2012

Linear matrix inequality-based robust model predictive control for time-delayed systems

IET CONTROL THEORY AND APPLICATIONS, v. 6, n. 1, pp. 37-50, JAN, 2012

http://www.producao.usp.br/handle/BDPI/34001 www.ietdl.org

Published in IET Control Theory and Applications
Received on 18th April 2011
Revised on 25th July 2011
doi: 10.1049/iet-cta.2011.0216

ISSN 1751-8644

Linear matrix inequality-based robust model predictive control for time-delayed systems

B.D.O. Capron   M.T. Uchiyama   D. Odloak

Department of Chemical Engineering, University of São Paulo, Av. Prof. Luciano Gualberto, n. 380, trav. 3, 05508-900, São Paulo, Brazil
E-mail: odloak@usp.br

Abstract: This work addresses the solution to the problem of robust model predictive control (MPC) of systems with model uncertainty. The case of zone control of multi-variable stable systems with multiple time delays is considered. The usual approach of dealing with this kind of problem is through the inclusion of non-linear cost constraint in the control problem. The control action is then obtained at each sampling time as the solution to a non-linear programming (NLP) problem that for high-order systems can be computationally expensive. Here, the robust MPC problem is formulated as a linear matrix inequality problem that can be solved in real time with a fraction of the computer effort. The proposed approach is compared with the conventional robust MPC and tested through the simulation of a reactor system of the process industry.

1 Introduction

In the process industry, model predictive control (MPC) is the most implemented multi-variable control strategy [1]. This control strategy is particularly interesting when there is a layered control structure and a real-time optimisation (RTO) algorithm lies at the top of this structure. In this case, the RTO defines optimum targets for some of the inputs and/or outputs of the process system [2]. In this case, the MPC controller is designed to work in the optimum target tracking scheme where it should drive the process to the optimum operating point, while maintaining the remaining inputs and outputs inside predefined zones. It is expected that in the target tracking operation, the process will be moved quite often from one operating point to another, and, as the process system is usually non-linear, the linear model, on which the MPC is based, will become uncertain. The robust MPC considered here accounts explicitly for some types of uncertainty in the parameters of the model on which the controller is based. In the multi-variable system with multiple time delays, besides the uncertainty in the process gains and time constants, uncertainty in the time delays may also be significant and should be considered in the robust control problem. The classes of model uncertainty that are most adopted in the robust MPC literature are the polytopic uncertainty [3] and the multi-plant uncertainty [4]. As the polytopic description of model uncertainty is more general than the multi-plant description, it becomes important to clearly define what sort of uncertainty can be tolerated by the robust controller.

In [4], a robust MPC is developed for the multi-plant uncertainty in the regulator case. In that approach, the infinite horizon cost function to be minimised is based on the nominal or most probable model and the controller includes a set of constraints that force the cost of all possible plants to decrease along the control horizon. In [5], the approach is extended to the output tracking case of process systems that can be represented by a finite set of plants. In these methods, stability is achieved by means of the recursive feasibility of the control problem and the cost function corresponding to the true plant becomes a Lyapunov function for the closed-loop system. Recently, the robust MPC was extended to the case of output zone control and input tracking for stable systems with multi-plant uncertainty [6]. In that work, the non-increasing cost constraint strategy is extended to the zone control of the outputs. They also consider that there are targets to be achieved by some of the inputs. Even more recently the approach was extended to the case of input delayed multi-plant systems, by introducing minor modifications in the state-space model and preserving the main structure of the control algorithm [7].

A shortcoming of the robust controller formulation presented in [7] is the complexity of the optimisation problem that is solved to produce the control law. The cost constraints that are introduced in the robust MPC turn the control problem to a non-linear program (NLP) that can be computationally expensive for real-time applications. Then, the computationally inexpensive linear matrix inequality (LMI)-based techniques that have been developed over the last few decades become an interesting option to the solution of robust control problems. An LMI-based robust MPC first appeared in [3] where a linear state feedback controller with infinite control horizon was proposed for polytopic uncertain models and for the case where the system can be represented by a linear model with a feedback uncertainty. The approach has been extended to improve the feasibility and to reduce conservatism in several works, by defining a parameter-dependent Lyapunov function [8, 9], by introducing relaxation matrices in the robust control problem formulation [10] or by introducing linear matrix inequalities as approximations to the robust system constraints [11]. The computer effort was also reduced through the offline solution of a sequence of explicit control laws corresponding to a sequence of asymptotically stable invariant ellipsoids [12]. The method was also extended to become more general by increasing the degrees of freedom of the controller through the consideration of a set of free control inputs along a finite horizon and a linear state feedback control law along the remaining infinite control horizon [13, 14]. Focusing the more practical case, the method was also extended to time-delayed systems [15, 16] and applied to an industrial reactor system [17]. However, the main practical limitation of all the known approaches derived from the robust MPC presented in [3] is that the method has not been extended to the case of output zone control and optimising input targets. This is a practical case that is considered here. The difficulty is mainly associated with the consideration of a linear state feedback control law that cannot be adopted when the output or state is controlled inside a zone instead of at a defined set point and the input may have an optimum target. In this paper, the robust MPC developed in [6, 7] is converted into an LMI problem and applied to systems with multi-plant uncertainty or polytopic uncertainty in some of the model parameters. In the approach considered here, it is assumed that the control horizon is finite while the output horizon is infinite. An algorithm to solve the resulting LMI problem is proposed. The performance and the computational cost of the new formulation are compared to the NLP-based robust MPC.

The paper is organised as follows. In Section 2, the adopted model that represents the system is discussed. In Section 3, the nominal stable MPC problem is formulated and in Section 4, the problem is extended to the multi-model plant and to the polytopic system in order to define the standard robust MPC problem. In Section 5, the problem is converted into an LMI-based problem. The stability of the proposed controller is shown and the convergence of the resulting closed-loop system is discussed. In Section 6, the application of the LMI-based robust MPC to a process system of the control literature is simulated and the performance of this controller is compared to the conventional NLP-based robust MPC algorithm. Finally, in Section 7 the paper is concluded.

2 System representation

Here, it is assumed that the system to be controlled has nu inputs that will be manipulated and ny outputs that need to be controlled inside pre-defined zones. The state-space model considered is an output prediction-oriented model [5] that is here extended to the case of systems with multiple time delays. For this purpose, let $\theta_{i,j}$ be the time delay between input u_j and output y_i and define $p > \max_{i,j}(\theta_{i,j}/T)$ where T is the sampling period. Then, the state-space model considered here can be defined as follows

$$\begin{aligned} x(k+1) &= A\,x(k) + B\,\Delta u(k) \\ y(k) &= C\,x(k) \end{aligned} \tag{1}$$

where

$$x(k) = \begin{bmatrix} y(k)^{\mathrm T} & y(k+1)^{\mathrm T} & \cdots & y(k+p)^{\mathrm T} & x^s(k)^{\mathrm T} & x^d(k)^{\mathrm T} \end{bmatrix}^{\mathrm T}$$

$$A = \begin{bmatrix} 0 & I_{ny} & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & I_{ny} & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & I_{ny} & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & I_{ny} & \Psi((p+1)T) \\ 0 & 0 & 0 & \cdots & 0 & I_{ny} & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & F \end{bmatrix},$$

$$B = \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_p \\ S_{p+1} \\ B^s \\ B^d \end{bmatrix}, \quad C = \begin{bmatrix} I_{ny} & 0 & \cdots & 0 \end{bmatrix} \tag{2}$$

$$x^s \in \Re^{ny}, \quad x^d \in C^{nd}, \quad F \in C^{nd\times nd}, \quad \Psi \in C^{ny\times nd}, \quad I_{ny} = \mathrm{diag}(1\ldots 1) \in \Re^{ny\times ny}$$

In the state vector defined in (1), the first $p+1$ components are associated to the output predictions at future time steps, and it can be shown [5] that x^s corresponds to the predicted output at steady state and x^d are the states corresponding to the stable modes of the system that tend to zero when the system approaches steady state.

When the poles of the system are non-repeated F is a diagonal matrix with components of the form $e^{r_i T}$ where r_i is a pole of the system. In the general case, matrix F is block diagonal. Here, it is assumed that the system has nd stable poles and, in this case, it can be shown [5] that B^s is the gain matrix of the system and $S_1, \ldots, S_{p+1}$ are the step response coefficients of the system. Matrix Ψ is defined as follows

$$\Psi(t) = \begin{bmatrix} \phi_1(t) & 0 & \cdots & 0 \\ 0 & \phi_2(t) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \phi_{ny}(t) \end{bmatrix}$$

where (see equation at the bottom of the page)

$r_{i,j,k}$ with $k = 1, \ldots, na$, are the poles of the transfer function that relates input u_j and output y_i and na is the order of this transfer function. To simplify, one can assume that na is the same for any pair (u_j, y_i). It is clear that the time delay affects the dimension of the state matrix A through parameter p and the components of matrix Ψ.

With this model formulation, it is easy to show that, if the system reaches a steady state with the output at y_{ss}, the system state will stabilise at the following equilibrium point

$$x_{ss}(k) = \begin{bmatrix} y_{ss}^{\mathrm T} & y_{ss}^{\mathrm T} & \cdots & y_{ss}^{\mathrm T} & y_{ss}^{\mathrm T} & 0 \end{bmatrix}^{\mathrm T}$$

$$\phi_i(t) = \begin{bmatrix} e^{r_{i,1,1}(t-\theta_{i,1})} & \ldots & e^{r_{i,1,na}(t-\theta_{i,1})} & \ldots & e^{r_{i,nu,1}(t-\theta_{i,nu})} & \ldots & e^{r_{i,nu,na}(t-\theta_{i,nu})} \end{bmatrix}$$

doi: 10.1049/iet-cta.2011.0216

2.1 Model uncertainty

In the model defined in (1), model uncertainty corresponds to uncertainties in matrices F, B^s, B^d and in the entries of matrix θ that represents the time delays. So, uncertainty in the time delays is also explicitly included in the robust controller presented here. These uncertainties also reflect in uncertainties in the step response coefficients, which appear in the input matrix B defined in (2). There are several ways to represent model uncertainty in model predictive control. The simplest form corresponds to the multi-plant representation [4], where a discrete set Ω of models is available, each model corresponding to a given operating point. In the trivial case, the real plant is unknown but is known to be one of the plants of this set. With this representation, one can define the set of possible plants as $\Omega = \{\Theta_1, \ldots, \Theta_L\}$ where each Θ_n corresponds to a particular plant: $\Theta_n = (F, B^s, B^d, \theta)_n$, $n = 1, \ldots, L$. One can also assume that the true plant is designated as θ_T and that there is a most likely plant that also lies in Ω and is represented as Θ_N.

Another more general form of representing model uncertainty that includes the multi-plant case is the polytopic uncertainty. In this representation, one still considers the set of models Ω, but now the true plant can be any convex combination of the components of Ω that become the vertices of a polytope. Then, in this case, the true plant can be represented as follows

$$\Theta_T = \sum_{i=1}^{L} \lambda_i \Theta_i, \quad \sum_{i=1}^{L} \lambda_i = 1, \quad \lambda_i \ge 0$$

In the development of the proposed LMI-based robust MPC presented here, it will be shown that the controller is robust to polytopic uncertainty in some of the model parameters defined in (2), while, for the other parameters, robustness is only assured for the multi-plant uncertainty. In addition, in the development presented here, it is assumed that the current estimated state corresponds to the true plant.

3 Nominal MPC with zone control and optimising targets

In this section, it is revised the infinite horizon MPC for state-space models in the incremental form as in (1) considering the zone control of the outputs and assuming that there are targets for some of the inputs [18]. This controller is based on the following control objective

$$\begin{aligned} V_k = {} & \sum_{j=0}^{\infty} (y(k+j|k) - y_{sp,k} - \delta_{y,k})^T \\ & \times Q_y (y(k+j|k) - y_{sp,k} - \delta_{y,k}) \\ & + \sum_{j=0}^{\infty} (u(k+j|k) - u_{des,k} - \delta_{u,k})^T \\ & \times Q_u (u(k+j|k) - u_{des,k} - \delta_{u,k}) \\ & + \sum_{j=0}^{m-1} \Delta u(k+j|k)^T R \Delta u(k+j|k) \\ & + \delta_{y,k}^T S_y \delta_{y,k} + \delta_{u,k}^T S_u \delta_{u,k} \end{aligned} \tag{3}$$

where $\Delta u(k+j|k)$ is the control move computed at time k to be applied at time $k+j$, m is the control horizon, Q_y, Q_u, R, S_y, S_u are positive weighting matrices of appropriate dimension, $y_{sp,k}$ and $u_{des,k}$ are, respectively, the output and input targets, $\delta_{y,k}$ and $\delta_{u,k}$ are slack variables that extend the feasibility and attraction domain of the controller to the whole definition set of the states.

The cost defined in (3) can be developed as follows

$$\begin{aligned} V_k = {} & \sum_{j=0}^{p} (y(k+j|k) - y_{sp,k} - \delta_{y,k})^T \\ & \times Q_y (y(k+j|k) - y_{sp,k} - \delta_{y,k}) \\ & + \sum_{j=1}^{\infty} (y(k+p+j|k) - y_{sp,k} - \delta_{y,k})^T \\ & \times Q_y (y(k+p+j|k) - y_{sp,k} - \delta_{y,k}) \\ & + \sum_{j=0}^{\infty} (u(k+j|k) - u_{des,k} - \delta_{u,k})^T \\ & \times Q_u (u(k+j|k) - u_{des,k} - \delta_{u,k}) \\ & + \sum_{j=0}^{m-1} \Delta u(k+j|k)^T R \Delta u(k+j|k) \\ & + \delta_{y,k}^T S_y \delta_{y,k} + \delta_{u,k}^T S_u \delta_{u,k} \end{aligned} \tag{4}$$

The first term on the right-hand side of (4) can also be expressed as follows

$$V_{k,1} = (\tilde{y}_k - \tilde{I}_y y_{sp,k} - \tilde{I}_y \delta_{y,k})^T \tilde{Q}_y (\tilde{y}_k - \tilde{I}_y y_{sp,k} - \tilde{I}_y \delta_{y,k})$$

where

$$\tilde{y}_k = N_x x(k) + \tilde{S} \Delta u_k \tag{5}$$

$$\tilde{y}_k = \begin{bmatrix} y(k|k) \\ y(k+1|k) \\ \vdots \\ y(k+p|k) \end{bmatrix}, \quad N_x = \begin{bmatrix} I_{(p+1)ny} & 0 \end{bmatrix} \in \Re^{(p+1)ny \times nx}$$

$$\tilde{S} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ S_1 & 0 & \cdots & 0 \\ S_2 & S_1 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ S_p & S_{p-1} & \cdots & S_{p-m+1} \end{bmatrix}$$

$$\Delta u_k = \begin{bmatrix} \Delta u(k|k) \\ \Delta u(k+1|k) \\ \vdots \\ \Delta u(k+m-1|k) \end{bmatrix}, \quad \tilde{I}_y = \begin{bmatrix} I_{ny} & \cdots & I_{ny} \end{bmatrix}^T$$

$$\tilde{I}_y \in \Re^{(p+1)ny \times ny}$$

$$\tilde{Q}_y = \text{diag}\left(\underbrace{Q_y \quad \cdots \quad Q_y}_{p+1}\right)$$

$$nx = (p+1)ny + ny + nd$$

Then, considering (5), $V_{k,1}$ can be written as follows

$$V_{k,1} = [N_x x(k) + \tilde{S}\Delta u_k - \tilde{I}_y y_{sp,k} - \tilde{I}_y \delta_{y,k}]^{\mathrm{T}} \times \tilde{Q}_y [N_x x(k) + \tilde{S}\Delta u_k - \tilde{I}_y y_{sp,k} - \tilde{I}_y \delta_{y,k}] \quad (6)$$

The term corresponding to the infinite horizon output error in (4) can be written as follows

$$V_{k,2} = \sum_{j=1}^{\infty} (x^s(k+m|k) + \Psi(p+j-m)x^d(k+m|k) - y_{sp,k} - \delta_{y,k})^{\mathrm{T}} Q_y (x^s(k+m|k) + \Psi(p+j-m)x^d(k+m|k) - y_{sp,k} - \delta_{y,k}) \quad (7)$$

where

$$x^s(k+m|k) = x^s(k) + \tilde{B}^s \Delta u_k, \ \tilde{B}^s = \left[\underbrace{B^s \quad \ldots \quad B^s}_{m}\right]$$

$$x^d(k+m|k) = F^m x^d(k) + \tilde{B}^d \Delta u_k$$

$$\tilde{B}^d = \left[F^{m-1}B^d \quad F^{m-2}B^d \quad \ldots \quad B^d \right]$$

$$\Psi(p+j-m) = \Psi(p-m)F^j$$

Then, in order to guarantee that $V_{k,2}$ will be bounded, the following constraint has to be included in the control problem

$$x^s(k+m|k) - y_{sp,k} - \delta_{y,k} = 0$$

or

$$x^s(k) + \tilde{B}^s \Delta u_k - y_{sp,k} - \delta_{y,k} = 0 \quad (8)$$

Now, assuming that (8) is satisfied, (7) becomes

$$V_{k,2} = \sum_{j=1}^{\infty} (\Psi(p-m)F^j x^d(k+m|k))^{\mathrm{T}} \times Q_y (\Psi(p-m)F^j x^d(k+m|k))$$

$$V_{k,2} = (F^m x^d(k) + \tilde{B}^d \Delta u_k)^{\mathrm{T}} Q_d (F^m x^d(k) + \tilde{B}^d \Delta u_k)$$

where

$$Q_d = \sum_{j=1}^{\infty} (\Psi(p-m)F^j)^{\mathrm{T}} Q_y (\Psi(p-m)F^j)$$

Also, the infinite sum corresponding to the error in the input along the prediction horizon in (4) can be written as follows

$$V_{k,3} = \sum_{j=0}^{\infty} (u(k+j|k) - u_{des,k} - \delta_{u,k})^{\mathrm{T}} \times Q_u (u(k+j|k) - u_{des,k} - \delta_{u,k}) \quad (9)$$

In order to force (9) to be bounded, the following constraint should be included in the control problem

$$u(k+m|k) - u_{des,k} - \delta_{u,k} = 0$$

that is equivalent to

$$u(k-1) + \tilde{I}_u^{\mathrm{T}} \Delta u_k - u_{des,k} - \delta_{u,k} = 0 \quad (10)$$

where

$$\tilde{I}_u^{\mathrm{T}} = \left[\underbrace{I_{nu} \quad \ldots \quad I_{nu}}_{m}\right]$$

Now, assuming that (10) is satisfied, (9) can be written as follows

$$V_{k,3} = (\tilde{I}_u u(k-1) + M\Delta u_k - \tilde{I}_u u_{des,k} - \tilde{I}_u \delta_{u,k})^{\mathrm{T}} \times \tilde{Q}_u (\tilde{I}_u u(k-1) + M\Delta u_k - \tilde{I}_u u_{des,k} - \tilde{I}_u \delta_{u,k})$$

where

$$M = \begin{bmatrix} I_{nu} & 0 & \cdots & 0 \\ I_{nu} & I_{nu} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ I_{nu} & I_{nu} & \cdots & I_{nu} \end{bmatrix} \quad M \in \Re^{(nu.m)\times(nu.m)}$$

$$\tilde{Q}_u = \text{diag}\left(\underbrace{Q_u \quad \cdots \quad Q_u}_{m}\right)$$

Finally, the control cost defined in (4) can be written as follows

$$\begin{aligned} V_k = {} & [N_x x(k) + \tilde{S}\Delta u_k - \tilde{I}_y y_{sp,k} - \tilde{I}_y \delta_{y,k}]^{\mathrm{T}} \\ & \times \tilde{Q}_y [N_x x(k) + \tilde{S}\Delta u_k - \tilde{I}_y y_{sp,k} - \tilde{I}_y \delta_{y,k}] \\ & + (F^m x^d(k) + \tilde{B}^d \Delta u_k)^{\mathrm{T}} Q_d (F^m x^d(k) + \tilde{B}^d \Delta u_k) \\ & + (\tilde{I}_u u(k-1) + M\Delta u_k - \tilde{I}_u u_{des,k} - \tilde{I}_u \delta_{u,k})^{\mathrm{T}} \\ & \times \tilde{Q}_u (\tilde{I}_u u(k-1) + M\Delta u_k - \tilde{I}_u u_{des,k} - \tilde{I}_u \delta_{u,k}) \\ & + \Delta u_k^{\mathrm{T}} \tilde{R} \Delta u_k + \delta_{y,k}^{\mathrm{T}} S_y \delta_{y,k} + \delta_{u,k}^{\mathrm{T}} S_u \delta_{u,k} \end{aligned} \quad (11)$$

In the formulation of the infinite horizon MPC with zone control and input target, the output set point is considered as an additional decision variable of the control problem and the controller is obtained from the solution to the following optimisation problem

$$\min_{\Delta u_k, y_{sp,k}, \delta_{y,k}, \delta_{u,k}} V_k \quad (12)$$

doi: 10.1049/iet-cta.2011.0216

subject to (8), (10), (11) and

$$y_{\min} \le y_{sp,k} \le y_{\max} \quad (13)$$

$$\Delta u_{\min} \le \Delta u(k+j|k) \le \Delta u_{\max} \quad j = 0, 1, \ldots, m-1 \quad (14)$$

$$u_{\min} \le u(k-1) + \sum_{i=0}^{j} \Delta u(k+i|k) \le u_{\max}$$
$$j = 0, 1, \ldots, m-1 \quad (15)$$

Constraints (13)–(15) correspond to the constraints that define the output zone, the input move limitation and the input range, respectively. The problem defined in (12) is a quadratic program (QP) that can be easily solved with the available QP solvers. Alternatively, it is easy to show that (12) can be transformed into the following equivalent optimisation problem:

Problem P1

$$\min_{\Delta u_k, y_{sp,k}, \delta_{y,k}, \delta_{u,k}, \gamma_k} \gamma_k \quad (16)$$

subject to (8), (10), (11), (13), (14), (15) and

$$V_k \le \gamma_k, \quad \gamma_k \ge 0 \quad (17)$$

Observe that applying the Schur complement to inequality (17) and using (11), the following expression can be obtained (see (18))

The simplified notation adopted in (18) considers that the matrix on the left hand side of this inequality is symmetric. It is clear that (18) is a LMI, and, if this inequality is used in place of (17), Problem P1 becomes an LMI optimisation problem that can be solved with available LMI solvers as the Matlab LMI Toolbox.

The solution of the problem defined in (16) produces a control law that stabilises the nominal system as established in the theorem below.

Theorem 1: For the stable system represented in (1) when the model is perfectly known, the state is measured and the desired target is reachable, Problem P1 is always feasible and the successive solution of this problem produces a control law that drives the system to its target while maintaining the remaining system inputs and outputs inside their bounds.

Proof: It is easy to show that Problem P1 is always feasible as the slack variables $\delta_{y,k}$ and $\delta_{u,k}$ are unbounded and assure that the equality constraints can always be satisfied. Now, assume that the state is known and there are no disturbances affecting the system. Then, consider the optimum solution to Problem P1 at time step k

$$\Delta u_k^* = \begin{bmatrix} \Delta u^*(k|k)^{\mathrm{T}} & \cdots & \Delta u^*(k+m-1|k)^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}},$$
$$y_{sp,k}^*,\ \delta_{y,k}^*,\ \delta_{u,k}^* \quad \text{and} \quad \gamma_k^*$$

Next, suppose that the first control move $\Delta u^*(k|k)$ is injected into the real plant and one moves to time step $k+1$. Then, consider the following set of variables

$$\Delta\tilde{u}_k = \begin{bmatrix} \Delta u^*(k+1|k)^{\mathrm{T}} & \cdots & \Delta u^*(k+m-1|k)^{\mathrm{T}} & 0 \end{bmatrix}^{\mathrm{T}}$$
$$\tilde{y}_{sp,k+1} = y_{sp,k}^*, \quad \tilde{\delta}_{y,k+1} = \delta_{y,k}^*, \quad \tilde{\delta}_{u,k+1} = \delta_{u,k}^*$$

and

$$\begin{aligned}\tilde{\gamma}_{k+1} = \gamma_k^* &- (y(k|k) - y_{sp,k}^* - \delta_{y,k}^*)^{\mathrm{T}} Q_y (y(k|k) - y_{sp,k}^* - \delta_{y,k}^*) \\ &- (u^*(k|k) - u_{des,k} - \delta_{u,k}^*)^{\mathrm{T}} Q_u (u^*(k|k) - u_{des,k} - \delta_{u,k}^*) \\ &- \Delta u^*(k|k)^{\mathrm{T}} R \Delta u^*(k|k)\end{aligned} \quad (19)$$

It is easy to show that the set of variables defined above satisfies constraints (8), (10), (13), (14) and (15). Now, to verify that (18) is also satisfied by these variables, define the following matrices (see equation at the bottom of the page)

$$\begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 & \sqrt{\tilde{Q}_y}(N_x x(k) + \tilde{S}\Delta u_k - \tilde{I}_y y_{sp,k} - \tilde{I}_y \delta_{y,k}) \\ 0 & I & 0 & 0 & 0 & 0 & \sqrt{\tilde{Q}_d}(F^m x^d(k) + \tilde{B}^d \Delta u_k) \\ 0 & 0 & I & 0 & 0 & 0 & \sqrt{\tilde{Q}_u}(\tilde{I}_u u(k-1) + M\Delta u_k - \tilde{I}_u u_{des,k} - \tilde{I}_u \delta_{u,k}) \\ 0 & 0 & 0 & I & 0 & 0 & \sqrt{\tilde{R}}\Delta u_k \\ 0 & 0 & 0 & 0 & I & 0 & \sqrt{S_y}\delta_{y,k} \\ 0 & 0 & 0 & 0 & 0 & I & \sqrt{S_u}\delta_{u,k} \\ (*)^{\mathrm{T}} & (*)^{\mathrm{T}} & (*)^{\mathrm{T}} & (*)^{\mathrm{T}} & (*)^{\mathrm{T}} & (*)^{\mathrm{T}} & \gamma_k \end{bmatrix} \ge 0 \quad (18)$$

$$Y = \begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 & \sqrt{\tilde{Q}_y}(N_x x(k+1) + \tilde{S}\Delta\tilde{u}_{k+1} - \tilde{I}_y \tilde{y}_{sp,k+1} - \tilde{I}_y \tilde{\delta}_{y,k+1}) \\ 0 & I & 0 & 0 & 0 & 0 & \sqrt{\tilde{Q}_d}(F^m x^d(k+1) + \tilde{B}^d \Delta\tilde{u}_{k+1}) \\ 0 & 0 & I & 0 & 0 & 0 & \sqrt{\tilde{Q}_u}(\tilde{I}_u u(k) + M\Delta\tilde{u}_{k+1} - \tilde{I}_u u_{des,k} - \tilde{I}_u \tilde{\delta}_{u,k+1}) \\ 0 & 0 & 0 & I & 0 & 0 & \sqrt{\tilde{R}}\Delta\tilde{u}_{k+1} \\ 0 & 0 & 0 & 0 & I & 0 & \sqrt{S_y}\tilde{\delta}_{y,k+1} \\ 0 & 0 & 0 & 0 & 0 & I & \sqrt{S_u}\tilde{\delta}_{u,k+1} \\ (*)^{\mathrm{T}} & (*)^{\mathrm{T}} & (*)^{\mathrm{T}} & (*)^{\mathrm{T}} & (*)^{\mathrm{T}} & (*)^{\mathrm{T}} & \tilde{\gamma}_{k+1} \end{bmatrix}$$

and (see equation at the bottom of the page)

It is easy to show that

$$X^{\mathrm{T}}YX = \begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \gamma_k^* - V_k^* \end{bmatrix}$$

Consequently, it is clear that

$$X^{\mathrm{T}}YX \geq 0$$

Now, since X is non-singular, one concludes that $Y \geq 0$ and (18) is also satisfied. Thus, the solution proposed in (19) is feasible and corresponds to an upper bound $\tilde{\gamma}_{k+1}$ to the cost function and, unless the desired steady-state (or target) has been reached, one has $\tilde{\gamma}_{k+1} < \gamma_k^*$ and $\gamma_{k+1}^* < \gamma_k^*$. This means that the upper bound γ_k to the cost function is a Lyapunov function that guarantees the stability of the closed-loop system with the controller resulting from the solution to Problem P1.

A similar analysis as in [18] can be performed here to show that if the input weight S_u is large enough and the desired steady state is reachable, slacks $\delta_{y,k}$ and $\delta_{u,k}$ will converge to zero and the targets will be reached. □

There is no apparent advantage in obtaining the stable control law through the solution to Problem P1 instead of solving the QP defined in (12). The advantage of the LMI gadgetry to produce a stable MPC becomes more evident when model uncertainty is included in the control problem as presented in the next section.

4 Robust MPC for time-delayed systems with optimising targets and zone control

Now, assume that the model uncertainty is defined as in Section 2 and is characterised by a set of parameters defined as $\Theta_n = \{B_n^s, B_n^d, F_n, \theta_n\}, n = 1, \ldots, L$. Also assume that in this case $p > \max_{i,j,n} \theta_n(i,j) + m$ (this condition guarantees that the state vector of all models have the same dimension). Then, for a given model Θ_n, following the same steps as in the previous section, one can define a cost function as follows

$$\begin{aligned} V_k(\Theta_n) &= (N_x x(k) + \tilde{S}(\Theta_n)\Delta u_k - \tilde{I}_y y_{sp,k}(\Theta_n) \\ &\quad - \tilde{I}_y \delta_{y,k}(\Theta_n))^{\mathrm{T}} \tilde{Q}_y (N_x x(k) + \tilde{S}(\Theta_n)\Delta u_k \\ &\quad - \tilde{I}_y y_{sp,k}(\Theta_n) - \tilde{I}_y \delta_{y,k}(\Theta_n)) \\ &\quad + ((F(\Theta_n))^m x^d(k) + B_m^d(\Theta_n)\Delta u_k)^{\mathrm{T}} \\ &\quad \times Q_d(\Theta_n)((F(\Theta_n))^m x^d(k) \\ &\quad + B_m^d(\Theta_n)\Delta u_k) + (\tilde{I}_u u(k-1) \\ &\quad + M\Delta u_k - \tilde{I}_u u_{des,k} - \tilde{I}_u \delta_{u,k})^{\mathrm{T}} \tilde{Q}_u (\tilde{I}_u u(k-1) \\ &\quad + M\Delta u_k - \tilde{I}_u u_{des,k} - \tilde{I}_u \delta_{u,k}) \\ &\quad + \Delta u_k^{\mathrm{T}} \tilde{R} \Delta u_k + \delta_{y,k}(\Theta_n)^{\mathrm{T}} S_y \delta_{y,k}(\Theta_n) + \delta_{u,k}^{\mathrm{T}} S_u \delta_{u,k} \end{aligned} \quad (20)$$

Also, considering the same steps as in the nominal system case, assume that the following constraints are included in the control problem

$$x^s(k) + \tilde{B}^s(\Theta_n)\Delta u_k - y_{sp,k}(\Theta_n) - \delta_{y,k}(\Theta_n) = 0$$
$$n = 1, \ldots, L$$

$$u(k-1) + \tilde{I}_u^T \Delta u_k - u_{des,k} - \delta_{u,k} = 0$$

Then, at any time step k the robust MPC for the system with time delays and multi-model uncertainty is obtained from the solution to the following problem [7]:

Problem P2

$$\min_{\substack{\Delta u_k, y_{sp,k}(\Theta_n), \delta_{y,k}(\Theta_n), \delta_{u,k} \\ n=1,\ldots,L}} V_k(\Theta_N)$$

subject to

$$\begin{aligned} &\Delta u_{\min} \leq \Delta u(k+j|k) \leq \Delta u_{\max} && j = 0, 1, \ldots, m-1 \\ &u_{\min} \leq u(k+j|k) \leq u_{\max} && j = 0, 1, \ldots, m-1 \\ &y_{\min} \leq y_{sp,k}(\Theta_n) \leq y_{\max} && n = 1, \ldots, L \\ &x^s(k) + \tilde{B}^s \Delta u_k - y_{sp,k}(\Theta_n) - \delta_{y,k}(\Theta_n) = 0 && n = 1, \ldots, L \\ &u(k-1) + \tilde{I}_u^T \Delta u_k - u_{des,k} - \delta_{u,k} = 0 \\ &V_k(\Delta u_k, \delta_{y,k}(\Theta_n), \delta_{u,k}, y_{sp,k}(\Theta_n), \Theta_n) \\ &\quad \leq \tilde{V}_k(\Delta \tilde{u}_k, \tilde{\delta}_{y,k}(\Theta_n), \tilde{\delta}_{u,k}, \tilde{y}_{sp,k}(\Theta_n), \Theta_n) && n = 1, \ldots, L \end{aligned} \quad (21)$$

where, assuming that $(\Delta u_{k-1}^*, y_{u,k-1}^*(\Theta_n), \delta_{u,k-1}^*, \delta_{u,k-1}^*(\Theta_n))$ is the optimum solution at the previous time step $k-1$, one defines

$$\Delta \tilde{u}_k = \left[\Delta u^*(k|k-1)^{\mathrm{T}} \ \ldots \ \Delta u^*(k+m-2|k-1)^{\mathrm{T}} \ \ 0 \right]^{\mathrm{T}},$$
$$\tilde{y}_{sp,k}(\Theta_n) = y_{sp,k-1}^*(\Theta_n)$$

and $\tilde{\delta}_{u,k}$ is such that

$$u(k-1) + \tilde{I}_u^{\mathrm{T}} \Delta \tilde{u}_k - u_{des,k} - \tilde{\delta}_{u,k} = 0$$

$$X = \begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 & -\sqrt{\tilde{Q}_y}(N_x x(k+1) + \tilde{S}\Delta \tilde{u}_{k+1} - \tilde{I}_y \tilde{y}_{sp,k+1} - \tilde{I}_y \tilde{\delta}_{y,k+1}) \\ 0 & I & 0 & 0 & 0 & 0 & -\sqrt{\tilde{Q}_d}(F^m x^d(k+1) + \tilde{B}^d \Delta \tilde{u}_{k+1}) \\ 0 & 0 & I & 0 & 0 & 0 & -\sqrt{\tilde{Q}_u}(\tilde{I}_u u(k) + M\Delta \tilde{u}_{k+1} - \tilde{I}_u u_{des,k} - \tilde{I}_u \tilde{\delta}_{u,k+1}) \\ 0 & 0 & 0 & I & 0 & 0 & -\sqrt{\tilde{R}}\Delta \tilde{u}_{k+1} \\ 0 & 0 & 0 & 0 & I & 0 & -\sqrt{S_y}\tilde{\delta}_{y,k+1} \\ 0 & 0 & 0 & 0 & 0 & I & -\sqrt{S_u}\tilde{\delta}_{u,k+1} \\ 0 & 0 & 0 & 0 & 0 & 0 & I \end{bmatrix}$$

IET Control Theory Appl., 2012, Vol. 6, Iss. 1, pp. 37–50
doi: 10.1049/iet-cta.2011.0216

and $\tilde{\delta}_{y,k}(\Theta_n)$ satisfies the equation

$$x^s(k) + \tilde{B}^s \Delta\tilde{u}_k - \tilde{y}_{sp,k}(\Theta_n) - \tilde{\delta}_{y,k}(\Theta_n) = 0 \quad n = 1, \ldots, L$$

Observe that in Problem P2, Θ_N corresponds to the nominal or most probable model of the system. So, in this problem the objective is to minimise the cost of the most probable plant, which may improve the performance of the controller.

Compared to the nominal control problem, the multi-model control problem includes the non-linear constraints represented in (21). These constraints turn the control problem into a NLP, which is more complicated to solve than the QP that is obtained for the nominal MPC. For systems with large dimension, the online solution to Problem P2 may be computer demanding. In the next chapter, Problem P2 is re-casted as an LMI problem that has a lower computational burden than the existing NLP-based robust MPC.

5 LMI formulation of the robust MPC

Here, the controller resulting from the solution to problem P2 that was formulated for the multi-plant uncertainty case is translated as an LMI problem. Before presenting the problem that defines the controller, one can extend the representation of the model uncertainty to be considered in the robust MPC as described in the following section.

5.1 Model uncertainty characterisation

Suppose that the process to be controlled is represented by the model defined in (1), where the model parameters are defined as in the previous section by the set $\Omega = \{\Theta_1, \ldots, \Theta_L\}$ where each element of this set corresponds to a set of parameters $\Theta_n = (F, B^s, B^d, \theta)_n$, $n = 1, \ldots, L$. Suppose also that the true model is such that $(B^s, B^d)_T = \sum_{i=1}^{L} \lambda_i (B^s, B^d)_i$, $\sum_{i=1}^{L} \lambda_i = 1$, $\lambda_i \geq 0$, $F_T \in (F_1, \ldots, F_L)$ and $\theta_T \in (\theta_1, \ldots, \theta_L)$. Then, it is interesting to separate (B^s, B^d) from the remaining model parameters and to define the two following sub-sets: $\Upsilon_n = (B^s, B^d)_n$, $n = 1, \ldots, L$ and $\Lambda_n = (F, \theta)_n$, $n = 1, \ldots, L$.

Observe that the uncertainty described above is more general than the multi-plant uncertainty considered in Problem P2. This means that the controller proposed here will be robust to a class of process gains that is larger than the class of process gains considered in the previous section. For the remaining model parameters $(F, \theta)_{n=1,\ldots,L}$ the same sort of uncertainty as in the controller defined through Problem P2 is considered, which means that the multi-plant uncertainty is assumed for these parameters.

5.2 LMI-based robust MPC

Now, following the same steps as in the nominal MPC, the robust MPC defined through Problem P2 can be reformulated as an LMI problem that minimises the upper bound to the control cost of the nominal system and forces the contraction of the upper bound associated to each of the possible process models. Then, the LMI formulation of the robust MPC can be written as follows:

Problem P3

$$\min_{\substack{\Delta u_k, y_{sp,k}(\Upsilon_n,\Lambda_i), \delta_{y,k}(\Upsilon_n,\Lambda_i), \delta_{u,k}, \gamma_k(\Upsilon_n,\Lambda_i) \\ n=1,\ldots,L \quad i=1,\ldots,L}} \gamma_k(\Upsilon_N, \Lambda_N)$$

subject to (see (22))

where

$$X_1 = \sqrt{\tilde{Q}_y}(N_x x(k) + \tilde{S}(\Upsilon_n)\Delta u_k - \tilde{I}_y y_{sp,k}(\Upsilon_n, \Lambda_i) - \tilde{I}_y \delta_{y,k}(\Upsilon_n, \Lambda_i))$$

$$X_2 = \sqrt{\tilde{Q}_d}(F(\Lambda_i)^m x^d(k) + \tilde{B}^d(\Upsilon_n, \Lambda_i)\Delta u_k)$$

$$X_3 = \sqrt{\tilde{Q}_u}(\tilde{I}_u u(k-1) + M\Delta u_k - \tilde{I}_u u_{des,k} - \tilde{I}_u \delta_{u,k})$$

$$\Delta u_{\min} < \Delta u(k+j|k) < \Delta u_{\max} \quad j = 0, 1, \ldots, m-1 \quad (23)$$

$$u_{\min} < u(k+j|k) < u_{\max} \quad j = 0, 1, \ldots, m-1 \quad (24)$$

$$y_{\min} < y_{sp,k}(\Upsilon_n, \Lambda_i) < y_{\max} \quad n = 1, \ldots, L, \; i = 1, \ldots, L \quad (25)$$

$$x^s(k) + \tilde{B}^s(\Upsilon_n)\Delta u_k - y_{sp,k}(\Upsilon_n, \Lambda_i) - \delta_{y,k}(\Upsilon_n, \Lambda_i) = 0 \quad n = 1, \ldots, L, \quad i = 1, \ldots, L \quad (26)$$

$$u(k-1) + \tilde{I}_u^{\mathrm{T}} \Delta u_k - u_{des,k} - \delta_{u,k} = 0 \quad (27)$$

$$-\gamma_k(\Upsilon_n, \Lambda_i) + \tilde{V}_k(\Delta\tilde{u}_k, \tilde{y}_{sp,k}(\Upsilon_n, \Lambda_i), \Upsilon_n, \Lambda_i) \geq 0 \quad n = 1, \ldots, L, \quad i = 1, \ldots, L \quad (28)$$

$$\tilde{X}_1 = \sqrt{\tilde{Q}_y}(N_x x(k) + \tilde{S}(\Upsilon_n)\Delta\tilde{u}_k - \tilde{I}_y \tilde{y}_{sp,k}(\Upsilon_n, \Lambda_i) - \tilde{I}_y \tilde{\delta}_{y,k}(\Upsilon_n, \Lambda_i))$$

$$\tilde{X}_2 = \sqrt{\tilde{Q}_d}(F(\Lambda_i)^m x^d(k) + \tilde{B}^d(\Upsilon_n, \Lambda_i)\Delta\tilde{u}_k)$$

$$\tilde{X}_3 = \sqrt{\tilde{Q}_u}(\tilde{I}_u u(k-1) + M\Delta\tilde{u}_k - \tilde{I}_u u_{des,k} - \tilde{I}_u \tilde{\delta}_{u,k})$$

Observe that LMI (22) is equivalent to $V_k(\Delta u_k, y_{sp,k}(\Upsilon_n, \Lambda_i), \Upsilon_n, \Lambda_i) \leq \gamma_k(\Upsilon_n, \Lambda_i)$ for $n = 1, \ldots, L$, $i = 1, \ldots, L$,

$$\begin{bmatrix} I & 0 & 0 & 0 & 0 & 0 & X_1 \\ 0 & I & 0 & 0 & 0 & 0 & X_2 \\ 0 & 0 & I & 0 & 0 & 0 & X_3 \\ 0 & 0 & 0 & I & 0 & 0 & \sqrt{\tilde{R}}\Delta u_k \\ 0 & 0 & 0 & 0 & I & 0 & \sqrt{S_y}\delta_{y,k}(\Upsilon_n, \Lambda_i) \\ 0 & 0 & 0 & 0 & 0 & I & \sqrt{S_u}\delta_{u,k} \\ X_1^{\mathrm{T}} & X_2^{\mathrm{T}} & X_2^{\mathrm{T}} & \Delta u_k^{\mathrm{T}}\sqrt{\tilde{R}} & \delta_{y,k}^{\mathrm{T}}(\Upsilon_n, \Lambda_i)\sqrt{S_y} & \delta_{u,k}^{\mathrm{T}}\sqrt{S_u} & \gamma_k(\Upsilon_n, \Lambda_i) \end{bmatrix} \geq 0, \quad n = 1, \ldots, L, \quad i = 1, \ldots, L \quad (22)$$

which together with (28) results in the following inequality

$$V_k(\Delta u_k, y_{sp,k}(\Upsilon_n, \Lambda_i), \Upsilon_n, \Lambda_i) \le \tilde{V}_k(\Delta\tilde{u}_k, \tilde{y}_{sp,k}(\Upsilon_n, \Lambda_i), \Upsilon_n, \Lambda_i), \quad n = 1, \ldots, L, \; i = 1, \ldots, L$$

These constraints are linear matrix inequalities in the decision variables of the control problem, and so, Problem P3 can be solved through an LMI solver.

At this point, it is interesting to observe how the uncertainty in the model parameters affects the constraints of Problem P3. It is not difficult to show that the parameters that appear in the input matrix B of the model defined in (1) will appear linearly in all the constraints of problem P3. This means that if these constraints are satisfied by a finite set of models characterised by different values of parameters: $S_1, \ldots, S_{p+1}$, B^s and B^d, they will also be satisfied by any convex combination of these models. For the model defined in (1), uncertainty in these parameters can be interpreted as uncertainty in the static gains of the process system, which is quite common in the process industries. On the other hand, uncertainty in the parameters that appear in the state matrix A of the model defined in (1) is related to the dynamic modes of the system, which, as it can be easily shown, do not appear linearly in the constraints of Problem P3. Consequently, only the multi-plant uncertainty can be considered for these parameters in the controller defined through Problem P3. It is quite straightforward to show that uncertainty in the time delays will also appear non-linearly in the constraints of Problem P3 and, consequently, follows the same pattern.

Owing to the inclusion of slack variables, the controller defined through Problem P3 is always feasible and if the state of the true system is measured, the stability of the closed-loop system can be guaranteed by the following theorem.

Theorem 2: Suppose that the process to be controlled is represented by a model as defined in (1), where the true model is unknown but it is known to belong to the family of models defined in Section 5.1, then the controller resulting from the solution to Problem P3 stabilises the true plant. This means that, for the true plant, if the desired steady-state is reachable, then the system inputs and outputs with targets will converge to their targets while the remaining inputs and outputs will converge to values inside their respective zones.

Proof: Suppose that the true plant can be represented by a model as defined in (1) where matrices A, B and C are such that $(B^s, B^d)_T = \sum_{i=1}^{L} \lambda_i (B^s, B^d)_i$, $\sum_{i=1}^{L} \lambda_i = 1$, $\lambda_i \ge 0$, $F_T \in (F_1, \ldots, F_L)$ and $\theta_T \in (\theta_1, \ldots, \theta_L)$. Suppose also that, at any time step k, the solution to Problem P3 is represented by: $\{\Delta u_k^*, \; y_{sp,k}^*(\Upsilon_n, \Lambda_i), \; \delta_{y,k}^*(\Upsilon_n, \Lambda_i), \; \delta_{u,k}^*, \; \gamma_k^*(\Upsilon_n, \Lambda_i)\}$ with $n = 1, \ldots, L$ and $i = 1, \ldots, L$, then, it is easy to show that the following set of variables

$$\left\{\Delta u_k^*, \; \sum_{n=1}^{L} \lambda_n y_{sp,k}^*(\Upsilon_n, \Lambda_T), \; \sum_{n=1}^{L} \lambda_n \delta_{y,k}^*(\Upsilon_n, \Lambda_T), \; \delta_{u,k}^*, \; \sum_{n=1}^{L} \lambda_n \gamma_k^*(\Upsilon_n, \Lambda_T)\right\} \tag{29}$$

is a feasible solution to problem P3 if this problem is written only for the true plant (assuming that the true plant is known). Observe that in this case, the upper bound to the true process cost function is given by $\gamma_k^*(\Upsilon_T, \Lambda_T) = \sum_{n=1}^{L} \lambda_n \gamma_k^*(\Upsilon_n, \Lambda_T)$.

Now, consider that $\Delta u^*(k|k)$ is injected in the real process and one moves to time step $k+1$ where Problem P3 has to be solved again. Now following a similar procedure as in Theorem 1, one can show that if $\Delta\tilde{u}_{k+1}$ is defined as in Theorem 1 and one defines $\tilde{y}_{sp,k+1}(\Upsilon_n, \Lambda_T) = y_{sp,k}^*(\Upsilon_n, \Lambda_T)$, $\tilde{\delta}_{y,k+1}(\Upsilon_n, \Lambda_T) = \delta_{y,k}^*(\Upsilon_n, \Lambda_T)$, $\tilde{\delta}_{u,k+1} = \delta_{u,k}^*$, and

$$\begin{aligned}\tilde{\gamma}_{k+1}(\Upsilon_T, \Lambda_T) = \gamma_k^*(\Upsilon_T, \Lambda_T) &- (y(k|k) - y_{sp,k}^*(\Upsilon_T, \Lambda_T) \\ &- \delta_{y,k}^*(\Upsilon_T, \Lambda_T))^{\mathrm{T}} Q_y (y(k|k) - y_{sp,k}^*(\Upsilon_T, \Lambda_T) \\ &- \delta_{y,k}^*(\Upsilon_T, \Lambda_T)) - (u^*(k|k) - u_{des,k} \\ &- \delta_{u,k}^*)^{\mathrm{T}} Q_u (u^*(k|k) - u_{des,k} - \delta_{u,k}^*) \\ &- \Delta u^*(k|k)^{\mathrm{T}} R \Delta u^*(k|k)\end{aligned}$$

then

$$\left\{\Delta\tilde{u}_{k+1}, \; \sum_{n=1}^{L} \lambda_n \tilde{y}_{sp,k+1}(\Upsilon_n, \Lambda_T), \; \sum_{n=1}^{L} \lambda_n \tilde{\delta}_{y,k+1}(\Upsilon_n, \Lambda_T), \; \tilde{\delta}_{u,k+1}, \; \sum_{n=1}^{L} \lambda_n \tilde{\gamma}_{k+1}(\Upsilon_n, \Lambda_T)\right\}$$

is a feasible solution to Problem P3 at $k+1$ for the true process system and obviously $\tilde{\gamma}_{k+1}(\Upsilon_T, \Lambda_T) \le \gamma_k^*(\Upsilon_T, \Lambda_T)$ where equality holds only if a steady state has been reached. Consequently, $\gamma_{k+1}^*(\Upsilon_T, \Lambda_T) \le \gamma_k^*(\Upsilon_T, \Lambda_T)$ and $\gamma_k^*(\Upsilon_T, \Lambda_T)$ is a Lyapunov function to the closed-loop system that is asymptotically stable. □

5.3 Implementation of the robust MPC in the LMI framework

Observe that in Problem P3, constraints (26) and (27) are equality constraints that cannot be included in some of the available LMI solvers as for the Matlab LMI Toolbox, for which all the constraints need to be expressed in terms of inequality constraints. However, as the slack variables $\delta_{y,k}(\Upsilon_n, \Lambda_i)$ and $\delta_{u,k}$ are unbounded, (26) and (27) can be used to eliminate these variables from the remaining constraints of the control problem. Then, the cost function can be simplified and represented in terms of the reduced set of decision variables as follows

$$\begin{aligned}V_k(\Upsilon_n, \Lambda_i) = {} & [N_x x(k) - \tilde{I}_{xs} x^s(k) + (\tilde{S}(\Upsilon_n) - \tilde{B}^s(\Upsilon_n))\Delta u_k]^{\mathrm{T}} \tilde{Q}_y \\ & \times [N_x x(k) - \tilde{I}_{xs} x^s(k) + (\tilde{S}(\Upsilon_n) - \tilde{B}^s(\Upsilon_n))\Delta u_k] \\ & + ((F(\Lambda_i))^m x^d(k) + \tilde{B}^d(\Upsilon_n, \Lambda_i)\Delta u_k)^{\mathrm{T}} Q_d(\Lambda_i) \\ & \times ((F(\Lambda_i))^m x^d(k) + \tilde{B}^d(\Upsilon_n, \Lambda_i)\Delta u_k) \\ & + [(M - \tilde{\tilde{I}}_u)\Delta u_k]^{\mathrm{T}} \tilde{Q}_u [(M - \tilde{\tilde{I}}_u)\Delta u_k] + \Delta u_k^{\mathrm{T}} \tilde{R} \Delta u_k \\ & + (x^s(k) + \tilde{B}^s(\Upsilon_n)\Delta u_k - y_{sp,k}(\Upsilon_n, \Lambda_i))^{\mathrm{T}} \\ & \times S_y (x^s(k) + \tilde{B}^s(\Upsilon_n)\Delta u_k - y_{sp,k}(\Upsilon_n, \Lambda_i)) \\ & + (u(k-1) + \tilde{I}_u^{\mathrm{T}} \Delta u_k - u_{des,k})^{\mathrm{T}} S_u (u(k-1) \\ & + \tilde{I}_u^{\mathrm{T}} \Delta u_k - u_{des,k})\end{aligned}$$

IET Control Theory Appl., 2012, Vol. 6, Iss. 1, pp. 37–50
doi: 10.1049/iet-cta.2011.0216

that can be written in the quadratic form

$$V_k(\Upsilon_n, \Lambda_i) = \begin{bmatrix} \Delta u_k^{\mathrm{T}} & y_{sp,k}^{\mathrm{T}}(\Upsilon_n, \Lambda_i) \end{bmatrix} \times \begin{bmatrix} H_{11}(\Upsilon_n, \Lambda_i) & H_{12}(\Upsilon_n) \\ H_{21}(\Upsilon_n) & H_{22} \end{bmatrix} \begin{bmatrix} \Delta u_k \\ y_{sp,k}(\Upsilon_n, \Lambda_i) \end{bmatrix} + 2 \times \begin{bmatrix} C_{fk,1}(\Upsilon_n, \Lambda_i) & C_{fk,2} \end{bmatrix} \begin{bmatrix} \Delta u_k \\ y_{sp,k}(\Upsilon_n, \Lambda_i) \end{bmatrix} + c_k(\Upsilon_n, \Lambda_i) \quad (30)$$

where

$$H_{11}(\Upsilon_n, \Lambda_i) = [\tilde{S}(\Upsilon_n) - \tilde{B}^s(\Upsilon_n)]^{\mathrm{T}} \tilde{Q}_y [\tilde{S}(\Upsilon_n) - \tilde{B}^s(\Upsilon_n)] + \tilde{B}^d(\Upsilon_n, \Lambda_i)^{\mathrm{T}} Q_d(\Lambda_i) \tilde{B}^d(\Upsilon_n, \Lambda_i) + \tilde{R} + \tilde{B}^s(\Upsilon_n)^{\mathrm{T}} S_y \tilde{B}^s(\Upsilon_n) + \tilde{I}_u S_u \tilde{I}_u^{\mathrm{T}}$$

$$H_{12}(\Upsilon_n) = -\tilde{B}^s(\Upsilon_n)^{\mathrm{T}} S_y$$

$$H_{21}(\Upsilon_n) = H_{12}(\Upsilon_n)^{\mathrm{T}}$$

$$H_{22} = S_y$$

$$C_{fk,1}(\Upsilon_n, \Lambda_i) = [N_x x(k) - \tilde{I}_{xs} x^s(k)]^{\mathrm{T}} \tilde{Q}_y [\tilde{S}(\Upsilon_n) - \tilde{B}^s(\Upsilon_n)] + [(F(\Upsilon_n, \Lambda_i))^m x^d(k)]^{\mathrm{T}} Q_d(\Lambda_i) \tilde{B}^d(\Upsilon_n, \Lambda_i) + x^s(k)^{\mathrm{T}} S_y \tilde{B}^s(\Upsilon_n) + [u(k-1) - u_{des,k}]^{\mathrm{T}} S_u \tilde{I}_u^{\mathrm{T}}$$

$$C_{fk,2} = -x^s(k)^{\mathrm{T}} S_y$$

$$c_k(\Upsilon_n, \Lambda_i) = [N_x x(k) - \tilde{I}_{xs} x^s(k)]^{\mathrm{T}} \tilde{Q}_y [N_x x(k) - \tilde{I}_{xs} x^s(k)] + [(F(\Lambda_i))^m x^d(k)]^{\mathrm{T}} Q_d(\Lambda_i) [(F(\Lambda_i))^m x^d(k)] + x^{s\mathrm{T}}(k) S_y x^s(k) + [u(k-1) - u_{des,k}]^{\mathrm{T}} \times S_u [u(k-1) - u_{des,k}]$$

Equation (30) can also be written as follows

$$V_k(\Upsilon_n, \Lambda_i) = Z_k(\Upsilon_n, \Lambda_i)^{\mathrm{T}} H(\Upsilon_n, \Lambda_i) Z_k(\Upsilon_n, \Lambda_i) + 2C_{f,k}(\Upsilon_n, \Lambda_i) Z(\Upsilon_n, \Lambda_i) + c_k(\Upsilon_n, \Lambda_i)$$

where

$$Z_k(\Upsilon_n, \Lambda_i) = \begin{bmatrix} \Delta u_k \\ y_{sp,k}(\Upsilon_n, \Lambda_i) \end{bmatrix}$$

$$H(\Upsilon_n, \Lambda_i) = \begin{bmatrix} H_{11}(\Upsilon_n, \Lambda_i) & H_{12}(\Upsilon_n) \\ H_{21}(\Upsilon_n) & H_{22} \end{bmatrix}$$

$$C_{fk}(\Upsilon_n, \Lambda_i) = \begin{bmatrix} C_{fk,1}(\Upsilon_n, \Lambda_i) & C_{fk,2} \end{bmatrix}$$

Finally, to be solved with an LMI solver such as the one available in the Matlab LMI Toolbox, Problem P3 is reformulated as follows:

Problem P4 (see (31))

where

$$\tilde{V}_k(\Upsilon_n, \Lambda_i) = \begin{bmatrix} \Delta \tilde{u}_k^{\mathrm{T}} & y_{sp,k-1}^{*\,\mathrm{T}}(\Upsilon_n, \Lambda_i) \end{bmatrix} \times \begin{bmatrix} H_{11}(\Upsilon_n, \Lambda_i) & H_{12}(\Upsilon_n) \\ H_{21}(\Upsilon_n) & H_{22} \end{bmatrix} \begin{bmatrix} \Delta \tilde{u}_k \\ y_{sp,k-1}^{*}(\Upsilon_n, \Lambda_i) \end{bmatrix} + 2\begin{bmatrix} C_{fk-1,1}(\Upsilon_n, \Lambda_i) & C_{fk-1,2} \end{bmatrix} \times \begin{bmatrix} \Delta \tilde{u}_k \\ y_{sp,k-1}^{*}(\Upsilon_n, \Lambda_i) \end{bmatrix} + c_{k-1}(\Upsilon_n, \Lambda_i)$$

Observe that Problem P4 is equivalent to Problem P3, but now the robust control problem is an LMI problem, or, an optimisation problem with a linear objective function and linear matrix inequalities as constraints. This problem can then be solved, for instance, with the Matlab LMI Toolbox.

Besides being more general in terms of considering a broader class of model uncertainties, the LMI-based robust MPC that results from the solution to Problem P4, shows a better potential in terms of numerical efficiency when compared with the robust NLP-based MPC based on the

$$\min_{\substack{\Delta u_k, y_{sp,k}(\Upsilon_n, \Lambda_i) \\ n=1,\ldots,L \quad i=1,\ldots,L}} \gamma_k(\Upsilon_N, \Lambda_N)$$

subject to

$$\begin{bmatrix} I & \sqrt{H(Z(\Upsilon_n, \Lambda_i))}\, Z_k(\Upsilon_n, \Lambda_i)^{\mathrm{T}} \\ Z_k(\Upsilon_n, \Lambda_i)\sqrt{H(\Upsilon_n, \Lambda_i)} & \gamma_k(\Upsilon_n, \Lambda_i) - 2C_{fk}(\Upsilon_n, \Lambda_i) Z(\Upsilon_n, \Lambda_i) - c_k(\Upsilon_n, \Lambda_i) \end{bmatrix} > 0$$

$$n = 1, \ldots, L \quad i = 1, \ldots, L$$

$$\tilde{V}_k(\Upsilon_n, \Lambda_i) - \gamma_k(\Upsilon_n, \Lambda_i) > 0 \quad n = 1, \ldots, L \quad i = 1, \ldots, L \quad (31)$$

$$\Delta u_i(k+j|k) - \Delta u_{i,\min} > 0 \quad i = 1, \ldots, nu \quad j = 0, 1, \ldots, m-1$$

$$\Delta u_{i,\max} - \Delta u_i(k+j|k) > 0 \quad i = 1, \ldots, nu \quad j = 0, 1, \ldots, m-1$$

$$u_i(k+j|k) - u_{i,\min} > 0 \quad i = 1, \ldots, nu \quad j = 0, 1, \ldots, m-1$$

$$u_{i,\max} - u_i(k+j|k) > 0 \quad i = 1, \ldots, nu \quad j = 0, 1, \ldots, m-1$$

$$y_{j,sp,k}(\Upsilon_n, \Lambda_i) - y_{j,\min} > 0 \quad n = 1, \ldots, L \quad j = 1, \ldots, ny \quad i = 1, \ldots, L$$

$$y_{j,\max} - y_{j,sp,k}(\Upsilon_n, \Lambda_i) > 0 \quad n = 1, \ldots, L \quad j = 1, \ldots, ny \quad i = 1, \ldots, L$$

solution to Problem P2. In the next section, a low-order simulation example is used to compare the performances of these two methods.

6 Simulation results

The system considered here is part of the fluid catalytic cracking (FCC) system studied in [19] where more details about this system can be found. The FCC system shows to be highly non-linear and a conventional MPC may have a poor performance if the operating point of the system is frequently changed, which may result in large uncertainties in the linear model of the reactor system.

In the study developed here, the system considered has two inputs and three outputs. In this low-order system, the manipulated inputs correspond to: u_1 air flow rate to the catalyst regenerator, u_2 opening of the regenerated catalyst valve, and the controlled outputs are: y_1 riser temperature, y_2 regenerator dense phase temperature, y_3 regenerator dilute phase temperature.

Three models were obtained experimentally corresponding to different operating points of the FCC system. These models may be assumed to constitute the multi-model set Ω on which the robust controller is based. Based on these models, one may construct the polytope in which the gain of the true model is supposed to lie. The parameters corresponding to each of the models can be seen in the following transfer functions

$$G_1(s) = \begin{bmatrix} \dfrac{0.4515e^{-2s}}{2.9846s+1} & \dfrac{0.2033e^{-4s}}{1.7187s+1} \\ \dfrac{1.5e^{-6s}}{20s+1} & \dfrac{(0.1886s-3.8087)e^{-3s}}{17.7347s^2+10.8348s+1} \\ \dfrac{1.7455e^{-6s}}{9.1085s+1} & \dfrac{-6.1355e^{-5s}}{10.9088s+1} \end{bmatrix}$$

$$G_2(s) = \begin{bmatrix} \dfrac{0.25e^{-2s}}{3.5s+1} & \dfrac{0.135e^{-5s}}{2.77s+1} \\ \dfrac{0.9e^{-3s}}{25s+1} & \dfrac{(0.1886s-2.8)e^{-4s}}{19.7347s^2+10.8348s+1} \\ \dfrac{1.25e^{-5s}}{11.1085s+1} & \dfrac{-5e^{-6s}}{12.9088s+1} \end{bmatrix}$$

$$G_3(s) = \begin{bmatrix} \dfrac{0.7e^{-3s}}{1.98s+1} & \dfrac{0.5e^{-4s}}{2.7s+1} \\ \dfrac{2.3e^{-5s}}{25s+1} & \dfrac{(0.1886s-4.8087)e^{-3s}}{15.7347s^2+10.8348s+1} \\ \dfrac{3e^{-4s}}{7s+1} & \dfrac{-8.1355e^{-6s}}{7.9088s+1} \end{bmatrix}$$

In the first simulation shown in this section, the multi-plant uncertainty is considered. Model G_1 is the true model (G_T), while model G_3 represents the nominal model (G_N). One initially compares the performances of the LMI-based robust MPC resulting from the solution to Problem P4 to the NLP-based robust MPC resulting from the solution to Problem P2. For the implementation of this case, the Matlab LMI Toolbox was used to solve Problem P4 and the Matlab 'fmincon' routine was used to solve Problem P2.

The following tuning parameters were adopted for both controllers.

$$T = 1\,\text{min},\ m = 3,\ Q_y = \text{diag}(0.5, 0.5, 0.5),$$
$$R = \text{diag}(10, 10),\ Q_u = \text{diag}(1, 1),$$
$$S_y = \text{diag}(1, 1, 1) \times 10^3,\ S_u = \text{diag}(1, 1) \times 10^3$$

Figs. 1 and 2 show the responses of both robust controllers for the case where the input target is $u_{des} = (225, 71)$, the output

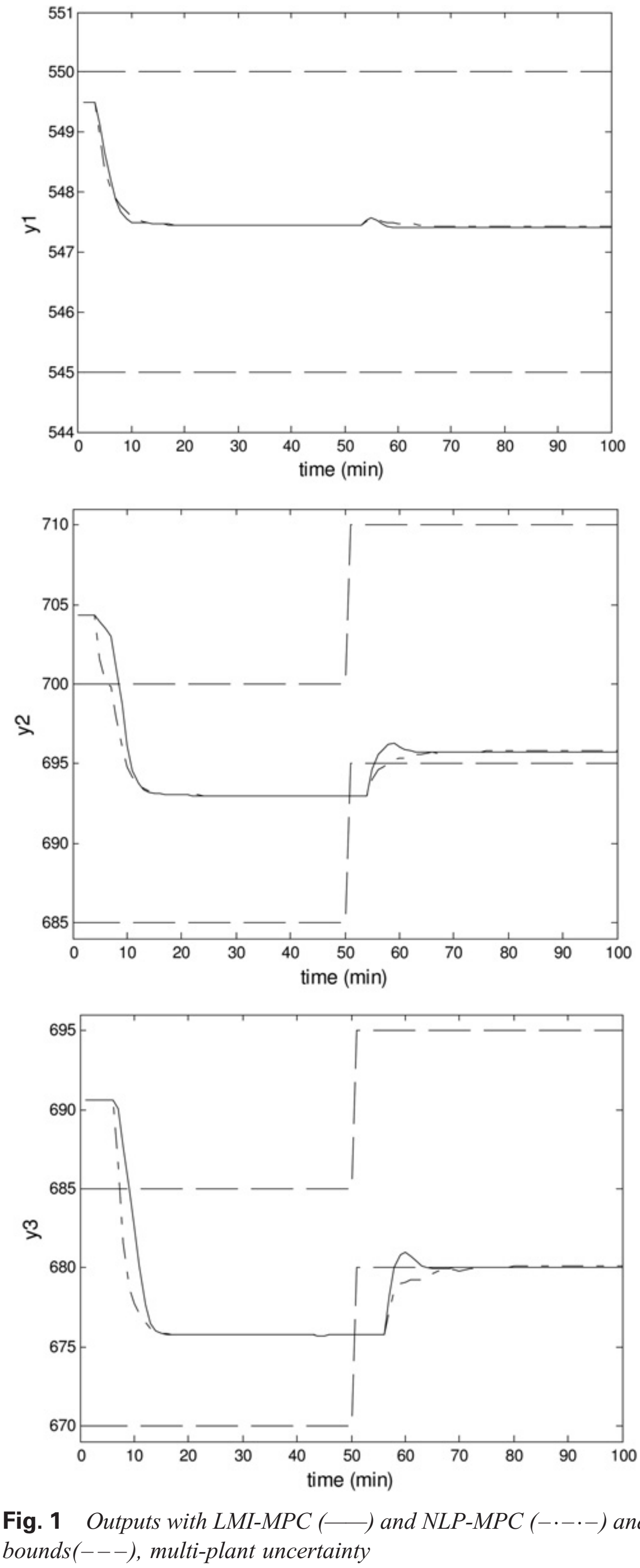

Fig. 1 *Outputs with LMI-MPC (——) and NLP-MPC (–·–·–) and bounds(– – –), multi-plant uncertainty*

doi: 10.1049/iet-cta.2011.0216

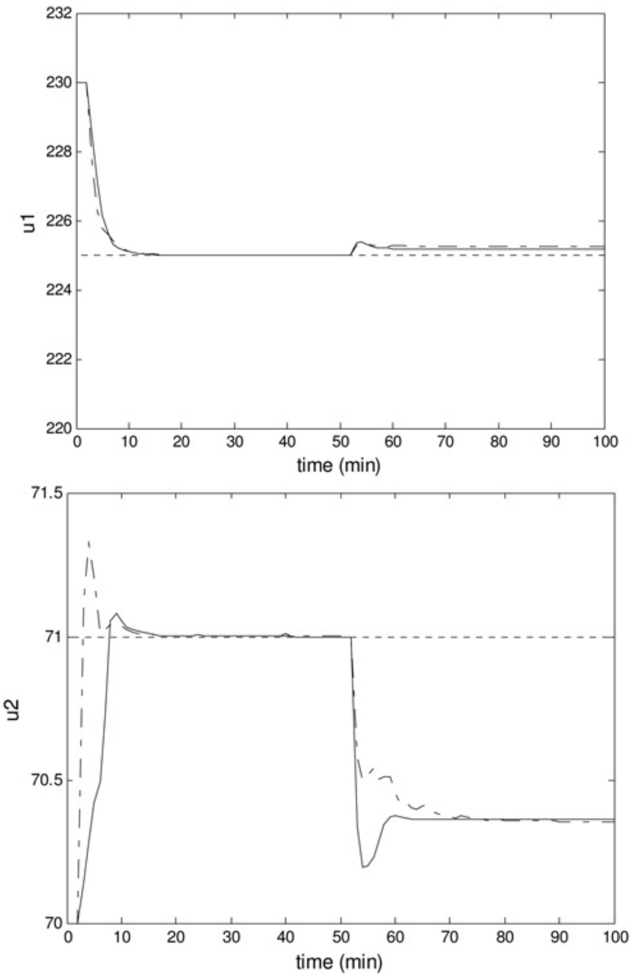

Fig. 2 *Inputs with LMI-MPC (——) and NLP-MPC (–·–·–) and targets (– – –), multi-plant uncertainty*

zone limits are $y_{\min} = \begin{bmatrix} 545 & 685 & 670 \end{bmatrix}^{\mathrm{T}}$ and $y_{\max} = \begin{bmatrix} 550 & 700 & 685 \end{bmatrix}^{\mathrm{T}}$, the input bounds are $u_{\min} = \begin{bmatrix} 175 & 25 \end{bmatrix}^{\mathrm{T}}$ and $u_{\max} = \begin{bmatrix} 250 & 100 \end{bmatrix}^{\mathrm{T}}$, the input move bounds is $\Delta u_{\max} = \begin{bmatrix} 5 & 2 \end{bmatrix}^{\mathrm{T}}$. The reactor system starts from the initial operating point defined through $u(0) = \begin{bmatrix} 230 & 70 \end{bmatrix}^{\mathrm{T}}$ and $y(0) = \begin{bmatrix} 549.5 & 704.3 & 690.6 \end{bmatrix}^{\mathrm{T}}$. The responses of the two controllers are quite similar and there are no major differences between the performances of the NLP-based MPC and the LMI-based MPC. Note that y_2 starts from a value outside the control zone and that both controllers can easily bring this output back to its control zone. At the same time the inputs are driven to their targets after a few time steps while all the outputs converge to steady-state values inside their respective control zones. This is so because the input targets are reachable, which means that at steady state, when the inputs are at their targets, the corresponding values of the outputs lie inside their control zones. The only significant difference between the two controllers is the computer time to perform the simulation. With the NLP-based MPC the simulation takes 45.4 s, while with the LMI-based MPC the simulation lasts only 10.5 s. Then, the proposed LMI-based approach is capable of a significant reduction of the computer time when compared with the previous approach. At time instant $k = 50$ min, the output bounds are modified to the following values: $y_{\min} = \begin{bmatrix} 545 & 695 & 680 \end{bmatrix}^{\mathrm{T}}$ and $y_{\max} = \begin{bmatrix} 550 & 710 & 695 \end{bmatrix}^{\mathrm{T}}$. One can observe that, with these new bounds, the steady-state values of y_2 and y_3 corresponding to the input targets will lie outside their bounds, which means that with the new control specifications, the input targets are unreachable. The controllers have to change the system inputs in order to bring the outputs to a point inside the new zones. Both controllers reach the same steady-state where the distance to the input target values is minimised. Fig. 3 shows for the LMI-based MPC the variable $\gamma_{k,T}$ that is the upper bound to the cost function of the true model that in this simulation is G_1. One can observe that for $k < 50$ where the target is reachable, $\gamma_{k,T}$ tends to zero confirming Theorem 2. However, for $k > 50$, as the target becomes unreachable, $\gamma_{k,T}$ tends to a value that is not null confirming that the closed-loop system is stable but does not converge to the desired steady state.

In the second simulation case presented here, the multi-plant uncertainty is still considered and the LMI-based MPC is tested for the case where there are targets to output y_1 and to input u_2, while the remaining outputs should be controlled inside their zones. It is also investigated the effect of switching between MPC controllers that are based on different nominal models. The tuning of the LMI-based MPC is the same as in the first simulation case except for the input error and the input

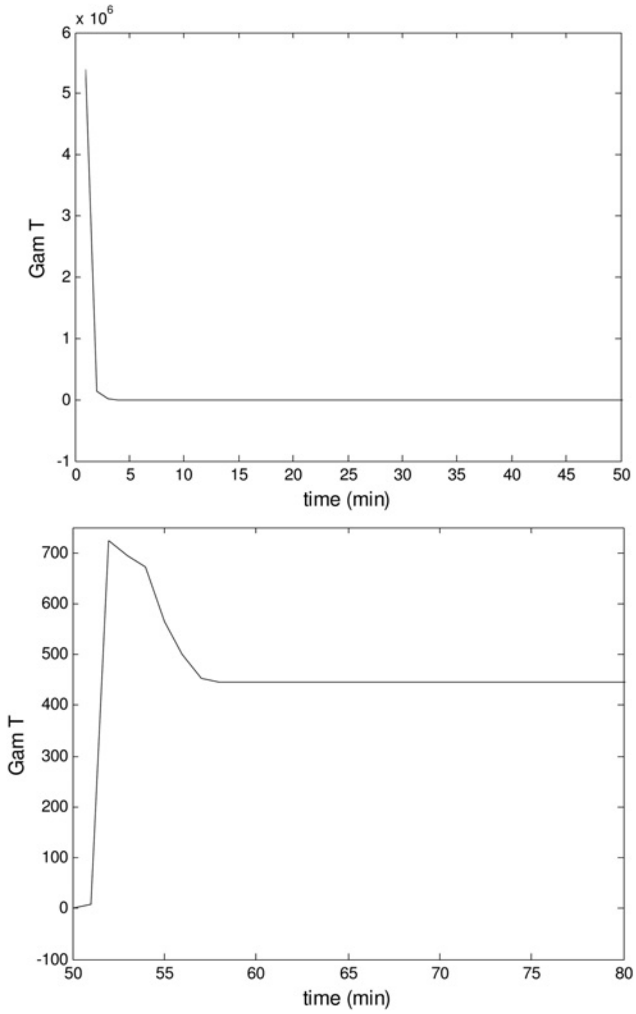

Fig. 3 *Upper bound to the cost function of the true plant with LMI-MPC, multi-plant uncertainty*

slack weights that are now $Q_u = \text{diag}(0, 1)$ and $S_u = \text{diag}(0, 1) \times 10^3$, respectively.

Figs. 4 and 5 show the outputs and inputs for the closed-loop system with the LMI-based MPC when $u_{des,2} = 71$ and the output zone limits are initially $y_{\min} = [548 \quad 685 \quad 670]^T$ and $y_{\max} = [548 \quad 715 \quad 710]^T$. The calculated artificial set-points to the outputs are also shown. The input bounds are the same as in the first case. Observe that the lower and upper bounds of output y_1 are the same, which corresponds to considering a fixed target to this

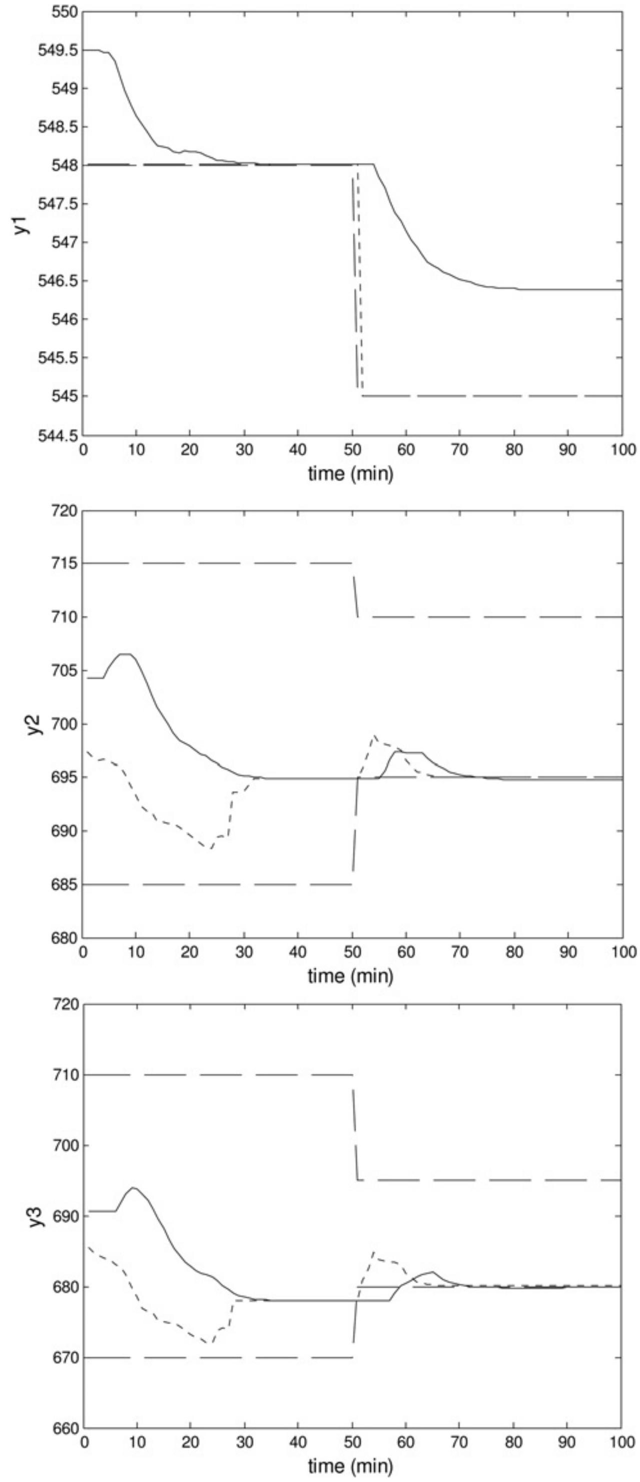

Fig. 4 *Outputs with LMI-MPC (——), bounds (– – –) and set-points (· · · · · ·), Switching nominal model*

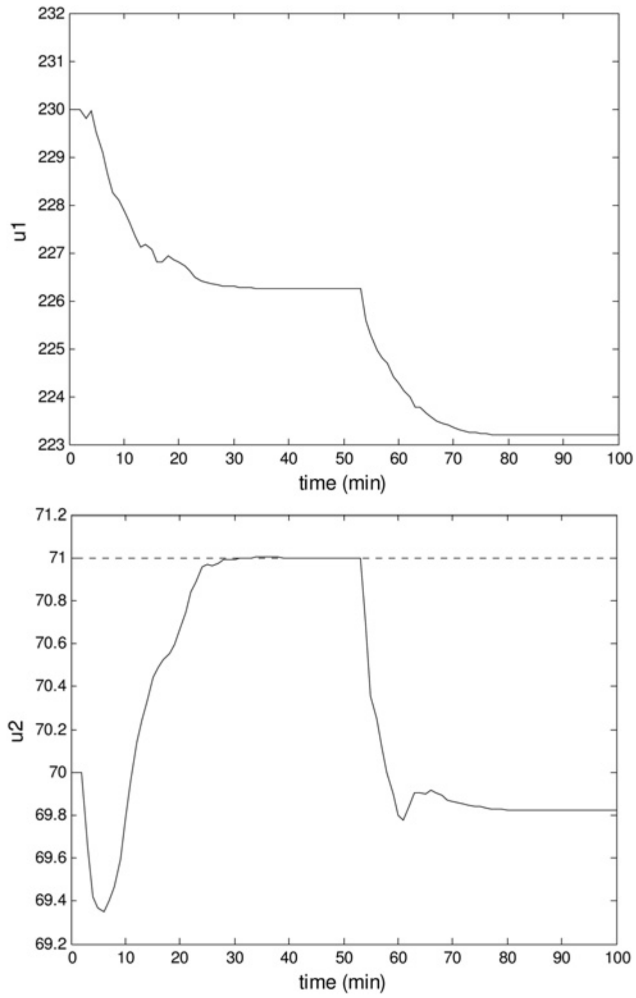

Fig. 5 *Inputs with LMI-MPC (——) and target (· · · · · ·), switching nominal model*

output. The system represented by model G_1 starts from the same initial steady state as in the first case and is driven by the controller to the desired targets smoothly and without offset since the steady state corresponding to the selected targets is reachable. At time step $k = 50$ min, the target of

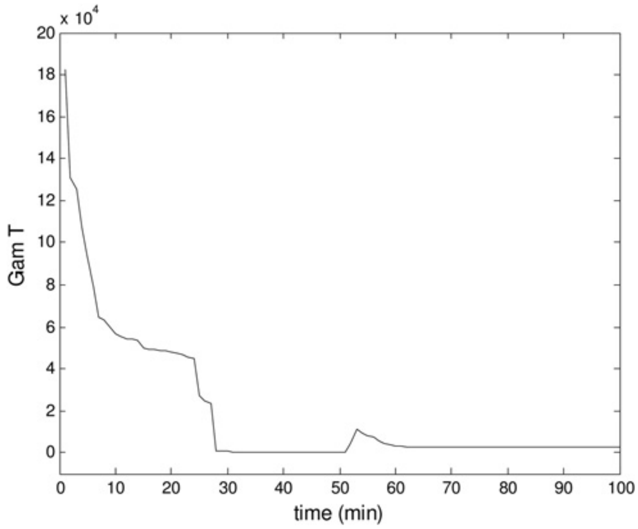

Fig. 6 *Upper bound to the cost function of the true plant with LMI-MPC, switching nominal model*

doi: 10.1049/iet-cta.2011.0216

output y_1 and the bounds of the remaining outputs are changed as shown in Fig. 4. Although the target to input u_2 remained the same, the new desired steady state becomes unreachable as outputs y_2 and y_3 tend to reach their minimum bounds. The closed loop is still stable but there is an offset in y_1 and u_2. To test the effect of switching between controllers that are based on different nominal models, the simulation is started with a controller based on model G_1, which means that in Problem P4, the objective function is the upper bound corresponding to model G_1. Then, at time step $k = 10$ min, the nominal model is switched to G_2 and, finally, at time step $k = 60$ min, the nominal model is switched to G_3. Fig. 6 shows $\gamma_{k,T}$ the upper bound to the true plant that is assumed to be G_1. One can easily see that $\gamma_{k,T}$ is strictly decreasing even when there is a switch in the nominal model considered by the controller. This interesting property of the proposed controller can be easily proved and allows the optimisation of the performance of the robust controller through the online selection of the most appropriate nominal model.

In the third simulation case presented here, the model uncertainty in the system gain is considered to be of the polytopic type and the LMI-based MPC is tested for the case where there are set-points to two outputs and the third output is controlled in a zone. To simulate this case, one considers models G_1, G_2^* and G_3^* where the denominator and time delays of the last two models are made equal to the denominator and time delays of model G_1 while the numerators are made equal to the numerators of models G_2 and G_3, respectively. The tuning parameters of the LMI-based MPC are the same as before, except for weights Q_u and S_u that are made equal to zero. Figs. 7 and 8 show the system responses when the true model is assumed to be the following

$$G_T = 0.3G_1 + 0.2G_2^* + 0.5G_3^*$$

Since the desired steady states defined through the output set-points are reachable, the LMI-based MPC can easily drive y_1

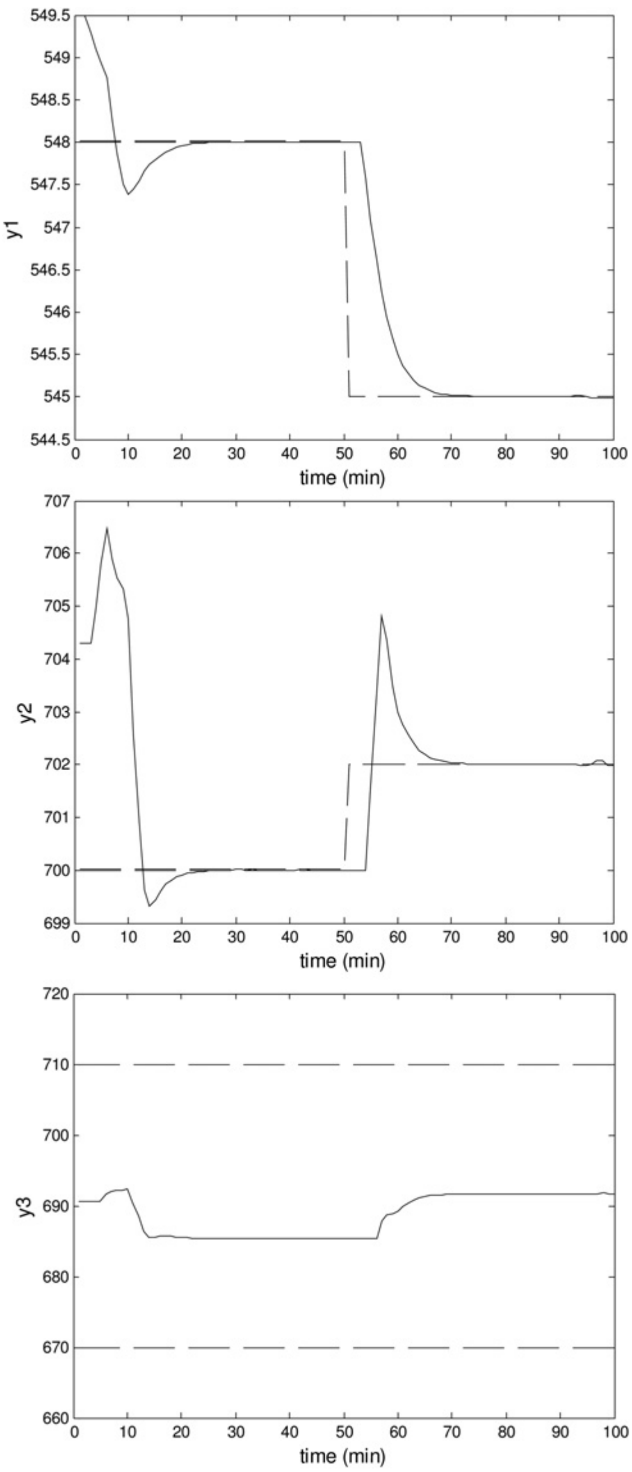

Fig. 7 *Outputs (——) and bounds (– –) with LMI-MPC and polytopic uncertainty*

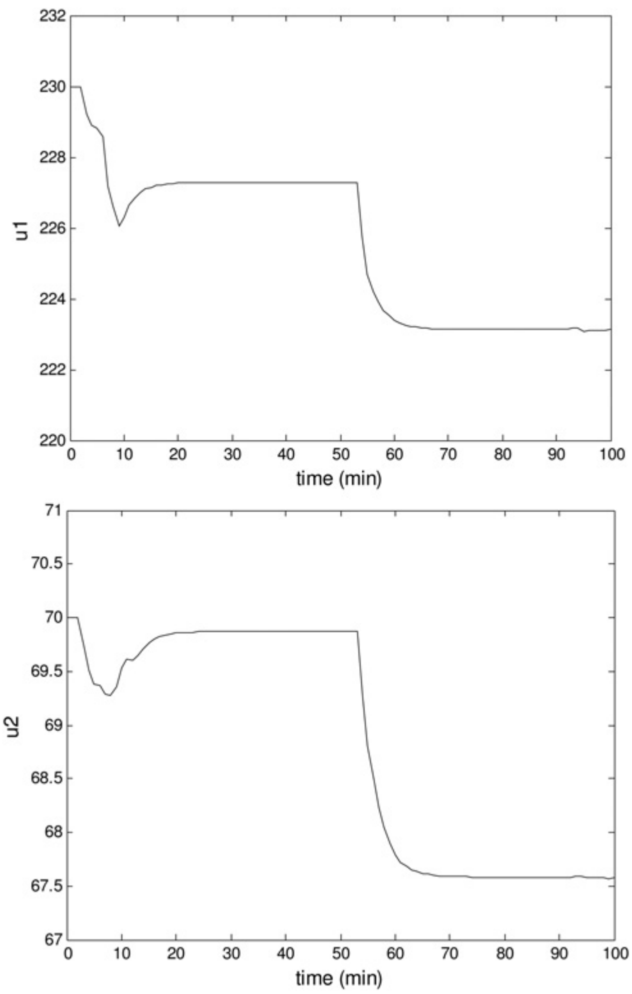

Fig. 8 *Inputs with LMI-MPC and polytopic uncertainty*

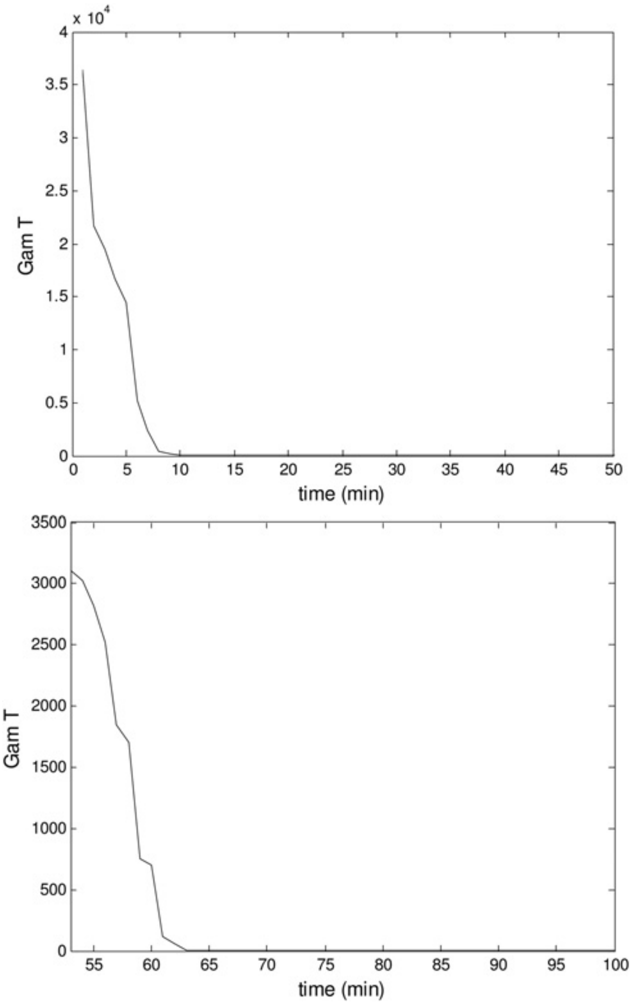

Fig. 9 *Upper bound to the cost function of the true plant with LMI-MPC and polytopic uncertainty*

and y_2 to their set-points without offset. Fig. 9 shows the bound to the cost of the true plant and it is clear that this bound is asymptotically decreasing and converges to zero, which shows, as demonstrated in Theorem 2 that the closed loop remains stable.

7 Conclusions

This paper addresses an LMI formulation to the problem of the robust MPC for systems with model uncertainty. It is shown that, for the multi-plant uncertainty case, the proposed approach is equivalent to the conventional method that is based on the solution of an NLP problem. The advantage of the LMI-based robust MPC proposed here over the NLP-based robust MPC proposed in [7] is a significant reduction in the computer time. A reduction of at least eighty percent in the computer time can be expected when the proposed controller is applied to multi-variable systems. The reduction of the computer effort allows the application of the LMI-based controller to larger systems when compared with the NLP-based controller. However, it remains to be investigated if the proposed controller can be implemented in all existing large-sized industrial processes.

It is also shown that the LMI approach extends the robust stability of the MPC to the polytopic uncertainty in the system gain and the method shows a good potential to be applied to real systems of the process industry.

8 References

1 Qin, S.J., Badgwell, T.A.: 'A survey of industrial model predictive control technology', *Control Eng. Pract.*, 2003, **11**, (7), pp. 733–764
2 Rotava, O., Zanin, A.C.: 'Multivariable control and real-time optimization – an industrial practical view', *Hydrocarbon Process.*, 2005, **84**, (6), pp. 61–71
3 Kothare, M.V., Balakrishnan, V., Morari, M.: 'Robust constrained model predictive control using linear matrix inequalities', *Automatica*, 1996, **32**, (10), pp. 1361–1379
4 Badgwell, T.A.: 'Robust model predictive control of stable linear systems', *Int. J. Control*, 1998, **68**, (4), pp. 797–818
5 Odloak, D.: 'Extended robust model predictive control', *AIChE J.*, 2004, **50**, (8), pp. 1824–1836
6 González, A.H., Odloak, D.: 'Robust model predictive control with zone control', *IET Control Theory Appl.*, 2009, **3**, (1), pp. 121–135
7 González, A.H., Odloak, D.: 'Robust model predictive control for time delayed systems with optimizing targets and zone control', in Bartoszewics, A. (Ed.): 'Robust control, theory and application' (INTECH, 2011), pp. 339–370
8 Cuzzola, F.A., Geromel, J.C., Morari, M.: 'An improved approach for constrained robust model predictive control', *Automatica*, 2002, **38**, (7), pp. 1183–1189
9 Mao, W.: 'Robust stabilization of uncertain time-varying discrete systems and comments on "An improved approach for constrained robust model predictive control"', *Automatica*, 2003, **39**, (6), pp. 1109–1112
10 Lee, S.M., Won, S.C., Park, J.H.: 'New robust model predictive control for uncertain systems with input constraints using relaxation matrices', *J. Optim. Theory Appl.*, 2008, **138**, (2), pp. 221–234
11 Jia, D., Krogh, B.H., Stursberg, O.: 'LMI approach to robust model predictive control', *J. Optim. Theory Appl.*, 2005, **127**, (2), pp. 347–365
12 Wan, Z., Kothare, M.V.: 'An efficient off-line formulation of robust model predictive control using linear matrix inequalities', *Automatica*, 2003, **39**, (5), pp. 837–846
13 Schuurmans, J., Rossiter, J.A.: 'Robust predictive control using tight set of predicted states', *IEE Proc. Control Theory Appl.*, 2000, **147**, (1), pp. 13–18
14 Ding, B., Xi, Y., Li, S.: 'A synthesis approach of on-line constrained robust model predictive control', *Automatica*, 2004, **40**, (1), pp. 163–167
15 Ding, B., Xie, L., Cai, W.: 'Robust MPC for polytopic uncertain systems with time-varying delays', *Int. J. Control*, 2008, **81**, (8), pp. 1239–1252
16 Ding, B.: 'Robust model predictive control for multiple time delay systems with polytopic uncertainty description', *Int. J. Control*, 2010, **83**, (9), pp. 1844–1857
17 Wu, F.: 'LMI-based robust model predictive control and its application to and its application to an industrial CSTR problem', *J. Process Control*, 2001, **11**, (6), pp. 649–659
18 González, A.H., Odloak, D.: 'A stable MPC with zone control', *J. Process Control*, 2009, **19**, (1), pp. 110–122
19 Sotomayor, O.A.Z., Odloak, D.: 'Observer-based fault diagnosis in chemical plants', *Chem. Eng. J.*, 2005, **112**, (1–3), pp. 93–108

IET Control Theory Appl., 2012, Vol. 6, Iss. 1, pp. 37–50
doi: 10.1049/iet-cta.2011.0216