under which we are able to stabilize uncertain unstable systems with limited feedback. Furthermore, by using the internal model principle [20], [21] we are also able to bound the steady-state plant output tracking error defined by:

$$E_t(z) = R(z) - Y(z) \tag{5}$$

**Theorem 1**. The plant output tracking error corresponding to the networked system (2) with model (1) is bounded for any bounded reference step input if

a) The term 1+$T(z)C(z)$ has all its zeros inside the unit circle.

b) The poles of $T_u(z)$ have magnitude less than one.

c) The poles of the controller $C(z)$ contain the factor ($z$-1).

*Proof.* Define model output tracking error.

$$\hat{E}_t(z) = R(z) - \hat{Y}(z)\,. \tag{6}$$

The output of the plant is given by:

$$Y(z) = T(z)C(z)\hat{E}_t(z) = T(z)C(z)[R(z) - \hat{Y}(z)]$$

and using (3) we obtain the following:

$$Y(z) = T_{cl}(z)R(z) - T_{cl}(z)E(z) \tag{7}$$

where $T_{cl}(z) = \dfrac{T(z)C(z)}{1+T(z)C(z)}$.

The output tracking error is given by:

$$E_t(z) = \frac{1}{1+T(z)C(z)}R(z) + T_{cl}(z)E(z). \tag{8}$$

The reference input term in (8) is asymptotically stable for constant reference inputs $r(k)$ since the zeros of 1+$T(z)C(z)$ are inside the unit circle and the poles of the controller contain the factor ($z$-1). The second term in (8) contains the stable (according to conditions a) and b)) closed loop transfer function $T_{cl}(z)$ but the output error $E(z)$ is not constant. However, the output error is bounded by updating the model (and resetting the output error) every time the error's absolute value is greater than some positive threshold $\alpha$. ■

Stability of the networked system can be obtained from Theorem 1.

**Corollary 2**. The networked system (2) with model (1) is bounded-input bounded-output stable with respect to the error (3) if

a) The term 1+$T(z)C(z)$ has all its zeros inside the unit circle.

b) The poles of $T_u(z)$ have magnitude less than one. ■

*Remark 1*. The selection of the constant threshold $\alpha$ is made considering the following tradeoff: A small threshold results in a smaller bound on the steady state tracking error but, in general, it increases communication rate by sending measurement updates more frequently. A reduction on network usage can be achieved by increasing the threshold at the cost of a larger steady-state-tracking error.

*Remark 2*. The controller $C(z)$ is designed in such a way that the closed loop model is stable and with desired properties by selection of desired closed loop poles in addition to providing zero steady-state tracking error in the absence of model uncertainties.

## IV. TWO-CHANNEL NETWORKED SYSTEM

Fig. 4 shows a more general NCS architecture in which the communication network is used to connect the sensor node to the controller node and the controller node to the actuator node. Both actuator and sensor node functions are simplified and most of the computations are performed in the controller node. This architecture does not restrict the controller to be implemented or attached to the actuator node. It also provides the option to use a controller node as a controller for more than one subsystem, which is useful in many applications.

The controller node contains the model $\hat{T}(z)$ and the controller $C(z)$. The controller has access to the reference input signal as well. When the controller node receives a measurement update from the sensor it predicts the sequence of inputs $u(k)$ and the corresponding sequence of model outputs $\hat{y}(k)$ for $k = t_k \ldots t_k + N$, where $t_k$ is the latest update instant. This input and output prediction is made assuming that the reference signal remains constant during the prediction horizon $N$. The input sequence is sent to the actuator node in a single and larger packet. The actuator synchronizes the sequence by applying each input value at the corresponding time instant. Similarly, the model output sequence is sent to the sensor node in order to obtain the output error and apply the event policy.

If no error event update has been generated before or at time $t_k + N$ then a time-triggered update takes place and a new output measurement is sent to the controller to compute new sequences in order to repeat the cycle again for $k = t_{k+1} \ldots t_{k+1} + N$ with $t_{k+1} = t_k + N$. Note that the control process can be periodic but not necessarily, since an error event can occur at time $t_k + M$, for $M<N$, which makes $t_{k+1} = t_k + M$.

Finally, the controller also contains the set-point detector and it will send a request for a measurement update to the sensor node if a change in the set-point value occurs. This new event will initialize the prediction cycle no matter if error or time events have not been generated yet. The results in the previous section can be extended to consider this type of implementation and by following some mild assumptions.

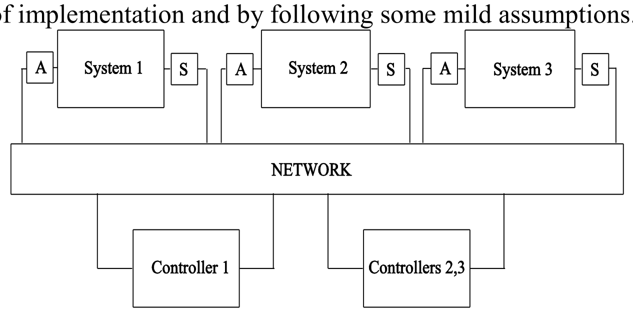


Fig. 4. Representation of two-channel NCSs.

**Theorem 3**. Assume that:

- All nodes (actuator, controller, and sensor) for a given networked system are synchronized.

- There exists a time-triggered sensor event equivalent to the duration of the input and model output sequences that are predicted by the controller.

Then the plant output tracking error is bounded for any bounded reference step input if

a) $1+T(z)C(z)$ has all its zeros inside the unit circle.

b) The poles of $T_u(z)$ have magnitude less than one.

c) The poles of the controller $C(z)$ contain the factor $(z-1)$.

*Proof.* The proof is similar to the one for Theorem 1 by noticing that the additional events trigger a measurement update which resets the output error in (8). ■

## V. Single-Channel Networked System with Delays

The results in section III assumed zero delay between the sensor and the controller nodes. By the nature of the event-triggered strategies, it may happen that several systems attempt to access the network at similar times. In this case only one node can gain access and the rest need to wait until the network turns into an idle state.

An approach to consider network induced delays within the MB-NCS framework is to propagate the delayed measurements received at the controller, i.e. to estimate the current output of the system based on the delayed measurements and using the model parameters. In this section we introduce a different approach for the single channel networked system that provides better results in terms of system performance and reduced network communication.

Assume that the measurement updates arrive at the controller/actuator node $d$ sampling times later, i.e. when a measurement is received at time $t_k$, it corresponds to an event generated at time $t_k - d$ which contains measurements $y(t_k - d) \ldots y(t_k - n - d)$. The advantage of using transfer function representation in this case is that the network induced delay can be represented by $z^{-d}$. Assuming a constant delay we are able to jointly model the dynamics of the system and the delay induced by the network as follows:

$$\hat{T}_d(z) = \hat{T}(z) \cdot z^{-d} \tag{9}$$

The new controller, represented by $C_d(z)$, is designed based on $\hat{T}_d(z)$, that is, the controller stabilizes $\hat{T}_d(z)$ and provides zero steady-state model output tracking error. The model $\hat{T}_d(z)$ is updated using the delayed measurements directly.

**Theorem 4**. The plant output tracking error corresponding to the networked system (2) with induced delays and with model (9) is bounded for any bounded reference step input if

a) The term $1+T_d(z)C_d(z)$ has all its zeros inside the unit circle, where $T_d(z) = T(z) \cdot z^{-d}$.

b) The poles of $T_u(z)$ have magnitude less than one.

c) The poles of $C_d(z)$ contain the factor $(z-1)$.

*Proof.* Let $\hat{Y}_d(z)$ represent the output of the model $\hat{T}_d(z)$ and $Y_d(z)$ the delayed output of the system, i.e. the output of $T_d(z)$. Define the errors $E_d(z) = \hat{Y}_d(z) - Y_d(z)$, $E_t^d(z) = R(z) - Y_d(z)$. It can be shown that the delayed output tracking error is given by

$$E_t^d(z) = \frac{1}{1+T_d(z)C_d(z)} R(z) + T_{cl}^d(z) E_d(z) \tag{10}$$

which is bounded by updating the model using the real output of the system, that is, when the controller receives delayed measurements it updates the model using those measurements directly, i.e. $\hat{y}_d(t_k) = y(t_k - d) = y_d(t_k)$ which makes $e_d(t_k) = 0$, where $T_{cl}^d(z) = \dfrac{T_d(z)C_d(z)}{1 + T_d(z)C_d(z)}$.

Since $E_t^d(z)$ represents the delayed version of $E_t(z)$ then the output tracking error is bounded as well. ■

**Corollary 5**. The networked system (2) with induced delays and with model (9) is bounded-input bounded-output stable with respect to the error (3) if

a) $1 + T_d(z)C_d(z)$ has all its zeros inside the unit circle.

b) The poles of $T_u(z)$ have magnitude less than one. ■

*Remark 3.* One important aspect in the implementation of this approach for the case of network delays is the computation of $e_d(k) = \hat{y}_d(k) - y(k-d)$. This task needs to be accomplished at every sampling time which requires the comparison of the outputs of the real system with no delay (2) and the delayed model (9). One way to compute this error is to use old system outputs, but since the current system output is available at the sensor node, then it can be used to compute the output error $e(k)$ instead of computing $e_d(k)$. Therefore we compute $e(k) = \hat{y}_d(k+d) - y(k)$ and the quantity $\hat{y}_d(k+d)$ is obtained by executing the model in the sensor node until time $k+d$ in order to obtain an estimate of $y_d(k+d) = y(k)$.

*Remark 4.* Controller complexity. The cost to be paid by using the approach described in this section compared to the usual prediction using the model with no delay is in the form of a more complex controller. The order of the controller increases since $C_d(z)$ is designed to control the higher order model $\hat{T}_d(z)$.

## VI. EXAMPLES

*Example 1*. (Uncertain system with delays). Consider the unstable model

$$\hat{T}_d(z) = \frac{z+1}{z^2 - 0.2z - 0.9} \cdot \frac{1}{z^3}. \qquad (11)$$

which models the dynamics of the system and a constant delay $z^{-3}$ equivalent to a 3-sample delay. The controller is designed in order to stabilize $\hat{T}_d(z)$ and to provide zero steady state model output tracking error and is given by:

$$C_d(z) = \frac{2.638z^5 - 1.102z^4 - 1.061z^3}{z^6 + 0.2z^5 + 0.8895z^4 + 0.3579z^3 + 0.8726z^2 - 2.141z - 1.179}. \qquad (12)$$

As it was mentioned before, the complexity of the controller increases by considering the model $\hat{T}_d(z)$ instead of $\hat{T}(z)$. The real system consists of the model dynamics with zero delay and the following multiplicative uncertainty:

$$\Delta T_M(z) = \frac{z + 0.61}{z + 0.55} \qquad (13)$$

then, the dynamics of the real system are given by:

$$T(z) = \frac{z^2 + 1.61z + 0.61}{z^3 + 0.35z^2 - 1.01z - 0.495}. \qquad (14)$$

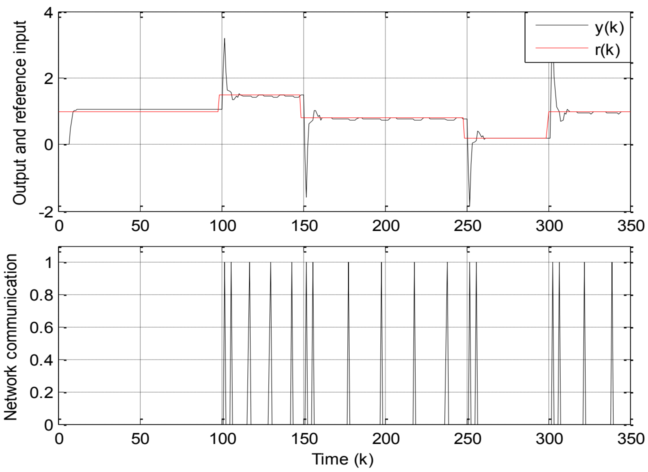


Fig. 5. System output and reference input for example 1 and for $\alpha$=0.05 (top). Network communication instants (bottom).

Note that the real system contains no delay since the delays are induced by the network when a feedback measurement is sent from the sensor node to the controller/actuator node. Good performance and reduction of communication are obtained as shown in the simulation results shown in Fig. 5. The network communication signal $n_c(k)$ represents the time instants at which output measurements are sent from the sensor node to the controller node. The rest of the time the networked system operates in open-loop mode.

$$n_c(k) = \begin{cases} 1 & \text{if measurements are sent at time } k \\ 0 & \text{if measurements are not sent at time } k \end{cases} \qquad (15)$$

The performance of the system using the delayed model and controller is considerably superior to the propagation approach used in previous work on MB-NCS [5], [24]. In order to show the imporved performance we simulate the same system and model using the same reference input, delay, and threshold. The difference is that we use the propagation method and the controller is designed for the no-delay model $\hat{T}(z)$. Results of simulation are shown in Fig. 6, which shows a poor tracking perfromance of the system and a significant increase in network communication.

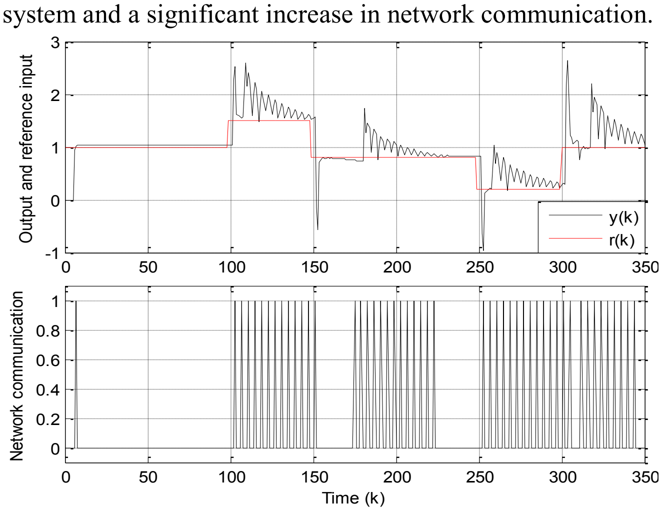


Fig. 6. Control of system with network delays using propagation for $\alpha$=0.05 (top). Network communication instants (bottom).

*Example 2*. Stabilization over an additive Gaussian channel. We consider the unstable system:

$$T(z) = \frac{1}{z^2 - 0.2z - 0.9}. \quad (16)$$

The system is assumed to be known. The measurements are tranmitted over an additive Gaussian channel modeled by:

$$y_g(k) = H_g y(k) + v(k) \quad (17)$$

where $v(k)$ is zero-mean Gaussian noise with variance $\sigma_v$ and $y_g(k)$ is the measurement received at the controller node. The parameter $H_g$ is assumed to be unknown and it represents the uncertainty in our approach in addition to the Gaussian noise. Fig. 7 shows the results of simulation when an impulse external disturbance perturbs the system and using $H_g = 0.8$ and $\sigma_v = 0.01$. The sytem is stable since the conditions in Corollary 2 are satisfied for this channel uncertainity.

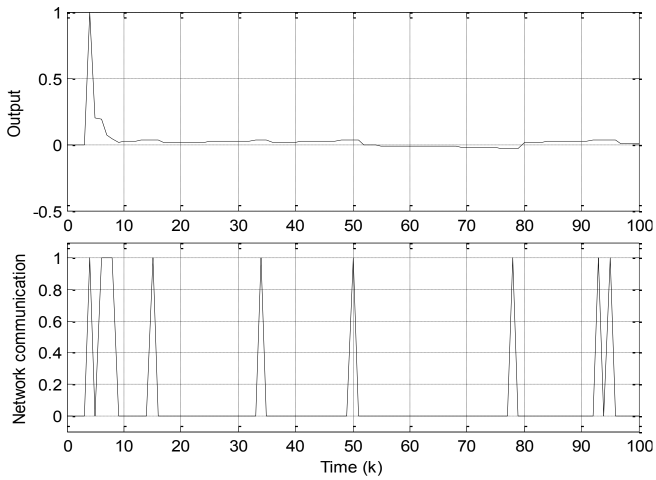


Fig. 7. Stabilization of system over an uncertain network. Output of the system for $\alpha$=0.03 (top). Network communication instants (bottom).

## VII. CONCLUSION

Output feedback control using the MB-NCS framework has been studied in this paper. The results in this paper also provide a way to control networked systems using models that do not necessarily have the same dimension as the uncertain system. In particular, the set-point tracking problem has been addressed in the presence of model-plant mismatch and in the absence of feedback measurements for extended periods of time. An important extension considered the same problem but in the presence of network induced delays. Simulation examples show the benefits of this scheme in relation to the tracking performance and the reduction of network bandwidth needed to control the system. It was also shown that the framework described in this paper is robust with respect to a class of uncertain channels when the dynamics of the system are known with certainty. The problem of stabilization when both the system and the channel are uncertain will be considered in future research. Future work will also address the time-varying delay case and for both the single and two-channel architecture.

## REFERENCES


[1] P. J. Antsaklis and J. Baillieul, "Special issue on technology of networked control systems," *Proc. IEEE*, vol. 95, no. 1, Jan. 2007.

[2] F. L. Lian, J. R. Moyne, D. M. Tilbury "Performance evaluation of control networks: Ethernet, Controlnet, and Devicenet," *IEEE Control Systems Magazine*, pp. 66 – 83, February 2001.

[3] F. L. Lian, J. M. Moyne, D. M. Tilbury, "Network design consideration for distributed control systems," *IEEE Transactions on Control Systems Technology,* vol. 10, no. 2, pp. 297–307, 2002.

[4] L. A. Montestruque and P. J. Antsaklis. "State and output feedback control in model-based networked control systems," in *Proceedings of the 41st IEEE Conference on Decision and Control*, 2002.

[5] L. A. Montestruque and P. J. Antsaklis "On the model-based control of networked systems," *Automatica*, vol. 39, no. 10, pp. 1837 - 1843, 2003.

[6] L. A. Montestruque and P. J. Antsaklis "Stability of model-based control of networked systems with time varying transmission times," *IEEE Transactions on Automatic Control,* vol. 49, no. 9, 2004.

[7] J. Lunze and D. Lehmann, "A state-feedback approach to event-based control," *Automatica*, vol. 46, no.1, pp.211-215, 2010.

[8] H. Yu, A. Wang, and Y. Zheng, "On the model-based networked control for singularly perturbed systems," *Control Theory and Applications,* vol. 6, no.2, pp.153-162, 2008.

[9] M. Yu, L. Wang, T. Chu, and G. Xie, "Stabilization of networked control systems with data packet dropout and network delays via switching system approach," in *Proceedings of the 43rd IEEE Conference on Decision and Control*, 2004.

[10] E. Garcia, G. Vitaioli, and P. J. Antsaklis, "Model-based tracking control over networks," in *Proceedings of IEEE International Conference on Control Applications,* 2011.

[11] Y. Sun and N. H. El-Farra, "Quasi-decentralized model-based networked control of process systems," *Computers and Chemical Engineering,* vol. 32, pp. 2016-2029, 2008.

[12] H. Gao and T. Chen, "Network-based H-inf output tracking control," *IEEE Trans. on Automatic Control,* vol. 53, no. 3, pp. 655-667, 2008.

[13] G. C. Goodwin, D. E. Quevedo, and E. I. Silva, "Architectures and coder design for networked control systems," *Automatica*, vol. 44, pp. 248-257, 2008.

[14] S. Yuksel, H. Hindi, L. Crawford, "Optimal tracking with feedback-feedforward control separation over a network", *in Proceedings of the American Control Conference*, 2006.

[15] N. V. Wouw, P. Naghshtabrizi, M. Cloosterman, J. P. Hespanha, "Tracking control for Networked Control Systems", *46th IEEE Conference on Decision and Control*, 2007.

[16] M. B. G. Posthumus-Cloosterman, "Control over communication networks: modeling, analysis, and synthesis," Ph.D. dissertation, Tech.Univ. Eindhoven, Eindhoven, The Netherlands, 2008.

[17] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *IEEE Trans. Automatic Control,* vol. 52, pp. 1680-1685, Sept. 2007.

[18] E. Garcia and P. J. Antsaklis, "Model-based event-triggered control with time-varying network delays," in *Proceedings of the IEEE Conference on Decision and Control-European Control Conference,* 2011.

[19] E. Garcia, "Model-based control over networks: architecture and performance," Ph. D. Dissertation, University of Notre Dame, Notre Dame, IN, June 2012.

[20] B. A. Francis and W. M. Wonham, "The internal model principle of control theory," *Automatica,* vol. 12, pp. 457-465, 1976.

[21] R. C. Dorf, and R.H. Bishop, *Modern control systems*, 11th ed. Prentice Hall, Upper Saddle River, NJ, 2008.

[22] J. Richalet, A. Rault, J. L. Testud, and J. Papon, "Model predictive heuristic control: applications to industrial processes," *Automatica*, vol. 14, pp. 413-428, 1978.

[23] N. F. Thornhill, B. Huang, and S. L. Shah, "Controller performance assessment in set point tracking and regulatory control," *Intl. Journal of Adaptive Control and Signal Processing,* vol. 17, pp. 709-727, 2003.

[24] Z. Wang, W. Liu, H. Dai, and D. S. Naidu, "Robust stabilization of model-based uncertain singularly perturbed systems with network time-delay," in *Proceedings of the 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, 2009.