INTERNATIONAL JOURNAL OF
NUMERICAL ANALYSIS AND MODELING
Volume 5, Number 4, Pages 590–611

A MIXED-INTEGER PROGRAMMING APPROACH TO NETWORKED CONTROL SYSTEMS

GUOFENG ZHANG, XIANG CHEN, AND TONGWEN CHEN

Abstract. This paper studies the problem of controller design for networked control systems regulated by a network data transmission protocol proposed in [50]. In this framework, the plant is first formulated as a mixed logical dynamical (MLD) system, then model predictive control (MPC) based on the mixed-integer programming is adopted to design a controller to guarantee certain control performance. It is shown that the solvability of the finite-horizon MPC is not equivalent to that of the infinite-horizon MPC, which is normally true for most existing MPC methods. The non-convexity feature of this type of networked control systems rules out explicit piecewise affine controllers that are designable for linear convex control systems. Notwithstanding these difficulties, controller design is still feasible due to the special nature of the data transmission strategy, i.e., only a small number of logic values are involved. Furthermore, control of higher-order systems and tracking of more complicated signals can be readily dealt with using this new approach. Two examples are presented to illustrate the strength of the proposed approach.

Key Words. model predictive control, networked control systems, non-convexity, mixed-integer programming, mixed logical dynamical systems, hybrid systems.

1. INTRODUCTION

The fast development of secure high-speed communication networks ([40, 29]) renders control over networks possible. The insertion of a communication channel into a control loop brings in many advantages, including wire reduction, low cost and easy installation and maintenance, etc.. Thanks to these merits, networked control systems have been built successfully in various fields like automotive control ([18, 28]), aircrafts manufacturing ([33, 36]), and robotic control ([20, 35]). Unfortunately, since most types of signals, like the encoded system output, controller output and other information, are transmitted through communication channels shared by a lot of users, traffic congestion always occurs which of course degrades control performance. Traffic congestion usually manifests itself in the form of time

Received by the editors May 12, 2007 and, in revised form, November 15, 2007.
2000 *Mathematics Subject Classification.* 49M25, 90C26, 93C10.
This research was supported by NSERC (Canada) and a Start-up Fund from the University of Electronic Science and Technology of China.

delays, packet loss, and other undesirable effects on the control systems. How to tackle these adverse effects has become a major subject of research in the control community and such closely related fields as communication and computing. A variety of network protocols and lots of control techniques have already been proposed and analyzed to tackle this issue. To classify very crudely, these considerations can be categorized into three classes.

The first category contains simplest approximations where networked control systems are modeled as feedback control systems having bounded deterministic time delays. For instance, a so-called try-once-discard (TOD) protocol is recently proposed and studied extensively by Walsh *et al.* ([41, 42, 43, 44, 45]). In this protocol, it is assumed there is no network from controller to actuator, while an upper bound of sensor-to-controller time delays induced by the network is obtained based on the perturbation theory, for which the resulting closed-loop system is exponentially stable. This protocol is further developed in [24], [25] to establish a class of Lyapunov UGES (Uniformly Globally Exponentially Stable) protocols in the L^p space framework. In reference [48], by assuming bounded time delays and packet dropouts, the authors investigate a robust H_∞ control problem for networked control systems. However, all the above papers did not study the issue of time delays or packet loss from controllers to actuators. A possible difficulty in dealing with controller-to-actuator delays and packet loss may be: A pre-defined controller is usually unable to predict and then compensate for time delays or packet loss from controllers to actuators. Based on the above discussion, techniques in this class usually are quite conservative because of their inherent limitations in system modeling, as has been widely acknowledged.

The foregoing conservativeness has triggered the development of the second category of methods, where network-induced time delays and packet loss are modeled as random processes, typically Markov chains. Via this modeling, specific properties of these stochastic processes can be deployed to facilitate the design of controllers guaranteeing desirable control performance. For instance, in [18], by assuming time delays as Markov chains, a jumped linear system is constructed via state augmentation. Moreover, several necessary and sufficient conditions for zero-state mean-square exponential stability have been established for this type of networked control systems. A similar approach is adopted in [49] where sensor-to-controller and controller-to-actuator time delays are assumed to behave according to two Markov chains respectively. LMI techniques are then employed to address the stabilization problem. In [26], both sensor-to-controller and controller-to-actuator time delays are modeled as independent white-noise with zero mean and unit variance; consequently, a (sub)optimal stochastic control problem is formulated and studied. Two Matlab toolboxes, Jitterbug and TrueTime, are proposed recently in [8] based on the assumption that a networked control system can be approximated by a sampled-data control system with signal quantization and time delays involved. These two toolboxes can be readily adopted to determine how sensitive a control system is to delays, jitters, and lost samples, etc. As a result, they can be utilized as testbed for designing and analyzing real-time networked control systems.

The above two classes of methodologies tackle adverse network effects passively, i.e., they just deal with the adverse influence of network traffic on control systems, while ignoring the interactions between communication channels and control systems. Clearly, interaction between networks and control systems are quite practical and hence should be take into account. This consideration has inspired the third category of methodologies that address the tradeoff between data transmission rate and performance of control systems. Some interesting work has been done in this direction. For instance, in order to minimize bandwidth utilization, Goodwin *et al.* [15] propose to use quantization to reduce the size of the transmitted data and solve the problem via a moving horizon technique. The adoption of moving horizon techniques is natural as it can effectively deal with constraints caused by quantization. The concept of containability is proposed in [47], which characterizes a certain stability of networked control systems. Moreover, the effect of signal quantization error, quantization time and propagation time on containability is investigated. In [37], two classic control concepts, observability and stabilizability, are generalized for networked control systems. Under finite data rate constraint, the lower bound of data rates is obtained for the system to be asymptotically observable (or stabilizable). The resultant lower bound turns out to be the summation of the logarithms of modules of the unstable system poles. These fundamental results are further generalized into the study of control problems over noisy channels in [38]. An LQG optimal control problem of an unstable scalar system is studied in [11], where the communication network is an additive white Gaussian noise (AWGN) channel. For this case, it is shown that the achievable data transmission rate is governed by the fundamental Bode sensitivity integral formula. This result is rather interesting because it establishes the equivalence between feedback stabilization through an analog communication channel and a communication scheme based on feedback, thus unifying the design of control systems and communication channels.

In [50], a new data transmission strategy is proposed aiming at reducing network traffic congestion. The basic idea is: By adding constant deadbands to both a controller and a plant to be controlled, signals will be sent *only* when necessary. By designing the deadbands carefully, a tradeoff between control performance and reduction of network data transmission rate can be achieved. This network data transmission protocol is able to fit a control network into an integrated communication network composed of control and data networks, so as to fulfill the need for a new breed geared toward total networking. Seamless integration of control systems into communication networks is clearly very appealing as well as promising as depicted by Raji [31]; This kind of integration is so fundamentally important that it is regarded as a fundamental future direction in control research in an information-rich world [23]. Essentially speaking, under the network data transmission strategy proposed in [50], in an integrated network consisting of data and control networks, it is requested that the network provide sufficient communication bandwidth upon the request of control systems. As a payoff, control systems will save network resources by deliberately dropping packets while without degrading system performance severely. This is a crucial tradeoff. On the one hand, control signals are normally time critical, hence the priority should be given to them whenever requested; on the other hand, due to one characteristic of control networks, namely, small packet size but frequent packet flows, they demands frequent transmissions. The proposed scheme aims at relieving this burden on the whole integrated communication network.

The dynamics of the above scheme are analyzed in [50, 51, 52], where it is found that in contrast to its very simple appearance, it gives rise to many unexpected and interesting dynamical phenomena and mathematical problems. For example, suppose that the system G to be controlled is first-order, linear and time-invariant, and a linear time-invariant controller C is designed without taking the network traffic into account, the system behaves chaotically. In fact, it is shown by Theorem 6 in [50] that if either G or C is unstable, the closed-loop system can not be asymptotically stable. A step signal tracking problem is discussed in [51], where it is shown that for an unstable second-order system, by modifying the control scheme carefully, tracking errors are reasonably small and simultaneously very low transmission rate is demanded by the system. However, it is hard to extend that method to higher-order systems and/or other tracking problems such as tracking a sinusoidal signal.

In this paper we will attack these difficulties by means of some tools that are fairly recent vintage. More specifically, we will transform the problem to a mixed-integer programming problem. The block consisting of G and H_1 (see Fig. 1) is converted to a mixed logical dynamical (MLD) system which is a system involving both real and integer variables as well as integer (in)equalities. Within this context, a finite-horizon MPC problem is formulated. Note that the problem under study is a problem of joint control and communication utilization reduction, besides the control performance specification, one performance index for network utilization is also included explicitly into the MPC performance specification. In general, such an optimization problem is hard to solve. Fortunately, due to the characteristic of the problem, the mixed logical dynamical (MLD) systems framework recently developed by Bemporad *et al.* ([2], [39]) can be adopted to design controllers to achieve satisfying control while at the same time reduce network traffic. Moreover, control of higher-order systems as well as more complicated signal tracking can be addressed readily via this approach. Two examples illustrate that the difficulty in sinusoidal signal tracking posed above can be addressed neatly using this method.

This paper is organized as follows: The proposed network protocol is presented in Section II. Stability of MPC is discussed in Section III. In Section IV, the networked system is first converted into a mixed logical dynamical system, then a finite-horizon MPC problem is formulated and investigated. Two examples are given in Section

V to illustrate the effectiveness of this new configuration. Section VI concludes this paper.

2. THE PROPOSED NETWORK TRANSMISSION STRATEGY

For completeness, the network data transmission strategy proposed in [50] is briefly reviewed in this section. Consider the feedback control system as shown in Fig. 1, where G is a plant in discrete time that is of the form:

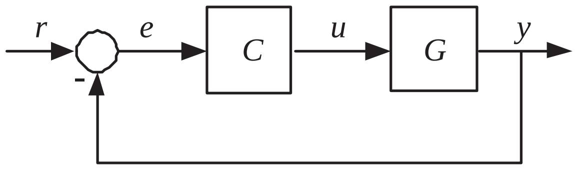

FIG. 1. A typical feedback system

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

with the state $x \in \mathbb{R}^n$, the input $u \in \mathbb{R}^q$, the output $y \in \mathbb{R}^p$, and the reference input $r \in \mathbb{R}^p$, respectively; C is a stabilizing discrete-time controller:

$$\begin{aligned} x_d(k+1) &= A_d x_d(k) + B_d e(k), \\ u(k) &= C_d x_d(k) + D_d e(k), \\ e\,(k) &= r\,(k) - y\,(k)\,, \end{aligned}$$

with its state $x_d \in \mathbb{R}^{n_c}$. Let $\zeta = \begin{bmatrix} x \\ x_d \end{bmatrix}$. It is easy to show that the closed-loop system from r to e can be formulated as

$$\begin{aligned} \zeta\,(k+1) &= \begin{bmatrix} A - BD_dC & BC_d \\ -B_dC & A_d \end{bmatrix} \zeta\,(k) + \begin{bmatrix} BD_d \\ B_d \end{bmatrix} r(k), \\ e(k) &= \begin{bmatrix} -C & 0 \end{bmatrix} \zeta\,(k) + r(k). \end{aligned}$$

Next, we add two logic blocks on both u and y. More concretely, consider the system as shown in Fig. 2. The logic block H_1 is defined as follows: For given $\delta > 0$,

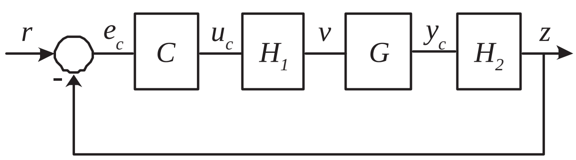

FIG. 2. A constrained feedback system

take $v(-1) = 0$; and for $k \geq 0$, let

$$v(k) = H_1\left(u_c(k), v(k-1)\right) = \begin{cases} u_c(k), & \text{if } \left\|u_c(k) - v(k-1)\right\|_\infty > \delta, \\ v(k-1), & \text{otherwise.} \end{cases}$$

Similarly, H_2 is defined as follows: for given $\delta_2 > 0$, take $z(-1) = 0$; for $k \geq 0$, let

$$z(k) = H_2\left(y_c(k), z(k-1)\right) = \begin{cases} y_c(k), & \text{if } \left\|y_c(k) - z(k-1)\right\|_\infty > \delta_2, \\ z(k-1), & \text{otherwise.} \end{cases}$$

It is provable that $\|H_1\|$, the ℓ_∞ induced norm of H_1, equals 2, and so is $\|H_2\|$.

In [27] *adjustable* deadbands are proposed to reduce network traffics, where the closed-loop system with deadbands is modeled as a perturbed system, and its exponential stability follows that of the original system [17]. The constraints, δ and δ_2 proposed here, are fixed. We have observed ([50, 51, 52]) that the stability problem of the system shown in Fig. 2 is fairly complicated and only local stability can possibly be obtained. However, the main advantage of fixed deadbands is that it will reduce network traffics more effectively. Furthermore, the stability region can be scaled as large as desired (at least for low order systems).

The effect of adding deadbands into the network is different from quantization. A one-dimensional quantizer with a quantization size Δ is a map from $\mathbb{R}$ to a countable subset $L = \{l_j\}_{j\in\mathbb{Z}}$ of $\mathbb{R}$. Such a quantizer can be constructed in the following manner: Given a scalar $\Delta > 0$, partition continuously $\mathbb{R}$ into countably many subintervals $\Gamma := \{\Gamma_j\}_{j\in\mathbb{Z}}$ satisfying 1) $\bigcup_{j\in\mathbb{Z}} \Gamma j = \mathbb{R}$, 2) $\Gamma_j \bigcap \Gamma_i = \phi$, $\forall i \neq j$, 3) $0 \in \Gamma_0$. One choice of the partition methods is: $\Gamma_j = [(j-1/2)\Delta, (j+1/2)\Delta)$, $j \in \mathbb{Z}$. Choose a countable subset $L = \{j\}_{j\in\mathbb{Z}}$ of $\mathbb{R}$. Then the quantization is a map from $\mathbb{R} \to L$ defined as $x \longmapsto j$, $\forall x \in \Gamma_j$. If Δ is fixed, then the quantization is called *time-invariant*, otherwise *time-varying*. Given a time-invariant quantizer, two signals arbitrarily close to one another may be mapped to two different values. For example, consider the quantizer given above with $\Delta = 1$. Let $j = 0$. Then $-1/2-\varepsilon$ and $-1/2+\varepsilon$ are mapped to -1 and 0 respectively, no matter how small ε is. Clearly this is undesirable. Our scheme does not assume a fixed partition of the space; instead, the partition is regulated by the time-varying $v(k-1)$ and $z(k-1)$, therefore it is more flexible.

3. STABILITY OF MPC

Model predictive control (MPC) has been accepted as a standard technique in controlling multivariable systems under various output/input/state constraints. At each sampling instant, assuming that the current state is available, an open-loop (sub)optimal control law is calculated over a finite-horizon by solving a quadratic optimal control problem, thus obtaining a sequence of future input, and the first one is then sent to the actuator. At the next sampling instant, this procedure is repeated using the new measured state and over a finite-horizon. So we can see that the finite-horizon optimization eases the solvability of the optimization problem involved while at the cost of the difficulty in ensuring closed-loop stability.

Of course on-line computation is also a serious problem in MPC. Here we will review some standard methods in the literature that address closed-loop stability. We find that the stability problem of a finite-horizon MPC is usually transformed to that of an infinite-horizon MPC, either explicitly or implicitly.

Consider the following linear discrete-time system

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

where the state $x \in \mathbb{R}^n$, the input $u \in \mathbb{R}^m$, and the output $y \in \mathbb{R}^p$. u and y satisfy the following constraints

$$u_{\min} \leq u(k) \leq u_{\max}, \tag{2}$$

$$y_{\min} \leq y(k) \leq y_{\max}, \tag{3}$$

for all $k \in Z^+$, the set of non-negative integers; "$\leq$" is element-wise. Clearly the constraints on u and y are convex.

Assume the current state $x(k)$ is available at the current sampling instant k, MPC solves the following quadratic optimization problem:

$$P(M, N, x(k)) : \min_{\mathcal{U}=\{u_{k|k}, u_{k+1|k}, \cdots, u_{k+M-1|k}\}} \left\|x_{k+N|k}\right\|_P + \sum_{i=0}^{N-1} \left(\left\|x_{k+i|k}\right\|_Q + \left\|u_{k+i|k}\right\|_R \right)$$

$$\text{Subject to: } x_{k+i+1|k} = Ax_{k+i|k} + Bu_{k+i|k}, \ i = 0, 1, \cdots, N-1, \tag{4}$$

$$y_{k+i|k} = Cx_{k+i|k}, \ i = 0, 1, \cdots, N-1, \tag{5}$$

$$u_{\min} \leq u_{k+j|k} \leq u_{\max}, \ j = 0, 1, \cdots, M-1, \tag{6}$$

$$y_{\min} \leq y_{k+i|k} \leq y_{\max}, \ i = 0, 1, \cdots, N-1, \tag{7}$$

where $x_{k|k} = x(k)$ is the current state, and $x_{k+i|k}$ $(1 < i \leq N)$ can be determined by Eq. (4) with $\mathcal{U}$ being the input. $\left\|x_{k+N|k}\right\|_P := x_{k+N|k}^T P x_{k+N|k}$; $\left\|x_{k+i|k}\right\|_Q$ and $\left\|x_{k+i|k}\right\|_R$ are defined similarly, where P, Q and R are all non-negative matrices of compatible sizes. Note that when $M < N$, $u_{k+j|k}$ $(j = M, \cdots, N-1)$ are undefined in the optimization problem $P(M, N, x(k))$. There are two choices adopted in the literature. One is simply to let them be 0, and the other is to assume that they equal $Kx_{k+j|k}$ $(j = M, \cdots, N-1)$ where K is a constant gain and $x_{k+j|k}$ are calculated states. At each sampling instant k, assuming that the current state $x(k)$ is available, the above optimization problem $P(M, N, x(k))$ is solved, bringing in a predicted input sequence $\{u_k, u_{k+1}, \cdots, u_{k+M-1}\}$, then the first, namely u_k, is applied to system (1) while the others are discarded. This procedure is repeated at the next sampling instant $k+1$, and so on. Hence one can see that when M and N are finite, the *finite-horizon* nature of this optimization problem allows the constraints on input and output to be satisfied, however this is at the cost of huge computation effort because at each sampling instant an optimization problem has to be solved *online*, which limits its application to systems whose dynamics are relatively slow. Therefore, quite remarkably, if the feasible sets of state and input are polytopes, it is proved in [3] that the optimization problem $P(M, N, x(k))$ in effect admits a convex and piecewise affine solution, thus the control law can be determined *off-line*, therefore moving the computation burden *off-line*. Clearly this greatly enlarges the applicability of MPC. For convenience, from now on we denote the optimization problem with $N = \infty$ by $P(M, \infty, x(k))$ and with $M = N = \infty$ by $P(\infty, \infty, x(k))$ respectively.

Stability is the most fundamental issue in all control systems. For simplicity, suppose that the stability problem is to steer the initial state $x(0)$ to the origin. To that end, almost all existing methods assume either explicitly or implicitly that the optimization problem $P(\infty, \infty, x(0))$ is solvable when $x(0)$ is in a small neighborhood of the origin, say, Ω. This is indeed quite intuitive: when the system starts somewhere close to the origin, *small* input effort is sufficient to drive the state to the origin asymptotically while the input constraint as well as the output constraint are satisfied automatically. In other words, the problem reduces to a traditional LQR problem. Based on this idea, the stability problem boils down to the feasibility of either the optimization problem $P(M, N, x(k))$ with finite M and N or $P(M, \infty, x(k))$. As far as $P(M, \infty, x(k))$ is concerned, by supposing that the input and the state are bounded, an input-horizon M is sought which steers an initial state into an invariant subset Λ of Ω. Due to the boundedness of the state, such M always exists. Then the state will enter Λ and stay there forever due to the invariance of Λ. As commented before, starting from a state $x \in \Omega$ the LQR problem with the input constraint and the output constraint is solvable; moreover, because Λ is an invariant subset of Ω, $P(M, \infty, x(k))$ is also solvable, i.e., a stabilizing law always exists ([12], Theorem 1 in [3] and the references therein). This idea is also used in [32] and [1] and has been the driving force behind much of the recent development. On the other hand, for $P(M, N, x(k))$, assuming $M = N$, Keerthi and Gilbert [16] prove that as $N \to \infty$, the finite-horizon optimization problem $P(N, N, x(k))$ is equivalent to the infinite-horizon optimization problem $P(\infty, \infty, x(0))$. Hence the solvability of $P(\infty, \infty, x(0))$ is assumed implicitly. This result has inspired much subsequent research (see, e.g., [22],[6],[55],[3]). So Mayne, *et al.* comment presciently in [21], "In some cases the finite-horizon optimal control problem solved on-line is exactly equivalent to the same problem with an infinite-horizon; in other cases it is equivalent to a modified infinite-horizon optimal control problem." Unfortunately we will show that due to the special characteristic of the problem being studied here, the solvability of the optimization problem $P(\infty, \infty, x(0))$ is *not* available. This leads to virtually no tools available for its stability analysis. Nevertheless, the prediction control problem can still be solved efficiently to address stability as well as tracking problem.

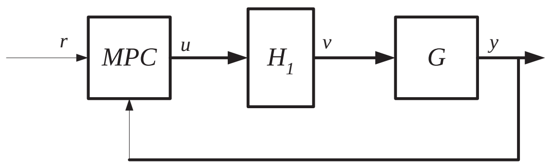

FIG. 3. Networked-based MPC control system

4. MAIN RESULTS

In this section, we study the network-based control problem raised in Section 2. Essentially speaking, this is a joint control and communication problem, where both control performance and network resource utilization are considered simultaneously. We first convert the system to be controlled under the network data transmission strategy into a mixed logical dynamical (MLD) system. Then for a performance specification consisting of both control performance and network traffic rate, a model predictive controller is designed. This re-configuration enables us to adopt optimization techniques recently developed for predictive control of hybrid systems to design a controller taking into account both control performance and reduction of network data transmission rate.

4.1. System Reformulation. In this subsection, we convert the compound block consisting of G and H_1 into a mixed logical dynamical system. For the sake of simplicity, we only consider the network traffic from control to actuator. More concretely, consider the configuration in Fig. 3, where the system G is given by

$$\begin{aligned} x(k+1) &= Ax(k) + Bv(k), \\ y(k) &= Cx(k), \end{aligned} \tag{8}$$

in which

$$v(k) = H_1(u(k), v(k-1)) = \begin{cases} u(k) & \text{if } |u(k) - v(k-1)| > \delta, \\ v(k-1) & \text{otherwise.} \end{cases} \tag{9}$$

For ease of presentation, define

$$z(k) := v(k-1).$$

Then the system composed of Eqs. (8)-(9) becomes

$$\begin{aligned} \begin{bmatrix} x(k+1) \\ z(k+1) \end{bmatrix} &= \begin{bmatrix} Ax(k) + Bu(k) \\ u(k) \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x(k) \\ z(k) \end{bmatrix} + \begin{bmatrix} B \\ I \end{bmatrix} u(k), \\ y(k) &= Cx(k), \end{aligned} \tag{10}$$

if $|u(k) - z(k)| > \delta$; otherwise,

$$\begin{aligned} \begin{bmatrix} x(k+1) \\ z(k+1) \end{bmatrix} &= \begin{bmatrix} Ax(k) + Bz(k) \\ z(k) \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x(k) \\ z(k) \end{bmatrix} + \begin{bmatrix} B \\ I \end{bmatrix} z(k), \\ y(k) &= Cx(k). \end{aligned} \tag{11}$$

Denote

$$\bar{A} = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}, \ \bar{B} = \begin{bmatrix} B \\ I \end{bmatrix}, \ \bar{C} = [C \ 0],$$

and define

$$\begin{aligned} \xi(k) &:= u(k) - z(k), \\ X &= \left[x^T \ z^T\right]^T, \end{aligned} \tag{12}$$

where the superscript "T" stands for the matrix transpose operator. Then the system composed of Eqs. (10)-(11) is equivalent to

$$\begin{aligned} X(k+1) &= \begin{cases} \bar{A}X(k) + \bar{B}u(k) & \text{if } |\xi| > \delta, \\ \bar{A}X(k) + \bar{B}u(k) - \bar{B}\xi(k) & \text{otherwise.} \end{cases} \\ y(k) &= \bar{C}X(k) \end{aligned} \tag{13}$$

Remark 1. Note that the constraint $|\xi| > \delta$ in Eq. (13) is *nonconvex*. This feature distinguishes system (13) from all the systems studied by Bemporad *et al.* ([2], [39], [3], [4], [9], [10], [19]). In fact, due to this characteristic, the newly developed hybrid system toolbox HYSDEL ([5]) is unable to convert system (13) from the form of a mixed logical dynamical system into that of an equivalent piece-wise affine system.

System (13) is a switched system under a logical law. In general, it is not easy to control such systems even if the consideration of network traffic rate reduction is neglected. Next we convert this logical law to a logical value. To do that, let us first recall some Boolean connectives as listed in Table 1. By means of these

Table 1. Boolean algebra connectives

$\wedge$	and
$\vee$	or
$\sim$	not
$\rightarrow$	implies
$\leftrightarrow$	if and only if
$\oplus$	exclusive or

Boolean connectives, the *literal* $|\xi(k)| > \delta$ can be associated with a *logical* value $\gamma(k)$ via

$$[|\xi(k)| > \delta] \leftrightarrow [\gamma(k) = 1]\,. \tag{14}$$

(For a comprehensive treatment of propositional calculus and integer programming, interested readers may refer to [46, 7].) In terms of Eq. (14), system (13) can be transformed to

$$\begin{aligned} X(k+1) &= \left(\bar{A}X(k) + \bar{B}u(k)\right)\gamma(k) + \left(\bar{A}X(k) + \bar{B}u(k) - \bar{B}\xi(k)\right)(1 - \gamma(k)) \\ &= \bar{A}X(k) + \bar{B}u(k) - \bar{B}\xi(k)\,(1 - \gamma(k))\,, \\ y(k) &= \bar{C}X(k). \end{aligned} \tag{15}$$

Define

$$\eta(k) := \xi(k)\,(1-\gamma(k)) \tag{16}$$

to remove the nonlinearity, system (15) is then equivalent to

$$\begin{aligned} X(k+1) &= \bar{A}X(k)+\bar{B}u(k)-\bar{B}\eta(k), \\ y(k) &= \bar{C}X(k). \end{aligned} \tag{17}$$

Define an upper and a lower bound of ξ to be

$$M = \max\{\xi\}\,,\ m = \min\{\xi\}. \tag{18}$$

These bounds are usually specified by the practical consideration, and they facilitate the derivation of linear inequalities that will serve as constraints in the optimization problem. It is easy to show that

$$\eta(k) \le M\,(1-\gamma(k))\,,\ \eta(k) \ge m\,(1-\gamma(k))\,,\ \eta(k) \le \xi(k)-m\gamma(k),\ \eta(k) \ge \xi(k)-M\gamma(k). \tag{19}$$

Furthermore, define

$$[\xi(k) > \delta] \leftrightarrow [\alpha(k)=1]\,,\ [\xi(k) < -\delta] \leftrightarrow [\beta(k)=1]\,.$$

Consequently,

$$\{[\alpha(k)=1] \vee [\beta(k)=1]\} \leftrightarrow [\gamma(k)=1]\,,$$

which is equivalent to

$$\alpha(k) \le \gamma(k),\ \beta(k) \le \gamma(k),\ \gamma(k) \le \alpha(k)+\beta(k). \tag{20}$$

Also, it is easy to verify that

$$[\xi(k) > \delta] \leftrightarrow [\xi(k)-\delta > 0] \leftrightarrow [\alpha(k)=1]$$

if and only if

$$\xi\,(k)-\delta < (M-\delta)\alpha(k),\ \xi\,(k)-\delta > \epsilon+(m-\delta-\epsilon)(1-\alpha(k)). \tag{21}$$

where ϵ is a positive number which is sufficiently small. Similarly,

$$[\xi(k) < -\delta] \leftrightarrow [\xi(k)+\delta < 0] \leftrightarrow [\beta(k)=1]$$

if and only if

$$\xi\,(k)+\delta < (M+\delta)(1-\beta(k)),\ \xi\,(k)+\delta > \epsilon+(m+\delta-\epsilon)\beta(k)). \tag{22}$$

Via the above procedure, the system consisting of G and H_1 has become a mixed logical dynamical (MLD) system (17) with (in)equality constraints (12), (16) and (19)-(22). For convenience, we hereafter denote it by Σ. It is clear that the state of Σ is X, the input is u, and the output is y. ξ, η, α, β and γ are all auxiliary variables. Given an MLD system, the first thing to check is if it is *well-posed.* It can be verified that, once the initial condition $x(0)$ and $z(0)$ (which is $v(-1)$) and a control sequence are given, all other variables are uniquely determined, hence Σ is well-posed. It is worthwhile to notice that, due to the nonconvexity of the constraint (see *Remark* 1), system Σ is different from those MLD systems defined in [2], and thus make the derivation of explicit control laws impossible.

Keeping in mind that we are now studying a network-based control problem, hence in addition to stability and performance of the control systems, network resource utilization must also be taken into account. In light of this, we define a new integer variable

$$\omega(k) = 1 - \gamma(k). \tag{23}$$

Note that when $\omega(k) = 0$, there is one network data transmission; otherwise, no data transmission.

4.2. MPC Control. In this subsection we study the following problem: How to design a controller for system Σ such that the closed-loop system has satisfactory control performance and meanwhile network traffic is reduced reasonably. Observe that the logic law has been converted to linear (in)equalities constraints, hence we are motivated to seek a desirable control law using MPC techniques. In general, it is not easy to find an appropriate prediction control law for an MLD system because the system is essentially nonlinear and integer variables are involved. Fortunately some effective tools have been developed recently ([2], [3], [39], [4]) based on mixed-integer algorithms ([14]). Hereafter we will borrow this idea to reduce our controller design problem to a problem of predictive control.

Suppose the objective of control is to force the output y to track a reference signal y_r. Also let X_r, u_r be desired references of the state and input respectively. Then at the current sampling instant k, the predictive control problem can be formulated as:

$$\mathscr{P}(N, X(k)): \min_{\mathcal{U}} \Bigg\{ \|Q_x (X(k+N|k) - X_r)\|_p + \sum_{i=0}^{N-1} \|Q_y (y(k+i|k) - y_r)\|_p$$
$$+ \|Q_x (X(k+i|k) - X_r)\|_p + \|Q_u (u(k+i|k) - u_r)\|_p + |Q_\omega(\omega(k+i|k) - 1)| \Bigg\}$$

subject to $X(k|k) = X(k)$, Eq. (12), and Eqs. (16)-(23), where the positive integer N is the prediction (as well as control) horizon, and

$$\mathcal{U} = \{u(k|k), u(k+1|k), ..., u(k+N-1|k)\},$$

is a future input sequence to be determined by solving the above optimization problem. Furthermore,

$$\|Q_x (X(k+N|k) - X_r(k+N|k))\|_p$$
$$:= \begin{cases} (X(k+N|k) - X_r(k+N|k))^T Q_x (X(k+N|k) - X_r(k+N|k)) & \text{if } p = 2, \\ \|Q_x (X(k+N|k) - X_r(k+N|k))\|_\infty & \text{if } p = \infty. \end{cases}$$

Other matrix norms are defined in a similar way. Weighting matrices satisfy

$$Q_x \geq 0,\ Q_y \geq 0,\ Q_u \geq 0,\ Q_\omega > 0.$$

Because ω reflects the transmission rate, it is separated from state variables. Moreover, Q_ω must be a strictly positive number. The bigger Q_ω is, the more severe the demand is on the network transmission reduction. So consideration of network traffic is integrated into the above optimization *explicitly*.

Here the control and prediction horizon are set equal. As is known in the literature of MPC, an infinite prediction horizon MPC problem is always assumed to be solvable when a finite-horizon MPC problem is to be dealt with. This key observation has enabled many effective approaches to solving various MPC problems (see, e.g., [21], [3]). in fact, If the system involved is of linear structure and its constraints are *convex*, explicit piece-wise affine MPC controllers can be constructed ([3]). Unfortunately, due to the very nature of the problem studied here, i.e., control performance with transmission rate reduction, the corresponding infinite prediction horizon MPC problem (i.e., $N = \infty$ in $\mathscr{P}(N, X(k))$) is *not* solvable. This is stated as the following theorem.

Theorem 1. *If G is unstable, the set of $X(k)$ such that the optimization problem $\mathscr{P}(N, X(k))$ with $N = \infty$ admits finite solutions is of zero Lebesgue measure.*

Proof. For a given $X(k)$, suppose that the optimization problem $\mathscr{P}(N, X(k))$ with $N = \infty$ has a solution. Then it has finite cost. Therefore, there is a time K_0 such that $v(k) = v(K_0)$ for all $k \geq K_0$, i.e., there will be no more new input update. Consequently,

$$\begin{aligned} x(k+1) &= Ax(k) + Bv(k), \\ y(k) &= Cx(k), \ (\forall k \geq K_0), \end{aligned} \tag{24}$$

which yields

$$x(K_0 + L) = A^L x(K_0) + \sum_{i=K_0}^{L-1} A^{L-i} B v(K_0), \tag{25}$$

for any integer $L > 0$. Suppose the problem under consideration is to regulate the state to the origin, namely, drive $x(K_0 + L) \to 0$ as $L \to \infty$. Since G is unstable, i.e., the matrix A is unstable. Denote by E_{K_0} the set of $x(K_0)$ such that $x(K_0+L)$, governed by Eq. (25), tends to zero as L goes to ∞. Then the Lebesgue measure, $m(E_{K_0})$, is zero. Note that K_0 is a non-negative integer. For convenience, denote E_{K_0} by E_i when $K_0 = i$. Thus the union of E_i for i from 0 to ∞ contains all $x(K_0)$ such that $\mathscr{P}(N, X(k))$ with $N = \infty$ is solvable. Observe that $m(E_i) = 0$ for each non-negative integer i, thus

$$m(\cup(E_i)) \leq \sum_{i=0}^{\infty} m(E_i) = 0,$$

which indicates that the set of all $X(K)$ such that $\mathscr{P}(N, X(k))$ with $N = \infty$ is solvable has Lebesgue measure 0. □

Remark 2. Tracking problems can be treated similarly.

Remark 3. Suppose a linear time-invariant controller C is designed for the system G in Fig. 1, it is proved by Theorem 6 in [50] that if either G or C is unstable, the closed-loop system is not asymptotically stable. Theorem 1 shows that asymptotic stability is not possible either even if a time-varying controller is adopted.

Theorem 1 tells us that, as far as the optimization problem at hand is concerned, the finite-horizon and infinite-horizon problems are not "equivalent" in the sense of solvability. This causes much difficulty in proving the stability of the closed-loop system in Fig. 3. Nevertheless, the finite-horizon optimization problem $\mathscr{P}(N, X(k))$ is always feasible for any given $X(k)$. For a given horizon length N, each element $u(k+i|k)$ $(i = 0, ..., N-1)$ in the future input sequence $\mathcal{U}$ either satisfies $|u(k+i|k) - z(k+i|k)| > \delta$ (when δ=1) or $|u(k+i|k) - z(k+i|k)| \leq \delta$ (when δ=0). Hence there are together 2^N optimization problems to be solved at each sampling instant k. The solution of these optimization problems will produce the future input sequence $\mathcal{U}$, the first of which will be sent to the system while the others are to be discarded. This procedure repeats at the next sampling instant $k+1$. However, one can not hope that an optimal input sequence $\mathcal{U}$ can be found, one reason being the nonconvexity of the the constraint $|u(k+i|k) - z(k+i|k)| > \delta$. Moreover, this nonconvexity rules out the existence of piece-wise affine controllers. Nonetheless, in the next section we demonstrate that the method proposed here can be used to design controllers which provide satisfactory control performance and simultaneously reduce network transmission rate to a certain degree.

5. EXAMPLES

In this section, two examples are given to illustrate the effectiveness of the approach presented in Section 4. The first example is used in [51]. It will be shown that better control can be achieved via the propose method. The second example is a double integrator which is an unstable second-order system. The problem of sinusoidal signal tracking is addressed, which can not be addressed using the approach proposed in [51].

Example 1. Consider the following system:

$$\begin{aligned} \Sigma_1 : \ x(k+1) &= ax(k) + bv(k), \\ y(k) &= x(k), \end{aligned}$$

with $v(-1) \in \mathbb{R}$ without loss of generality, and for $k \geq 0$,

$$v(k) = \begin{cases} u(k), & \text{if } |u(k) - v(k-1)| > \delta, \\ v(k-1), & \text{otherwise,} \end{cases}$$

The chaotic dynamics of system Σ_1 have been analyzed in detail in [50, 51, 52]. Now we discuss its control problem. Specifically, we address the following tracking problem.

According to the development in Section 4, we define the following optimization index:

$$\tilde{X}(k) = \begin{bmatrix} x(k) & v(k-1) & \omega(k) \end{bmatrix}^T.$$

$$\min_{\mathcal{U}} \sum_{i=0}^{N} |Q_y (y(k+i|k) - y_r(k+i))| + \|Q_x (x(k+i|k) - x_r(k+i))\|_\infty$$

where

$$y_r \equiv 1, \; x_r = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T.$$

$$Q_y = 110, \; Q_x = \begin{bmatrix} 100 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Note that p in Eq. (24) is chosen to be ∞. In terms of propositional logic, system Σ_1 can be written into a HYSDEL (Hybrid System DEscription Language) code [5]. Then by running this code a mixed logical dynamical model is obtained (However it can not be converted to a piece-wise affine form, due to the nonconvexity of the constraints). Choose $\delta = 0.04$ and $N = 3$. Take an initial condition $(x(0), v(-1)) = (0.5, -10)$. Now we study two cases. Case 1 is with $a = 0.9$ and $b = -0.3$ and case 2 is with $a = 1.2$ and $b = -0.3$. Hence, the original system in case 1 is stable while that in case 2 is not. If we desire that the output y tracks a sinusoidal signal, $0.2 * sin((0 : Tstop)'/5)$, where $Tstop$ is the simulation time (it is set to be 150 in this example), then Figs. 4-7 are obtained. In case 1, the transmission

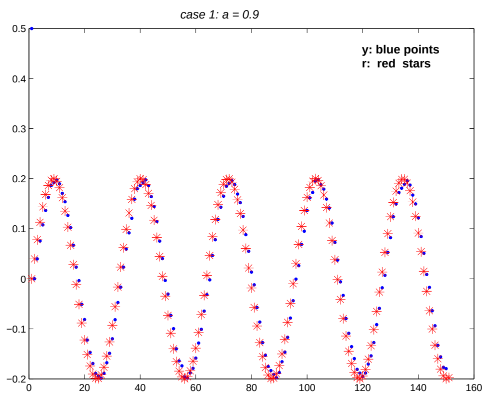

FIG. 4. Tracking of a sinusoidal signal: Output

rate is 66.67% while that of case 2 is 72%. We conclude that tracking is achieved while the network transmission rate is also reduced to a certain degree. When the system is excessively unstable, for example, $a = 20$, the optimization process with a prediction horizon $N = 2$ generates a sequence of input $u(k+i|k)$ which makes all $\omega(k+i+1|k)$ equal to 0. Note that the zero value of ω indicates the successful network transmission. By using a longer prediction horizon, say, $N = 8$, some values of ω are 1, i.e., the controller based on the prediction control can still reduce

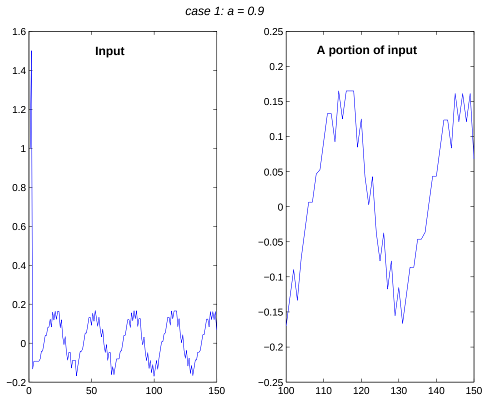

FIG. 5. Tracking of a sinusoidal signal: Input

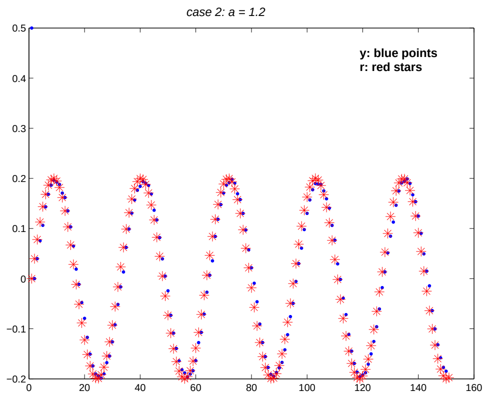

FIG. 6. Tracking of a sinusoidal signal: Output

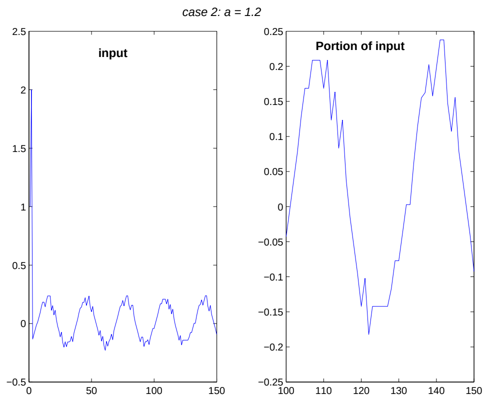

FIG. 7. Tracking of a sinusoidal signal: Input

network traffic to some extent. However, when $a = 50$, the prediction horizon N will have to be extremely big.

Example 2. Consider the double integrator

$$y(s) = \frac{1}{s^2} u(s).$$

Application of the Euler method to it with the sampling period $h = 1s$ yields the following discrete-time system:

$$\Sigma_2 : \; x(k+1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k),$$
$$y(k) = [\; 1 \quad 0 \;] x(k).$$

According to the development in Section 4, we define the following optimization index:

$$\min_{\mathcal{U}} \sum_{i=0}^{N} |Q_y (y(k+i|k) - y_r(k+i))| + \|Q_x (x(k+i|k) - x_r(k+i))\|_\infty$$

where

$$Q_y = 100, \; Q_x = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Following the procedure in Example1, choose $\delta = 0.04$ and $N = 12$. Take an initial condition $(x_1(0), x_2(0), v(-1)) = (1, 1, 0)$. The control objective is to force the output y to track a sinusoidal signal, $0.5 * sin((0 : Tstop)'/5)$ where the simulation time *Tstop* is 400s. Simulation result is shown in Figs. 8-9. Here only the first 360 iterations are plotted. The network transmission rate is 81.75%.

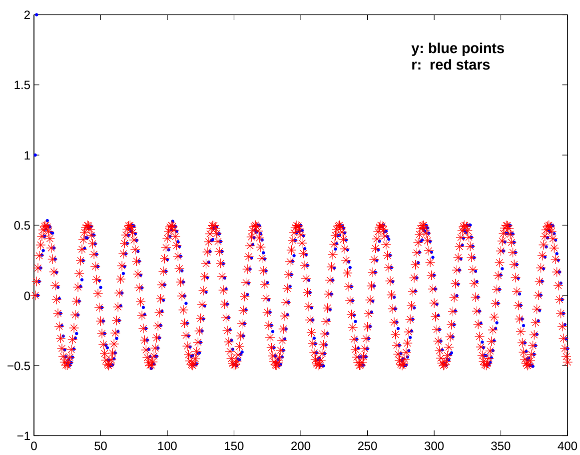

Fig. 8. Tracking of a sinusoidal signal: Output

Some comments may be appropriate.

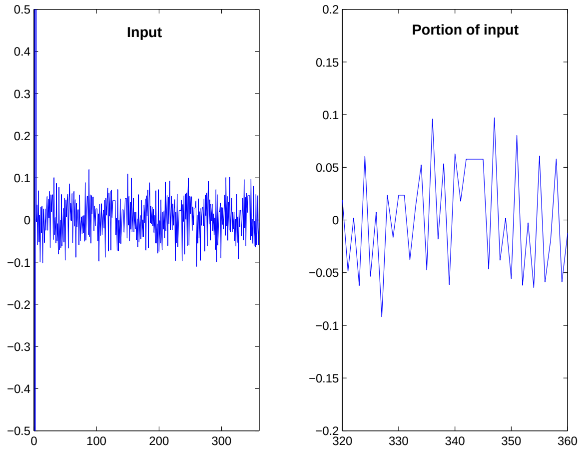

FIG. 9. Tracking of a sinusoidal signal: Input

Remark 4. Notice that the input y is one step behind the reference input because the sinusoidal signals are not known advance. Hence each trajectory y in Fig. 8 are shifted one step to the left when plotted.

Remark 5. In Example 2, if there are no uncertainties, no matter whether asymptotic stability or step tracking is concerned, it is found that there is always a time K_0 such that $x(K_0 + 1) = 0$ (in asymptotic stability) or $y(K_0 + 1) = d$ (in tracking a step of magnitude d), and there are no more transmission anymore. So deadbeat control is achieved.

Remark 6. In Example 2, the choice of the prediction horizon N is crucial. For example, if the problem involved is to track a sinusoidal signal, a small N, say, $N = 2$, leads to no solution (note that $N = 12$ in Example 2). However, a larger N will demand more computation time.

6. CONCLUSIONS

We have studied a network-based control problem for a newly proposed network data transmission scheme. By re-formulating the system into a mixed logical dynamical system, we are able to use some recently developed optimization tools to achieve desired control performance while reducing network traffic simultaneously. Two examples have demonstrated the effectiveness of this treatment. Clearly, many open problems remain to be addressed. Here we collect three of them.

(1) In this paper, systems studied in the examples are low dimensional and SISO. Given a multi-input-multi-output (MIMO) system, the choice of δ in the switching law and the prediction horizon N will have to be designed carefully.

(2) In the two examples in Section 5, it is found that when $p = 2$ in Eq. (24), there exist no solutions to the pertinent optimization problem. The reason is still unknown at this stage of this research.

(3) How to extend the method proposed here to the networked control problem with the presence of both H_1 and H_2? If H_2 is added, there will be time delays from sensor to controller. To make good use of the MPC method, a suitable estimation of the state of the plant becomes very important.

These constitute our future research.

APPENDIX

The mixed logical dynamical system model of Example 1 in Section 5.

```
/*  x1 is the real state x, and x2 is v(k-1). Each time x2 and u are
compared to determine the value of x3. If x3=0, there is one
transmission, otherwise, there is no transmission. */

SYSTEM ncs {

 INTERFACE {
    STATE { /* States can only be REAL or BOOL, not INTEGER,
                    the same holds for input and output. */
            REAL x1 [-100,100];
            REAL x2 [-100,100];
            BOOL x3; }
    INPUT {  REAL u [-100,100]; }
    OUTPUT { REAL y; }
    PARAMETER { REAL a = 1.2, b = -0.3, delta = 0.04;
               }
      }

  IMPLEMENTATION {
       AUX { REAL z1, z2;
             BOOL alpha, beta;
              }

       AD {alpha = u-x2>=delta;  /* delta has a very significant effect
                                   on the transmission rate. */
           beta = u-x2<=-delta;}

       DA { z1 = {IF alpha|beta THEN a*x1+b*u ELSE a*x1+b*x2};
            z2 = {IF alpha|beta THEN u ELSE x2};
           }
```

```
        CONTINUOUS { x1 = z1;
                     x2 = z2; }
 AUTOMATA {
x3 = ~(alpha|beta); }

OUTPUT { y = x1; }
    }

}
```

References

[1] M. Alamir, and G. Bornard, Stability of a truncated infinite constrained receding horizon scheme: The general discrete nonlinear case, Automatica, 31 (1995) 1353-1356.

[2] A. Bemporad and M. Morari, Control of systems integrating logic, dynamics, and constraints, Automatica, 35 (1999) 407-427.

[3] A. Bemporad, M. Morari, V. Dua and E.N. Pistikopoulos, The explicit linear quadratic regulator for constrained systems, Automatica, 38 (2002) 3-20.

[4] A. Bemporad, Efficient conversion of mixed logical dynamical systems into an equivalent piecewise affine form, IEEE, Trans. Automat. Contr., 49 (2004) 832-838.

[5] HYSDEL: Hybrid Toolbox for Matlab, http://www.dii.unisi.it/∼bemporad/.

[6] S. Bououden, O. Benelmir, S. Ziani and S. Filali, A new adaptive fuzzy model and output terminal constraints in predictive control, International Journal of Information and Systems Sciences, 3 (2007) 25-35.

[7] T. Cavalier, P. Pardalos, and A. Soyster, Modeling and integer programming techniques applied to propositional calculus, Comput. Oper. Res., 17 (1990) 561-570.

[8] A. Cervin, D. Henriksson, B. Lincoln, J. Eker, and K. Arzén, How does control timing affect performance? IEEE Control Systems Magazine, 23 (2003) 16-30.

[9] F.J. Christophersen, M. Baotić, and M. Morari, Stability analysis of hybrid systems with a linear performance index, Proc. Conference on Decision and Control, (2004) 4589-4594.

[10] F.J. Christophersen, M. Baotić, and M. Morari, Optimal control of piecewise affine systems: a dynamic programming approach, Lecture Notes in Control and Information Sciences: Control and Observer Design for Nonlinear Finite and Infinite Dimensional Systems, Editors: T. Meurer, et al., (2005) 183-198.

[11] N. Elia, When Bode meets Shannon: control-oriented feedback communication schemes, IEEE Trans. Automat. Contr., 49 (2004) 1477-1488.

[12] E.G. Gilbert and K.T. Tan, Linear systems with state and control constraints: the theory and application of maximal output admissible sets, IEEE Trans. Automat. Contr., 36 (1991) 1008-1020.

[13] M. Feigenbaum, The transition to chaos, Chaos: A New Sciences, the 26th Nobel Conference, (1990) 45-53.

[14] R. Fletcher and S. Leyffer, Numerical experience with lower bounds for MIQP branch-and-bound, SIAM J. on Optimization, 8 (1998) 604-616.

[15] G. Goodwin, H. Haimovich, D. Quevedo, and J. Welsh, A moving horizon approach to networked control system design, IEEE Trans. Automat. Contr., 49 (2004) 1427-1445.

[16] S.S. Keerthi, and E.G. Gilbert, Optimal infinite-horizon feedback laws for a general class of constrained discrete time systems: Stability and moving-horizon approximations, J. of Optimization Theory and Application, 57 (1988) 265-293.

[17] H. Khalil, Nonlinear Systems, Prentice Hall (Second Edition), 1996.

[18] R. Krtolica, Ü. Özgüner, D. Chan, G. Göktas, J. Winkelman, and M. Liubakka, Stability of linear feedback systems with random communication delays, Int. J. Control, 59 (1994) 925-953.

[19] M. Lazar, W.P.M.H. Heemels, S. Weiland, and A. Bemporad, Stabilization Conditions for Model Predictive Control of Constrained PWA Systems, Proc. Conference on Decision and Control, (2004) 4595-4600.

[20] A. Malinowski, T. Booth, S. Grady, and B. Huggins, Real time control of a robotic manipulator via unreliable internet connection, Proc. IEEE, conf. Indus. Elect. Society, IECON'01, (2001) 170-175.

[21] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, Constrained model predictive control: Stability and optimality, Automatica, 36 (2000) 789-814.

[22] E.S. Meadows, M.A., Henson, J.W. Eaton, and J.B. Rawlings, Receding horizon control and discontinuous state feedback stabilization, Int. J. of Control, 62 (1995) 1217-1229.

[23] R. Murray, K. Åström, S. Boyd, R. Brockett, and G. Stein, Future directions in control in an information-rich world, IEEE Control Systems Magazine, 23 (2003) 20-33.

[24] D. Nesic and A. R. Teel, Input output stability properties of networked control systems, IEEE Trans. Automat. Contr., 49 (2004) 1650-1667.

[25] D. Nesic and A.R. Teel, Input to state stability of networked control systems, Automatica, 40 (2004) 2121-2128.

[26] J. Nilsson, B. Bernhardsson and B. Wittenmark, Stochastic analysis and control of real-time systems with random time delays, Automatica, 34 (1998) 57-64.

[27] P. Otanez, J. Moyne, and D. Tilbury, Using deadbands to reduce communication in networked control systems, Proc. American Control Conference, (2002) 3015-3020.

[28] U. Özgüner, H. Goktas, and H. Chan, Automotive suspension control through a computer communication network, Proc. IEEE, conf. Control Applications, (1992) 895-900.

[29] L. Peterson and B. Davie, Computer Networks: A Systems Approach, Morgan Kaufmann Publishers (2nd Edition), 2000.

[30] Z. Qiu, Q. Zhang, X. Zhang and L. Yang, Robust stability for a class of networked control systems based on state observer, International Journal of Information and Systems Sciences, 3 (2007) 594-603.

[31] R. Raji, Smart networks for control, IEEE Spectrum, 31 (1994) 49-53.

[32] J.B. Rawlings, and K.R. Muske, Stability of constrained receding horizon control, IEEE Trans. Automat. Contr., 38 (1993) 1512-1516.

[33] A. Ray, Performance evaluation of medium access control protocols for distributed digital avionics, ASME Journal of Dynamic Systems, Measurement and Control, 109 (1987) 370-377.

[34] D. Rao, M. Murthy, S. Rao and D. Harshal, Neural generalized predictive control for real time applications, International Journal of Information and Systems Sciences, 3 (2007) 1-15.

[35] R. Safaric, K. Jezernik, D. Calkin, and R. Parkin, Telerobot control via internet, Proceedings of IEEE sympo. Indus. Elect., 1 (1999) 298-303.

[36] J. Sparks, Low-cost technologies for aerospace applications, Microprocessors and Microsystems, 20 (1997) 449-454.

[37] S. Tatikonda and S. Mitter, Control under communication constraints, IEEE Trans. Automat. Contr., 49 (2004) 1056-1068.

[38] S. Tatikonda, A. Sahai, and S. Mitter, Stochastic linear control over a communication channel, IEEE Trans. Automat. Contr., 49 (2004) 1549-1561.

[39] F.D. Torrisi and A. Bemporad, HYSDEL - A tool for generating computational hybrid models, IEEE Trans. Contr. Syst. Techno., 12 (2004) 235-249.

[40] J. Walrand and P. Varaiya, High Performance Communication Networks, Morgan Kaufmann Publishers, 1996.

[41] G. Walsh, O. Beldiman, and L. Bushnell, Error encoding algorithms for networked control systems, Proc. IEEE, conf. Decision and Control, (1999) 4933-4938.

[42] G. Walsh, O. Beldiman, and L. Bushnell, Asymptotic behavior of nonlinear networked control systems, IEEE, Trans. Automat. Contr., 46 (2001) 1093 -1097.

[43] G. Walsh and H. Ye, Scheduling of networked control systems, IEEE Control Systems Magazine, 21 (2001) 57-65.

[44] G. Walsh, Y. Hong, and L. Bushnell, Stability analysis of networked control systems, IEEE Trans. Contr. Syst. Techno., 10 (2002) 438-446.

[45] G. Walsh, Y. Hong, and L. Bushnell, Error encoding algorithms for networked control systems, Automatica, 38 (2002) 261-267.

[46] H.P. Williams, Model Building In Mathematical Programming (3rd ed.), New York: Wiley, 1993.

[47] W. Wong and R. Brockett, Systems with finite communication bandwidth constraints, part II: stabilization with limited information feedback, IEEE Trans. Automat. Contr., 44 (1999) 1049-1053.

[48] D. Yue, Q. Han, and J. Lam, Network-based robust H_∞ control of systems with uncertainty, Automatica, 41 (2005) 999-1007.

[49] L. Zhang, Y. Shi, T. Chen and B. Huang, A new method for stabilization of networked control systems with random delays, IEEE Trans. Automat. Contr., 50 (2005) 1177-1181.

[50] G. Zhang and T. Chen, Networked control systems: a perspective from chaos, Int. J. of Bifurcation and Chaos, 15 (2005) 3075-3101.

[51] G. Zhang, G. Chen, T. Chen, and M. D'Amico, Dynamical analysis of a networked control system, Int. J. of Bifurcation and Chaos, 17 (2007) 61-83.

[52] G. Zhang, G. Chen, T. Chen, and Y. Lin, Analysis of a type of nonsmooth dynamical systems, Chaos, Solitons & Fractals, 30 (2006) 1153-1164.

[53] G. Zhang, Dynamical analysis and synchronization of a chaotic networked system, the 3rd National Conference on Complex Dynamical Networks, Beijing, (2006) 119-134.

[54] G. Zhang, X. Chen, and T. Chen, Performance comparison of digital implementation of analog systems, to appear at the 46th Conference on Decision and Control, New Orleans, December 12-14, 2007.

[55] Y. J. Zhai, D. Yu and K. Wang, Comparison of single-dimensional and multi-dimensional optimization approaches in adaptive model predictive control for air-fuel ratio of SI engines, International Journal of Information and Systems Sciences, 3 (2007) 129-149.

College of Electronic Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan, China 610054.

E-mail: `gfzhang@ee.uestc.edu.cn`

Department of Electrical and Computer Engineering, University of Windsor, Windsor, Ontario, Canada N9B 3P4.

E-mail: `xchen@uwindsor.ca`

Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Alberta, Canada T6G 2V4.

E-mail: `tchen@ece.ualberta.ca`