Model Predictive Control over Delay-Based Differentiated Services Control Networks

Riccardo Muradore, Davide Quaglia and Paolo Fiorini
Department of Computer Science, University of Verona, Italy
Emails: {riccardo.muradore,davide.quaglia,paolo.fiorini}@univr.it

***Abstract*—Networked control systems are a well-known sub-set of cyber-physical systems in which the plant is controlled by sending commands through a digital packet-based network. Current control networks provide advanced channel access mechanisms to guarantee low delay on a limited fraction of packets (low-delay class) while the other packets (un-protected class) experience a higher delay which increases with channel utilization. We investigate the extension of model predictive control to choose both the command value and its assignment to one of the two classes according to the predicted state of the plant and the knowledge of network condition. Experimental results show that more commands are assigned to the low-delay class when either the tracking error is high or the network condition is bad.**

I. Introduction

Networked Control Systems (NCS) are well-known sub-set of cyber-physical systems in which the plant is controlled by sending commands through a digital packet-based network. In the past decade a lot of work was done for the correct design of such systems [7], [2]. In particular, the presence of a lossy channel has been addressed [6], [15], [16].

Another important issue when dealing with digital packet-based networks is delay [10] which reduces control promptness and may lead to system instability. Even if many current applications, e.g., automotive networks, are quite reliable from the loss perspective, delay is always an issue when different controller-sensor-actuator groups share the same channel. In [4], a new stochastic control approach for nonlinear network-induced time delay systems is presented by combining on-line reset control, neural networks, and dynamic Bayesian networks. The work in [19] investigates the problem of H_∞ control for a class of networked control systems with time-varying delay in both forward and backward channels. In [17] a new cooperative medium access control is proposed to handle delay issues in the context of wireless local area networks. The work in [18] deals with the problem of guaranteed cost control for a class of uncertain networked control systems with time-varying delay. An improved predictive controller design strategy is proposed to compensate for the delay and data dropout in both the forward and backward channels to achieve the desired control performance.

An interesting way to handle communication issues in distributed applications is relying on network mechanisms which provide quality-of-service guarantees. Traditionally these mechanisms are based on a smarter allocation of network resources so that a sub-set of the packets receive a premium service at the cost of less performance for the rest of the packets. This approach is well-known as *Differentiated Services model* in the TCP/IP community [13] but a similar mechanism is present in WLANs [14] and FlexRay control network [1]. For instance, FlexRay provides two channel access modes, a time-division multiple access mode with guaranteed access delay and an event-driven statistical access mode with traffic-dependent delay.

978-3-9815370-0-0/DATE13/©2013 EDAA

While the use of Differentiated Services model for NCS has been applied in the context of lossy differentiation [11], [12] we are interested in investigating the more realistic scenario in which service differentiation leads to different delay values in packet delivery with the corresponding impact on control performance; in particular, the solution approach in [12] should be deeply changed moving from lossy differentiation to delay differentiation. In [5], control packets are assigned to one of the two transmission modes provided by FlexRay according to the system state (transient or steady).

In this paper we theoretically analyze the design of a model predictive control (MPC) over a Differentiated Services network which provides two different "virtual wires" featuring different constant delays. The MPC approach is extended to choose both the command value and its assignment to one of the two service classes according to the predicted state of the plant and the knowledge of network condition. In fact, when the plant behavior is far from the desired one then a more urgent reaction is needed; vice-versa, the controller should avoid the waste of the more expensive low-delay bandwidth if the delay featured by the un-protected class still guarantees an acceptable control performance. In general, the value of delay of these two classes is not constant since it may depend on their utilization; then such values can be estimated by the controller through network statistics. This work is a first step on MPC for delay-based Differentiated Services and thus the proposed analytical framework does not allow to change delay assumptions at run-time. However different simulations have been performed with different delay values to show the behavior when network condition changes.

The paper is organized as follows. Section II introduces MPC and the control problem is stated with reference to the presence of delay-based service differentiation. Section III describes the solution based on the mixed-integer quadratic programming. Simulation results are reported in Section IV, while conclusions are drawn in Section V.

II. Problem Statement

In this work we assume that the plant is a stable linear time-invariant discrete-time system

$$P(z) : \begin{cases} x_{k+1} = Ax_k + Bu_k \\ y_k = x_k \end{cases} \tag{1}$$

where u_k is the command and the measurement is the state vector (i.e. full-information case). The sample time of the system is denoted by T_s. The plant is controlled by $C(z)$ located on the other side of a packet-based communication network as shown in Figure 1.

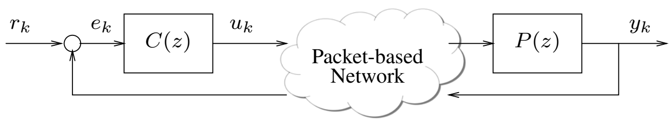

Fig. 1. Block diagram of a networked control system.

In the present setup, the network is designed in such a way that provides advanced channel access mechanisms to guarantee low delay on a limited fraction of packets (low-delay class, H service) while the other packets (un-protected class, L service) experience a higher delay which increases with channel utilization. These differentiated services can be modeled as two "virtual wires" from plant to controller

- H service: $k_H T_s$ delay
- L service: $k_L T_s$ delay

with $k_L > k_H$ integer numbers.

It is well know that the MPC controller for linear plant such as (1) has to minimize a quadratic cost function $J_{MPC}(\cdot)$ defined as

$$J_{MPC}(k) = \sum_{i=0}^{N_p} \|\hat{x}_{k+i|k}\|^2_{Q_i} + \sum_{i=0}^{N_c} \|\hat{u}_{k+i|k}\|^2_{R_i} \tag{2}$$

where $Q_i > 0$, $R_i > 0$ are weighting matrices, N_p is the prediction horizon, N_c is the control horizon, and $\hat{x}_{k+i|k}$ and $\hat{u}_{k+i|k}$ are the i-ahead predictors of the state and of the command, respectively.

At each sample time the optimal control $u_k^\star$ is determined by going through the following steps: ([9]):

1) acquire the value of the current state x_k,
2) solve the constrained optimization problem

$$\begin{aligned} \{\hat{u}^\star(k+\cdot|k)\} = \arg \min_{\{\hat{u}(k+\cdot|k)\}} \quad & J_{MPC}(k) \\ \text{subject to:} \quad & m_u \le \hat{u}_{k+\cdot|k} \le M_u, \\ & m_x \le \hat{x}_{k+\cdot|k} \le \bar{M}_x, \end{aligned}$$

where $\{m_u, M_u\}$ and $\{m_x, M_x\}$ are bounds for inputs and states,
3) set $u_k^\star = \hat{u}^\star_{k+0|k}$.

The goal of this work is to extend the MPC paradigm to take into account the differentiated services by modeling the choice between the two different services.

III. MPC over Delay-Based Differentiated Services Networks

Figure 2 shows the block diagram of a NCS with a two-classes Differentiated Services network: the data paths from controller to plant and vice-versa can be modeled as two "virtual wires" representing the forwarding services, i.e., H and L, characterized by different delays. In the following we will focus only on the forward path, i.e. on the assignment of the logical variable

$$\delta_k = \begin{cases} 1, & \text{if } H \text{ service is selected} \\ 0, & \text{if } L \text{ service is selected} \end{cases} \tag{3}$$

besides the command value. The new MPC formulation aims at jointly selecting the optimal command $u_k^\star$ and the optimal transmission strategy $\delta_k^\star$.

To make the exposition easier to follow, we assume here that $k_H = 0$ and $k_L = 1$ on the controller-to-plant path whereas the delay from plant to controller is not taken into account. The generalization of the equations below is quite trivial even if "cumbersome" in terms of space (see Remark 1 at the end of the section). However, the simulation results reported in the next section will cover the general case.

The joint modeling of the system and of the controller-to-plant virtual wires is

$$\begin{cases} x_{k+1} = Ax_k + (1-\delta_k)Bu_{k-1} + \delta_k Bu_k \\ y_k = x_k. \end{cases} \tag{4}$$

This means that when $\delta_k = 0$ the plant experiences one step delay (since $k_L = 1$) whereas when $\delta_k = 1$ the plant experiences no delay (since $k_H = 0$) and the command is immediately applied.

Enhancing the state vector to take into account the delay we have the following equivalent representation

$$\begin{aligned} \begin{bmatrix} x_{k+1} \\ u_k \end{bmatrix} &= \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_k \\ u_{k-1} \end{bmatrix} + \begin{bmatrix} -B \\ 0 \end{bmatrix} \delta_k u_{k-1} + \\ &+ \begin{bmatrix} B \\ -I \end{bmatrix} \delta_k u_k + \begin{bmatrix} 0 \\ I \end{bmatrix} u_k \end{aligned} \tag{5}$$

By defining the new state vector $\mathbf{x}_k$ and the auxiliary variable as

$$\mathbf{x}_k := \begin{bmatrix} x_k \\ u_{k-1} \end{bmatrix}, \qquad a_k := \delta_k \begin{bmatrix} \mathbf{x}_k \\ u_k \end{bmatrix}$$

the model can be re-written in a more compact way as

$$\mathbf{x}_{k+1} = \underbrace{\begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}}_{F} \mathbf{x}_k + \underbrace{\begin{bmatrix} 0 & -B & B \\ 0 & 0 & -I \end{bmatrix}}_{G_a} a_k + \underbrace{\begin{bmatrix} 0 \\ I \end{bmatrix}}_{G_u} u_k \tag{6}$$

According to [3], the auxiliary variable behaves as expected (i.e. $a_k = \begin{bmatrix} \mathbf{x}_k \\ u_k \end{bmatrix}$ iff $\delta_k = 1$, $a_k = 0$ iff $\delta_k = 0$) if the following

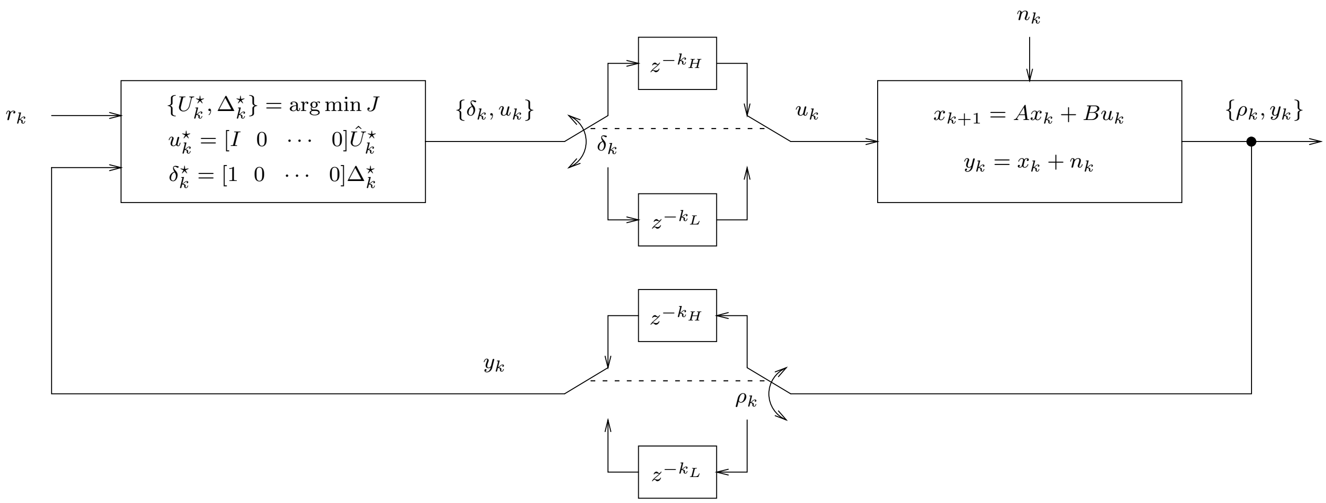

Fig. 2. Block diagram of a NCS with Differentiated Services network.

set of inequalities are satisfied

$$\begin{aligned} a_k &\leq M\delta_k \\ a_k &\geq m\delta_k \\ a_k &\leq \begin{bmatrix} \mathbf{x}_k \\ u_k \end{bmatrix} - m(1-\delta_k) \\ a_k &\geq \begin{bmatrix} \mathbf{x}_k \\ u_k \end{bmatrix} - M(1-\delta_k) \end{aligned}$$

where M and m are the component-wise maximum and minimum of $\begin{bmatrix} \mathbf{x}_k \\ u_k \end{bmatrix}$. The above set of inequalities can be re-written as a matrix inequality as follows

$$\begin{bmatrix} I \\ -I \\ I \\ -I \end{bmatrix} a_k \leq \begin{bmatrix} M \\ -m \\ m \\ -M \end{bmatrix} \delta_k + \begin{bmatrix} 0 \\ 0 \\ I \\ -I \end{bmatrix} \begin{bmatrix} \mathbf{x}_k \\ u_k \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -m \\ M \end{bmatrix}$$

or, equivalently, by separating $\mathbf{x}_k$ and u_k, as

$$\begin{aligned} \begin{bmatrix} I \\ -I \\ I \\ -I \end{bmatrix} a_k \quad \leq \quad & \begin{bmatrix} M \\ -m \\ m \\ -M \end{bmatrix} \delta_k + \begin{bmatrix} 0 \\ 0 \\ I \\ 0 \\ -I \\ 0 \end{bmatrix} \mathbf{x}_k + \\ & + \begin{bmatrix} 0 \\ 0 \\ 0 \\ I \\ 0 \\ -I \end{bmatrix} u_k + \begin{bmatrix} 0 \\ 0 \\ -m \\ M \end{bmatrix} \end{aligned}$$

where Is are the identity matrices of proper dimension. This inequality is a constraint of the following type

$$E_a a_k \quad \leq \quad E_\delta \delta_k + E_{\mathbf{x}} \mathbf{x}_k + E_u u_k + E, \qquad \forall k \quad (7)$$

and has to be satisfied for each k. It is worth remarking that the state equation (6) with the measurement equation

$$y_k = x_k = \begin{bmatrix} I & 0 \end{bmatrix} \mathbf{x}_k \qquad (8)$$

and with constraints like (7) belongs to the class of mixed logical dynamical systems [3], i.e. systems of the form

$$\begin{cases} x_{k+1} &= Ax_k + B_u u_k + B_\delta \delta_k + B_a a_k \\ y_k &= Cx_k + D_u u_k + D_\delta \delta_k + D_a a_k \\ E &\leq E_x x_k + E_u u_k + E_\delta \delta_k + E_a a_k. \end{cases}$$

The standard MPC problem recalled briefly in Section II takes the following form when the differentiated services (with the corresponding auxiliary and logical variables) are taken into account

$$\begin{aligned} \min_{\hat{u}_{k+\cdot|k}, \delta_{k+\cdot}} \quad & J_{hMPC}(k) = \sum_{i=0}^{N} \Big[\|\hat{\mathbf{x}}_{k+i|k}\|^2_{Q_i} + \|\hat{u}_{k+i|k}\|^2_{R_i} + \\ & \qquad + \|\hat{a}_{k+i|k}\|^2_{H_i} + \|\hat{\delta}_k\|^2_{W_i} \Big] \\ \text{s. to} \quad & \mathbf{x}_{k+1} = F\mathbf{x}_k + G_u u_k + G_a a_k \\ & y_k = \mathbf{x}_k \\ & \hat{\delta}_k \in \{0, 1\} \\ & E_a a_k \leq E_\delta \hat{\delta}_k + E_{\mathbf{x}} \mathbf{x}_k + E_u u_k + E \end{aligned}$$

where H_i and W_i are matrices weighting the auxiliary variables and the logical variables, respectively. Without such matrices that algorithm should always select the class with smaller delay which guarantees better performance. With reference to (2) we assume $N_p = N_c = N$ without loss of generality.

We call the new index $J_{hMPC}(k)$ because the plant and the network can be seen as a hybrid system with two locations (or modes) corresponding to the two different network models, i.e. z^{-k_H} and z^{-k_L} (see [8] for a complete hybrid modeling of the network). The change of location is in charge of the controller. In this sense our formulation of the MPC can be seen as the composition of a state-feedback controller (i.e. u_k is a function of the state vector) and a supervisor controller (i.e. δ_k is a signal which changes the location as a function of the expected performance $J_{hMPC}(k)$).

The previous problem can be solve by reformulate it as mixed integer quadratic programming (MIQP) problem. We define the following input-output data matrices

$$\hat{X}_k = \begin{bmatrix} \mathbf{x}_k \\ \hat{\mathbf{x}}_{k+1|k} \\ \vdots \\ \hat{\mathbf{x}}(k+N|k) \end{bmatrix}, \qquad \hat{U}_k = \begin{bmatrix} \hat{u}_{k|k} \\ \hat{u}_{k+1|k} \\ \vdots \\ \hat{u}_{k+N-1|k} \end{bmatrix},$$

$$\hat{\Delta}_k = \begin{bmatrix} \delta_{k|k} \\ \delta_{k+1|k} \\ \vdots \\ \delta_{k+N-1|k} \end{bmatrix}, \qquad \hat{A}_k = \begin{bmatrix} \hat{a}_{k|k} \\ \hat{a}_{k+1|k} \\ \vdots \\ \hat{a}_{k+N-1|k} \end{bmatrix}.$$

The state equation has the matrix-like representation

$$\begin{aligned} \hat{X}_k &= \mathcal{F}\mathbf{x}_k + \mathcal{B}_u \hat{U}_k + \mathcal{B}_a \hat{A}_k \\ &= \mathcal{F}\mathbf{x}_k + \begin{bmatrix} \mathcal{B}_u & \mathcal{B}_a & 0 \end{bmatrix} \begin{bmatrix} \hat{U}_k \\ \hat{A}_k \\ \hat{\Delta}_k \end{bmatrix} \\ &= \mathcal{K}_k + \mathcal{G} V_k \end{aligned}$$

on the time horizon $[k, k+N]$, whereas the constraint inequality (7) takes the form

$$\begin{aligned} \mathcal{E}_a \hat{A}_k &\leq \mathcal{E}_\delta \hat{\Delta}_k + \mathcal{E}_\mathbf{x} \hat{X}_k + \mathcal{E}_u \hat{U}_k \mathcal{E} \\ \underbrace{\begin{bmatrix} -\mathcal{E}_u & \mathcal{E}_a & -\mathcal{E}_\delta \end{bmatrix}}_{\mathcal{E}_v} V_k &\leq \mathcal{E}_\mathbf{x}(\mathcal{K}_k + \mathcal{G} V_k) + \mathcal{E} \\ (\mathcal{E}_v - \mathcal{E}_\mathbf{x}\mathcal{G}) V_k &\leq \mathcal{E}_\mathbf{x}\mathcal{K}_k + \mathcal{E}. \end{aligned}$$

The vector V_k collects all the unknowns and is computed by solving the mixed integer quadratic programming problem related to the hMPC index, i.e.

$$\begin{aligned} V_k^\star = \quad & \arg\min_{V_k} \quad V_k^T \left[\mathcal{G}^T \mathcal{L}^T \mathcal{Q} \mathcal{L} \mathcal{G} + \mathcal{R}\right] V_k + \\ & \qquad\qquad +2 V_k^T \mathcal{G}^T \mathcal{L}^T \mathcal{Q} \mathcal{L} \mathcal{K} + \mathcal{K}^T \mathcal{Q} \mathcal{K} \\ & \text{s. to} \qquad (\mathcal{E}_v - \mathcal{E}_\mathbf{x}\mathcal{G}) V_k \leq \mathcal{E}_\mathbf{x}\mathcal{K}_k + \mathcal{E} \\ & \qquad\qquad \Delta_k \in \{0,1\}^N. \end{aligned}$$

In [3] this kind of controller is called mixed integer predictive controller and the proof of the its stability (related to the feasibility of the MIQP problem) is also given in that paper.

Let $V_k^\star = \begin{bmatrix} \hat{U}_k^\star \\ \hat{A}_k^\star \\ \hat{\Delta}_k^\star \end{bmatrix}$ be the optimal solution of the MIQP problem at time k, then the optimal command and transmission policy are

$$u_k^\star = \begin{bmatrix} I & 0 & \cdots & 0 \end{bmatrix} \hat{U}_k^\star \tag{9}$$

$$\delta_k^\star = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \hat{\Delta}_k^\star. \tag{10}$$

In other words, $u_k^\star$ is the content of the packet sent through the network, and $\delta_k^\star$ decides the virtual wire to be used to deliver the packet to the plant.

Remark 1. In the general case with $k_L > k_H > 1$, the matrix F in (6) takes the form

$$F = \left[\begin{array}{ccccc|cccc} A & B & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & I & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & I & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 & I & 0 & \cdots & 0 \\ \hline 0 & 0 & 0 & \cdots & 0 & 0 & I & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & I \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \end{array}\right]$$

where the upper-left block models the "common delay", $D = k_H$, whereas the bottom-right block models the "relative delay", $d = k_L - k_H$. In fact the state vector looks like

$$\mathbf{x}_k = \left[\begin{array}{c} x_k \\ u_{k-k_L} \\ u_{k-k_L+1} \\ \vdots \\ u_{k-k_L+k_H} \\ \hline u_{k-k_L+k_H+1} \\ \vdots \\ u_{k-1} \end{array}\right]$$

It is worth remarking that when a command sent on the low-delay class is received, it overcomes other commands traveling on the un-protected class: more precisely, it overcomes $k_L - k_H$ commands. Our policy is to froze the last command whenever older commands are received, i.e. the $k_L - k_H$ commands traveling on the L service are discharged. The matrices G_a, G_u also implement this behavior. ■

Remark 2. The above design assumes that only the delay on the controller-to-plant path can be selected between the two available possibilities. Moreover it is possible to include in the model also the decision variable ρ_k for the plant-to-controller path, see Figure 2. In such a way, the controller can choose the overall loop delay (round trip time) amongst $k_H + k_H$, $k_H + k_L$ and $k_L + k_L$. The controller has also to send to the plant side the optimal value $\varrho_k^\star$ that will be used to select the transmission policy for sending back the measurements. ■

IV. Simulation Results

The plant is given by

$$P(z) : \begin{cases} x_{k+1} = 0.9x_k + 0.1u_k \\ y_k = x_k + n_k + d_k \end{cases} \tag{11}$$

where n_k is a Gaussian noise with zero mean and unitary variance and d_k is a step-wise disturbance. The initial condition is $x_0 = -10$ unknown to the controller. The controller is used to steer the system to zero.

To compare different delay configurations, we adopt constant weightings:

$$Q = 10, \quad R = 0.1, \quad H = 0.1,$$

while for W two values have been tested, i.e. 10 and 20. Moreover, we force the commands to belong to the range $[-25, 25]$ and the optimization horizon N is set equal to 7 steps.

Table I reports the standard deviation of the feedback error and the percentage of use of the low-delay class as a function of the different pairs (k_H, k_L).

k_H	k_L	Std	%
0	2	3.64	25.49
0	4	3.38	37.25
1	3	3.18	39.22
2	3	4.13	50.98
2	4	3.53	60.78

k_H	k_L	Std	%
0	2	3.63	19.61
0	4	3.35	35.29
1	3	3.36	31.37
2	3	4.25	49.02
2	4	4.18	54.90

TABLE I
RESULTS FOR $W = 10$ (LEFT) AND $W = 20$ (RIGHT).

From the data reported in Table I, some conclusions can be drawn:

- increasing W (the network cost to provide service differentiation) implies that a smaller fraction of the packets is assigned to the low-delay class;
- the use of the low-delay class decreases according to the delay difference between the two classes showing that the controller avoids the waste of low-delay bandwidth if the gain in using it is negligible;
- the percentage of the use of the low-delay class is an automatic way to keep the error as constant as possible; for this reason, the case $k_H = 2, k_L = 4$ has slightly better results than the case $k_H = 2, k_L = 3$ because a higher percentage of the low-delay class is used;
- the use of the low-delay class increases according to the average delay thus leading to lower error;
- results from the first two rows show that the control strategy takes into account network condition; in fact, if the un-protected class experiences higher delay (k_L) then a higher fraction of packets is assigned to the low-delay class.

In our experiments we fixed the weightings Q, R and H even though in the final design when the delays k_H and k_L are fixed and known, such values have to be optimized on that particular working condition.
Figures 3 and 4 compare the cases $k_H = 0$, $k_L = 2$ with $W = 10$ and $W = 20$, respectively. It is possible to see that when W increases the number of commands sent on the low-delay class is smaller but located on the most critical interval, i.e. when the disturbance step arrives. This means that some of the commands sent with delay k_H on the first case ($W = 10$) are less critical because the system is already close to the zero error condition and thus their loss can be tolerated.

An interesting remark can be done by comparing Figures 5 and 6. Even though the only difference is in the value of k_H, the response of the system is worse and higher use of the low-delay class is decided by the controller to mitigate the higher "common" delay.

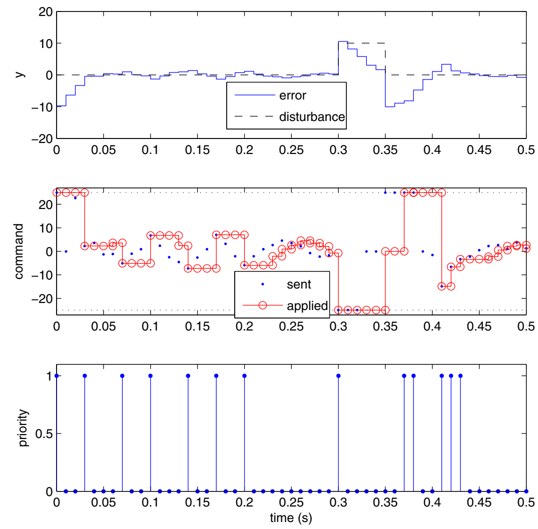

Fig. 3. Simulation results with $k_H = 0$, $k_L = 2$ and $W = 10$.

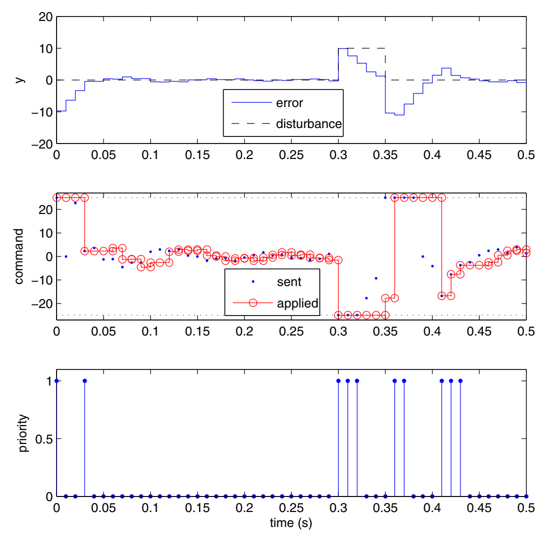

Fig. 4. Simulation results with $k_H = 0$, $k_L = 2$ and $W = 20$.

Another observation is in order about the constrain on u: commands are sometimes set to the maximum or minimum value by the MPC algorithm itself without introducing explicitly a saturation block. A proper design should weight the ratio between Q and W in order to reduce as much as possible long periods where the command is forced to the maximum/minimum value. In this way it is possible to avoid the increase of the settling time.

V. CONCLUSION

We presented a novel MPC design for systems controlled through networks providing delay-based differentiated services. The two services are characterized by different constant

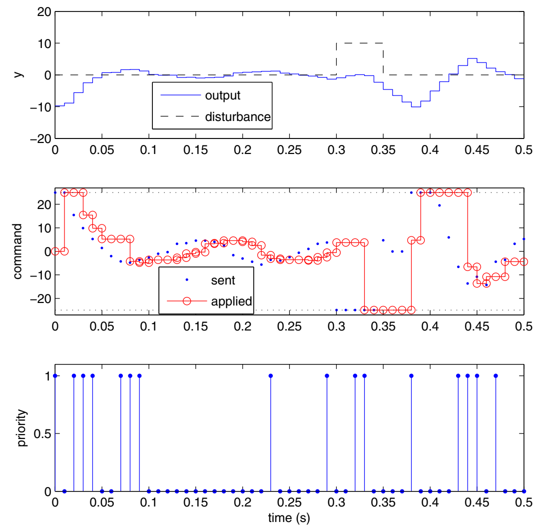

Fig. 5. Simulation results with $k_H = 1$, $k_L = 3$ and $W = 20$.

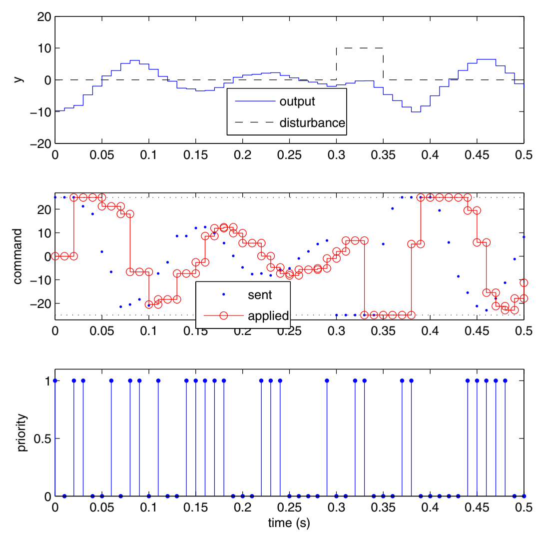

Fig. 6. Simulation results with $k_H = 2$, $k_L = 3$ and $W = 20$.

delays. The overall model of the system (plant and network) is extended by introducing logic variables. The optimal value of the commands and of the logic variables are computed by solving a hybrid version of the MPC performance index. Simulation results show that the proposed control architecture is a suitable approach to jointly decide the control commands and the transmission strategy.

Acknowledgments

This work has been partially supported by the I-SUR FP7 European Integrated Project, funded by the European Commission under Grant Agreement 270396.

References

[1] The FlexRay Communications System Specications, Ver. 2.1. *URL: http://www.exray.com*.

[2] J. Baillieul and P.J. Antsaklis. Control and Communication Challenges in Networked Real-Time Systems. 95(1):9–28, Jan. 2007.

[3] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35:407–428, 1999.

[4] Hyun Cheol Cho and M.S. Fadali. Nonlinear network-induced time delay systems with stochastic learning. *Control Systems Technology, IEEE Transactions on*, 19(4):843–851, July 2011.

[5] D. Goswami, R. Schneider, and S. Chakraborty. Re-engineering Cyber-Physical Control Applications for Hybrid Communication Protocols. In *Proc. IEEE Conf. on Design, Automation and Test in Europe (DATE)*, 2011.

[6] E. Henriksson, H. Sandberg, and K.H. Johansson. Predictive Compensation for Communication Outages in Networked Control Systems. In *Proc. IEEE Conf. on Decision and Control (CDC)*, 2008.

[7] J.P. Hespanha, P. Naghshtabrizi, and Y. Xu. A survey of recent results in networked control systems. *Proceedings of the IEEE*, 95(1):138–162, 2007.

[8] J. Lee, S. Bohacek, J.P. Hespanha, and K. Obraczka. Modeling communication networks with hybrid systems. *IEEE/ACM Transactions on Networking*, 15(3):630–643, 2007.

[9] J.M. Maciejowski. *Predictive control: with constraints*. Pearson education, 2002.

[10] J.R. Moyne and D.M. Tilbury. The Emergence of Industrial Control Networks for Manufacturing Control, Diagnostics, and Safety Data. 95(1):29–47, Jan. 2007.

[11] R. Muradore, D. Quaglia, and P. Fiorini. Adaptive LQ Control over Differentiated Service Lossy Networks. In *World Congress of the International Federation of Automatic Control, (IFAC)*, 2011.

[12] R. Muradore, D. Quaglia, and P. Fiorini. Predictive control of networked control systems over differentiated services lossy networks. In *Proc. IEEE Conf. on Design, Automation and Test in Europe (DATE)*, 2012.

[13] K. Nichols, V. Jacobson, and L. Zhang. A Two-bit Differentiated Services Architecture for the Internet. *RFC 2638*, July 1999.

[14] M.M. Rashid, E. Hossain, and V.K. Bhargava. Controlled channel access scheduling for guaranteed qos in 802.11e-based wlans. *Wireless Communications, IEEE Transactions on*, 7(4):1287–1297, April 2008.

[15] L. Schenato, B. Sinopoli, M. Franceschetti, K. Poolla, and S.S. Sastry. Foundations of control and estimation over lossy networks. *Proceedings of the IEEE*, 95(1):163–187, 2007.

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

[17] A. Ulusoy, O. Gurbuz, and A. Onat. Wireless model-based predictive networked control system over cooperative wireless network. *Industrial Informatics, IEEE Transactions on*, 7(1):41–51, feb. 2011.

[18] Rui Wang, Guo-Ping Liu, Wei Wang, D. Rees, and Y.B. Zhao. Guaranteed cost control for networked control systems based on an improved predictive control method. *Control Systems Technology, IEEE Transactions on*, 18(5):1226–1232, sept. 2010.

[19] Rui Wang, Guo-Ping Liu, Wei Wang, D. Rees, and Yun-Bo Zhao. h_∞ control for networked predictive control systems based on the switched lyapunov function method. *Industrial Electronics, IEEE Transactions on*, 57(10):3565–3571, oct. 2010.