Robust Disturbance Modeling for Model Predictive Control with Application to Multivariable Ill-conditioned Processes

Gabriele Pannocchia

Department of Chemical Engineering – University of Pisa
Via Diotisalvi, 2 – 56126 Pisa (Italy)
Tel.: +39 050 511238. Fax: +39 050 511266.
Email: pannocchia@ing.unipi.it

Submitted to Journal of Process Control on May 30, 2002.
First revision on August 28, 2002.
Second revision on November 28, 2002

Abstract

In this paper the disturbance model, used by MPC algorithms to achieve offset-free control, is optimally designed to enhance the robustness of single-model predictive controllers. The proposed methodology requires the off-line solution of a min-max optimization problem in which the disturbance model is chosen to guarantee the best closed-loop performance in the worst case of plant in a given uncertainty region. Application to a well-known ill-conditioned distillation column is presented to show that, for ill-conditioned processes, the use of a disturbance model that adds the correction term to the process inputs guarantees a robust performance, while the disturbance model that adds the correction term to the process output (used by industrial MPC algorithms) does not.

Key words: Disturbance modeling, MPC, Ill-conditioned systems

1 Introduction

Model Predictive Control (MPC) has become one of the most studied and applied control technique both in academia and in the process industries. The original industrial algorithms [1, 2] used finite impulse or step response models to predict the process behavior. An input sequence is computed from the minimization of a quadratic objective function that (usually) contains terms related to the error between the predicted output and its set point, and terms related to the amplitude of the control action. Input and output constraints can be included in the optimization problem, and the first control move of the optimal sequence is injected into the plant as input. The optimization is repeated at each sampling time because the model prediction is updated with a bias term computed from the difference between the actual and the predicted output. Despite its popularity, some of the assumptions on which the original formulations of MPC were based, limit the controller performance [3].

First of all, the use of convolution models limits the class of application to open-loop stable processes. Lee *et al.* [4] present a state-space formulation of DMC allowed to handle integrating processes, still using finite step response models. The easiest way to describe stable and unstable processes is, however, to use state-space models [5] or autoregressive models [6].

A disturbance model is used in MPC to achieve offset-free performance. A step disturbance is estimated from measured variables, and its effect on the controlled variables is removed by shifting the input and state targets. In most industrial MPC implementations a constant step disturbance is added to the measured process variables. While this method is acceptable for stable plants, it cannot be used if the plant is unstable because the observer contains the open-loop unstable poles. A disturbance model that adds step disturbances either to the state or to the process outputs is used by Muske and Badgwell [7]. This method is proven to remove offset when all the measured variables are controlled at a given set point. A general disturbance model for the case in which “some” of the measured variables are controlled at a given set point is proposed by Pannocchia and Rawlings [8].

The choice of the disturbance model may have a strong influence on the performance of MPC regulators. In fact, Shinskey [9] clearly pointed out that DMC is able to outperform PID controllers on set-point changes but not on load changes introduced upstream of a dominant lag. The problem of a sluggish rejection of slow disturbances in DMC is strictly related to the use of a constant bias term to update the model prediction. Lundström *et al.* [10] propose a solution to this problem by modeling the disturbance as a double integrated white noise sequence. Under this assumption, therefore, the disturbance is assumed to be a ramp with piece-wise constant slope. In the Generalized Predictive Control (GPC) framework, the disturbance is modeled by the choice of the observer polynomial (usually called $T(q^{-1})$) [11], which can be chosen to enhance the controller robustness. Typically, the polynomial T is chosen such that $1/T$ is a low-pass filter. Kouvaritakis *et al.* [12] and Yoon and Clarke [13] provide useful considerations for the choice of T. Prada *et al.* [14] present a comparative study of DMC and GPC, emphasizing the role of the polynomial T. Details for the multivariable extension of GPC can be found in [15, 16].

All the tuning considerations reported in the cited papers are given for SISO systems and they do not deal with multivariable ill-conditioned processes, which are rather frequent in the process industries, and in particular in chemical applications. Indeed, a great effort to face this problem can be found in the classical feedback PID [17, 18, 19, 20] and IMC frameworks [21, 22, 23]. A tight control of ill-conditioned processes, which show large interactions, requires an inversion of the process model. However, because of the ill conditioning, this inverse-based controller becomes sensitive to input uncertainties and plant-model mismatch [17]. DMC and other MPC algorithms suffer from sensitivity to uncertainties when the process is ill conditioned [10, 24] and, in order to avoid improper large control actions, it is common to increase the control move suppression factor [25].

In this paper the problem of robust control is addressed in the framework of state-space single-model predictive control by a suitable design of the disturbance model. A robust disturbance modeling procedure that can be applied to any kind of linear (square and non-square) multivariable systems is proposed. Application to ill-conditioned processes is presented in this paper because they are frequent in the process industries and for these systems a strong improvement in robustness can be accomplished by choosing a suitable disturbance model.

2 Problem formulation

2.1 Plant description

The plant is given by the following linear, discrete, time-invariant equations:

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k \ , \end{aligned} \tag{1}$$

in which $x \in \mathbb{R}^n$ is the state, $u \in \mathbb{R}^m$ is the manipulated input, $y \in \mathbb{R}^p$ is the measured output, and the matrices (A, B, C) have appropriate dimensions. In general the number of manipulated inputs (m) can be different from the number of measured outputs (p). Typically, a subset of the measured variables are controlled at a given set point. We denote the vector of the controlled variables as

$$z = Hy, \qquad \text{with } \dim z = n_c \ , \tag{2}$$

and assume that the following condition is satisfied (see *e.g.* [26, Corollary 3] and [8]):

$$\operatorname{rank} \begin{bmatrix} I - A & -B \\ HC & 0 \end{bmatrix} = n + n_c \ . \tag{3}$$

This condition implies that the number of controlled variables (n_c) cannot exceed either the number of manipulated variables (m) or the number of measured variables (p). Moreover, for square systems (*i.e.* when $n_c = m$) it implies that the gain matrix is non-singular.

Several descriptions of the uncertainty can be considered. Here we assume a polytopic plant description as in [27]. That is, the matrices (A, B) are unknown but constant and they belong to the convex hull of a number of pairs (A_1, B_1), (A_2, B_2), ..., (A_N, B_N). We denote this convex hull as Ω. The output matrix C, instead, is assumed to be known. It is also assumed that, for any pair $(A, B) \in \Omega$, (A, B) is stabilizable, (A, C) is detectable and (3) holds.

2.2 MPC algorithm

The regulator uses an MPC algorithm based on the following linear, discrete, time-invariant model:

$$\begin{aligned} \hat{x}_{k+1} &= A_m \hat{x}_k + B_m u_k + B_d \hat{d}_k \\ \hat{d}_{k+1} &= \hat{d}_k \\ \hat{y}_k &= C\hat{x}_k + C_d \hat{d}_k \ , \end{aligned} \tag{4}$$

in which the model matrices (A_m, B_m) belong to the convex hull previously defined. The vectors $\hat{x} \in \mathbb{R}^n$, $\hat{d} \in \mathbb{R}^p$ and $\hat{y} \in \mathbb{R}^p$ represent the model state, disturbance and output, respectively. The disturbance model matrices (B_d, C_d) have appropriate dimension and satisfy the following:

$$\text{rank} \begin{bmatrix} I - A_m & -B_d \\ C & C_d \end{bmatrix} = n + p \, , \tag{5}$$

which ensures that (4) is detectable [8]. A steady-state Kalman filter is used to estimate the current model state and disturbance from the plant measurements, y_k:

$$\begin{aligned} \hat{x}_{k|k} &= \hat{x}_{k|k-1} + L_x(y_k - C\hat{x}_{k|k-1} - C_d \hat{d}_{k|k-1}) \\ \hat{d}_{k|k} &= \hat{d}_{k|k-1} + L_d(y_k - C\hat{x}_{k|k-1} - C_d \hat{d}_{k|k-1}) \ , \end{aligned} \tag{6}$$

in which $L_x \in \mathbb{R}^{n \times p}$ and $L_d \in \mathbb{R}^{p \times p}$ are the steady-state Kalman filter gain matrices computed from the Riccati equation associated with (4). It is important to notice that the disturbance vector $\hat{d}$ is added to achieve integral control [8]. The modes associated with the disturbance vector are clearly non-controllable by the input u. However, since $\hat{d}$ is observable we use its estimate to remove its influence from the controlled variable, as discussed below.

Given the controlled variable set point $\bar{z} \in \mathbb{R}^{n_c}$ and the current disturbance estimate $\hat{d}_{k|k}$ we need to compute, at each sampling time, the input and state targets that drive the predicted controlled variable to this set point. If the system is square (*i.e.* $n_c = m$) and unconstrained these steady-state targets are unique, while if the dimension of manipulated variable is greater than the dimension of controlled variable (*i.e.* $m > n_c$) infinitely many combinations of the inputs can track the predicted controlled variable to the same set point. Therefore, we can search for the input target that is close to a given set point $\bar{u} \in \mathbb{R}^m$ and tracks the predicted controlled variable to its set point (notice that often the input set point, $\bar{u}$, is not specified it can be assumed zero in order to use, if possible, the optimal input in a least-squares sense). This can be done by solving the following quadratic program [5]:

$$\min_{x_t, u_t} (u_t - \bar{u})^T R_s (u_t - \bar{u}) \tag{7a}$$

subject to:

$$\begin{bmatrix} I - A_m & -B_m \\ HC & 0 \end{bmatrix} \begin{bmatrix} x_t \\ u_t \end{bmatrix} = \begin{bmatrix} B_d \hat{d}_{k|k} \\ -HC_d \hat{d}_{k|k} + \bar{z} \end{bmatrix} \tag{7b}$$

$$u_{\min} \leq u_t \leq u_{\max} \tag{7c}$$

$$y_{\min} \leq C_m x_t + C_d \hat{d}_{k|k} \leq y_{\max} \ , \tag{7d}$$

in which R_s is positive definite. We assume that (7) admits a feasible solution. If this is not satisfied, input and output constraints are too stringent, and the predicted controlled variable cannot be tracked to the set point $\bar{z}$ without offset. In this case we can solve a different quadratic program to find the state and input targets that minimize the steady-state offset in a least-square sense [5].

The constrained MPC algorithm used in this work is based on the solution of the following infinite horizon optimization problem:

$$\min_{\{u_j\}} \sum_{j=k}^{\infty} (H\hat{y}_j - \bar{z})^T Q (H\hat{y}_j - \bar{z}) + (u_j - u_{j-1})^T R (u_j - u_{j-1}) \tag{8a}$$

subject to:

$$\begin{cases} \hat{x}_k = \hat{x}_{k|k}, \quad \hat{d}_k = \hat{d}_{k|k} \\ \hat{x}_{j+1} = A_m \hat{x}_j + B_m u_j + B_d \hat{d}_j \\ \hat{d}_{j+1} = \hat{d}_j \\ \hat{y}_j = C\hat{x}_j + C_d \hat{d}_j \end{cases} \tag{8b}$$

$$u_{\min} \leq u_j \leq u_{\max} \tag{8c}$$

$$|u_j - u_{j-1}| \leq \Delta u_{\max} \tag{8d}$$

$$y_{\min} \leq \hat{y}_j \leq y_{\max} \ , \tag{8e}$$

in which Q and R are positive definite matrices of appropriate dimensions.

Given the targets computed from (7), the optimization problem (8) can be rewritten in a standard constrained LQR form by introducing the following change of variables (see *e.g.* [28] for more details):

$$\tilde{x}_j = \begin{bmatrix} \hat{x}_{k+j} - x_t \\ u_{k+j-1} - u_t \end{bmatrix}, \quad \tilde{u}_j = u_{k+j} - u_t, \quad \tilde{A} = \begin{bmatrix} A_m & 0 \\ 0 & 0 \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} B_m \\ I \end{bmatrix}$$

$$\tilde{C} = \begin{bmatrix} C & 0 \end{bmatrix}, \quad \tilde{Q} = \begin{bmatrix} C^T H^T Q H C & 0 \\ 0 & R \end{bmatrix}, \quad \tilde{R} = R, \quad \tilde{M} = \begin{bmatrix} 0 \\ -R \end{bmatrix}$$

$$\tilde{G} = \begin{bmatrix} I \\ I \end{bmatrix}, \quad \tilde{H} = \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix}, \quad \tilde{u}_{\max} = \begin{bmatrix} u_{\max} - u_t \\ \Delta u_{\max} \end{bmatrix}, \quad \tilde{u}_{\min} = \begin{bmatrix} u_{\min} - u_t \\ -\Delta u_{\max} \end{bmatrix}$$

$$\tilde{y}_{\max} = y_{\max} - C x_t - C_d \hat{d}_{k|k}, \quad \tilde{y}_{\min} = y_{\min} - C x_t - C_d \hat{d}_{k|k} \,.$$

Thus, (8) becomes:

$$\min_{\{\tilde{u}_j, \tilde{x}_j\}} \sum_{j=0}^{\infty} \tilde{x}_j^T \tilde{Q} \tilde{x}_j + \tilde{u}_j^T \tilde{R} \tilde{u}_j + 2 \tilde{x}_j^T \tilde{M} \tilde{u}_j \tag{9a}$$

subject to:

$$\tilde{x}_0 = \begin{bmatrix} \hat{x}_{k|k} - x_t \\ u_{k-1} - u_t \end{bmatrix}, \quad \tilde{x}_{j+1} = \tilde{A}\tilde{x}_j + \tilde{B}\tilde{u}_j \tag{9b}$$

$$\tilde{u}_{\min} \leq \tilde{G}\tilde{u}_j - \tilde{H}\tilde{x}_j \leq \tilde{u}_{\max} \tag{9c}$$

$$\tilde{y}_{\min} \leq \tilde{C}\tilde{x}_j \leq \tilde{y}_{\max} \,. \tag{9d}$$

Clearly, this formulation cannot be implemented as written because it involves an infinite number of decision variables and constraints. However, it is well known [29, 30] that this problem can be rewritten in a finite number of decision variables and constraints. In this work we use the method discussed in [28] to solve (9) in the presence of input and output constraints. The first component of the optimal solution of (9) is used as process input (after removing the change of variables), and the optimization is repeated at the next sampling time, when a new state and disturbance estimate is available and, consequently, new input and state targets are computed.

It is also clear that the first component of the unconstrained solution of (9) (that is (9a) subject to (9b) only) is the standard linear quadratic control law, *i.e.*:

$$\tilde{u}_0 = K\tilde{x}_0, \tag{10}$$

in which $K \in \mathbb{R}^{m\times(n+m)}$ is the optimal LQR gain matrix computed from the Riccati equation associated with (9).

3 Robust disturbance modeling method

In this section we propose a method for finding the disturbance model, *i.e.* the matrices (B_d, C_d), that guarantees the best unconstrained closed-loop performance for the worst case of plant within the uncertainty region. In practice, even the nominal model matrices (A_m, B_m) may not be known. Indeed, we can search for the nominal model that guarantees (along with the disturbance model) the best closed-loop performance for the worst case of plant within the uncertainty region. Therefore, in general, we consider the case in which both nominal model and disturbance model are to be chosen.

3.1 Unconstrained closed-loop system

Let the optimal LQR gain matrix K in (10) be partitioned as

$$K = \begin{bmatrix} K_x & K_u \end{bmatrix}, \qquad K_x \in \mathbb{R}^{m\times n}, \quad K_u \in \mathbb{R}^{m\times m}.$$

Thus, the optimal unconstrained input as in (10) can be rewritten as

$$u_k = u_t + K_x(\hat{x}_{k|k} - x_t) + K_u(u_{k-1} - u_t), \tag{11}$$

in which the state and input targets are computed from (7) without including input and output constraints, that is from (7a) subject to (7b) only (notice that this equality-constrained quadratic program is always feasible because of (3)). When the feedback law in (11) is applied to the process (1), we have that the closed-loop system has the following linear form:

$$w_{k+1} = \Lambda w_k + \Gamma v, \qquad w_k = \begin{bmatrix} x_k \\ \hat{x}_{k|k-1} \\ \hat{d}_{k|k-1} \\ u_{k-1} \end{bmatrix}, \quad v = \begin{bmatrix} \bar{z} \\ \bar{u} \end{bmatrix}, \tag{12}$$

in which the matrices Λ and Γ are shown in Appendix A.1. It is interesting to notice that the closed-loop system is stable if and only if the matrix Λ is strictly stable.

By using (12) and (11) we can easily compute the true closed-loop objective function

$$\Phi = \sum_{k=0}^{\infty} (z_k - \bar{z})^T Q (z_k - \bar{z}) + (u_k - u_{k-1})^T R (u_k - u_{k-1}) \tag{13}$$

as

$$\Phi = \xi_0^T \mathcal{P} \xi_0, \qquad \text{with } \xi_0 = \begin{bmatrix} w_0 \\ v \end{bmatrix}. \tag{14}$$

See Appendix A.2 for the definition of the positive semi-definite matrix $\mathcal{P}$.

3.2 Min-max optimization

The nominal model and disturbance model are chosen as the solution of the following min-max optimization problem:

$$\min_{\substack{(A_m,B_m)\in\Omega \\ B_d,C_d}} \quad \max_{\substack{(A,B)\in\Omega \\ \xi_0}} \; \Phi = \xi_0^T \mathcal{P} \xi_0 \,. \tag{15}$$

The matrices (A_m, B_m) and (A, B) can be easily expressed as

$$\begin{aligned} A_m &= \sum_{j=1}^{N} \mu_j^m A_j, \quad B_m = \sum_{j=1}^{N} \mu_j^m B_j, &\qquad \mu_j^m \geq 0, \quad \sum_{j=1}^{N} \mu_j^m = 1 \\ A &= \sum_{j=1}^{N} \mu_j A_j, \quad B = \sum_{j=1}^{N} \mu_j B_j, &\qquad \mu_j \geq 0, \quad \sum_{j=1}^{N} \mu_j = 1 \,. \end{aligned}$$

Similarly, the disturbance model matrices (B_d, C_d) are chosen as a convex combination of a number of disturbance models. That is

$$B_d = \sum_{j=1}^{L} \eta_j B_j^d, \quad C_d = \sum_{j=1}^{L} \eta_j C_j^d, \quad \eta_j \geq 0, \quad \sum_{j=1}^{L} \eta_j = 1 \,,$$

in which (B_j^d, C_j^d) are L pairs of matrices of appropriate dimension defined by the user. Given the vectors μ, μ^m, η, the matrix $\mathcal{P}$ is computed by solving a Lyapunov equation as reported in Appendix A.2. The augmented vector ξ_0 includes the initial closed-loop state $w_0 = \begin{bmatrix} x_0^T & \hat{x}_0^T & \hat{d}_0^T & u_{-1}^T \end{bmatrix}^T$ and the exogenous term $v = \begin{bmatrix} \bar{z}^T & \bar{u}^T \end{bmatrix}^T$. In general, all the terms that define the initial augmented state ξ_0 can be included in the maximization. However, for simplicity we only include the initial plant state x_0 and the controlled variable set point $\bar{z}$, since they are more likely to vary. It is also important to notice that stability of the unconstrained closed-loop system (12) does not depend on the initial conditions, but only on the existence of $\mathcal{P}$ (see Appendix A.2). Thus, the min-max problem (15) is rewritten as:

$$\min_{\mu^m, \eta} \quad \max_{\mu, x_0, \bar{z}} \ \Phi = \xi_0^T \mathcal{P}(\mu, \mu^m, \eta) \xi_0 \tag{16a}$$

subject to:

$$\mu_j^m \geq 0, \quad \sum_{j=1}^{N} \mu_j^m = 1, \qquad \eta_j \geq 0, \quad \sum_{j=1}^{L} \eta_j = 1, \qquad \mu_j \geq 0, \quad \sum_{j=1}^{N} \mu_j = 1 \tag{16b}$$

$$\|x_0\|_\infty \leq r_x, \qquad \|z\|_\infty \leq r_z \ , \tag{16c}$$

in which r_x, r_z are non-negative scalars. In this way, (16) is a nonlinear optimization problem subject to linear constraints, and can be solved by using SQP (Sequential Quadratic Programs) optimizers for both the maximization and the minimization problems. In particular, we use an SQP algorithm with trust region as described by Nocedal and Wright [31].

3.3 Comments and remarks

In the proposed method the nominal model and disturbance model are chosen to obtain the best closed-loop performance in the worst case of plant within the uncertainty region and the worst case of initial conditions (plant state and controlled variable set point), and this search is accomplished without considering the effect of the constraints because in this way we have a simple relation to evaluate the true closed-loop objective function. However, once the nominal model and disturbance model are chosen, a constrained MPC algorithm as described in Section 2.2 is used. Clearly, this does not guarantee that the chosen models are optimal in the presence of constraints. However, if a disturbance model is optimal in the unconstrained case the corrected model prediction and the actual plant response are similar over the uncertainty region (*i.e.* the reconstruction error $e_k = y_k - \hat{y}_k$ quickly goes to zero with time). Thus, the resulting controller is expected to perform reasonably well even in the presence of constraints because it is based on reliable model predictions. Moreover, it is important to remark that input and output constraints do not affect the state and disturbance estimator (6). Indeed, constrained simulations show that this method provides a practical robust design of single-model predictive controllers, which are the most common advanced regulators used in the process industries.

Most industrial MPC algorithms use $B_d = 0$ and $C_d = I$, which is often called "output disturbance model" as the disturbance is assumed to be a constant step added to the process output. For ill-conditioned processes they suffer from sensitivity to modeling errors and in particular to input uncertainties, because small errors on the process inputs are amplified by the ill conditioning. In fact, for such processes it is frequent to increase the input rate-of-change penalty R [25], thus slowing down the control action and making the controller less sensitive to uncertainties. Here, instead, we show that one can still use a relatively low input penalty for fast performance, and one can make the controller robust by using a more appropriate disturbance model. As shown in the next section, by solving the min-max problem (16) we find that for ill-conditioned processes a more robust choice for the disturbance model is $B_d = B$ and $C_d = 0$, which is often called "input disturbance model" as the disturbance is assumed to be a constant step added to the process input. In fact, in this way the additional disturbance is able to lump the input uncertainties quickly thus making the model prediction correct. On the other hand the output disturbance model corrects the input uncertainties only after a long transient in which relatively small differences between the plant response and the model prediction are source of performance degradation because they are amplified by the ill conditioning.

It is also important to point out that the optimization problem (16) is, in general, non-convex, so that there is no guarantee that a global solution is found. Once again, this limitation appears to be more theoretical than practical, because the problem (16) can be solved starting from several initial guesses, and the "best" local optimum can be chosen. In fact, it is important to mention that there are no concerns about the computational cost for solving (16) since this problem needs to be solved off line.

4 Application example

4.1 Plant description

The well known ill-conditioned LV distillation column presented by Morari and coworkers [17, 21] is chosen as case study. The nominal transfer function matrix is

$$G(s) = \frac{1}{75s+1} \begin{bmatrix} 0.878 & -0.864 \\ 1.082 & -1.096 \end{bmatrix} . \tag{17}$$

This process is ill conditioned since the steady-state condition number is 142. Choosing a sampling time $T_s = 5\,\text{min}$, a state-space minimal representation of the nominal system is found, which is defined by the matrices (A, B, C) in which the dimension of the state is $n = 2$.

We consider bounded input uncertainties, *i.e.* the actual process is

$$G_{\text{actual}} = G(s) \begin{bmatrix} 1+\delta_1 & 0 \\ 0 & 1+\delta_2 \end{bmatrix} \qquad \text{with } |\delta_1|, |\delta_2| \leq 0.2 \, . \tag{18}$$

This uncertainty region is a convex hull Ω defined by four pairs of matrices (A_1, B_1), $\ldots$, (A_4, B_4) not shown for the sake of space. Notice that the nominal case (17) is the center of the convex hull and corresponds to choosing the plant weight vector as $\mu = \begin{bmatrix} 0.25 & 0.25 & 0.25 & 0.25 \end{bmatrix}$.

4.2 "Optimal" regulator design

The disturbance model is chosen as a convex combination of the following "candidate" disturbance models (input and output disturbance model, respectively):

$$B_1^d = B, \quad C_1^d = 0_{2\times 2}, \qquad \text{and } B_2^d = 0_{2\times 2}, \quad C_2^d = I_{2\times 2} \, . \tag{19}$$

Zero output and state noise is assumed to tune the estimator (standard DMC-like tuning), and the regulator tuning matrices are $Q = 500 I_{2\times 2}$ and $R = I_{2\times 2}$.

We solve the optimization problem (16) with $r_x = 0$ and $r_z = 1$, obtaining the following (local) optimum vectors μ^m and η, which define the nominal model and the disturbance model, respectively:

$$\mu^m = \begin{bmatrix} 0.249 & 0.251 & 0.249 & 0.251 \end{bmatrix}, \quad \eta = \begin{bmatrix} 1.000 & 0.000 \end{bmatrix} .$$

It is interesting to notice that the "optimal" nominal model nearly corresponds to (17), while the "optimal" disturbance model corresponds to the input disturbance model.

4.3 Closed-loop simulations

Two model predictive controllers, based on the same "optimal" nominal model but different disturbance model, are compared in a set-point change $\bar{z} = \begin{bmatrix} -0.781 & .625 \end{bmatrix}^T$, which corresponds to the unfavorable direction for this plant. MPC 1 uses the optimal disturbance model obtained with the proposed method, while MPC 2 uses the output disturbance model. Both controllers are tuned as specified in the previous paragraph.

In Figures 1 and 2 we report the unconstrained simulation results (outputs and inputs, respectively) for the worst case of plant in the uncertainty region, which corresponds to the plant matrices (A_2, B_2), *i.e.* input uncertainties of opposite sign. The closed-loop objective function is 5182 in the nominal case for both controllers, while it is 6080 for MPC 1 and 32435 for MPC 2 in the worst uncertainty case.

In Figures 3 and 4 we report the simulation results for the worst case of plant in the uncertainty region in the presence of the following input constraints:

$$\begin{bmatrix} -70 \\ -70 \end{bmatrix} \leq u_k \leq \begin{bmatrix} 70 \\ 70 \end{bmatrix}, \qquad \begin{bmatrix} -10 \\ -10 \end{bmatrix} \leq u_k - u_{k-1} \leq \begin{bmatrix} 10 \\ 10 \end{bmatrix} .$$

The closed-loop objective function is 5267 in the nominal case for both controllers, while it is 6353 for MPC 1 and 27565 for MPC 2 in the worst uncertainty case.

5 Conclusions

MPC algorithms use a disturbance model to lump the sources of plant-model mismatch and achieve offset-free control. Most industrial implementations of Model Predictive Control (*e.g.* DMC) use an "output disturbance model" to correct the model prediction in the presence of plant-model mismatch, and for ill-conditioned processes this is not a very robust choice.

In this work, it has been shown that the degrees of freedom given by the disturbance model allows the designer to increase the robustness of MPC algorithms significantly without having to modify the computational complexity of the on-line implementation. To this aim, a general methodology for designing a disturbance model that is able to improve the robustness properties of predictive controllers has been proposed, which requires the off-line solution of a min-max optimization problem. The method is based on a state-space realization of the unconstrained closed-loop system, which permits one to evaluate the implications of the choice of the disturbance model on the closed-loop performance over the plant uncertainty region. The proposed algorithm also permits one to choose the nominal model, used by the regulator, within the plant uncertainty region.

Application to a well-known ill-conditioned distillation column has been presented to show that for ill-conditioned processes a great improvement in robust performance can be achieved by adopting a more appropriate disturbance model, which for the case study and other examples that we tested is (close to) the "input disturbance model". This occurs because the input disturbance model corrects quickly the input uncertainties to which the common industrial MPC algorithms are sensitive. That is, the input disturbance model makes the reconstruction error (*i.e.* the difference between the actual plant output and the model prediction) go to zero in a short period of time. Moreover, the input disturbance model describes in a more realistic way the unmeasured disturbances, which generally enter upstream of a dominant lag [9], thus making more efficient the rejection of unmeasured disturbances [7]. Finally, results have also shown that for robust control of ill-conditioned processes the choice of the nominal model is not as important as the choice of the disturbance model.

6 Acknowledgments

The author thanks James B. Rawlings and Matthew J. Tenny from the University of Wisconsin – Madison for helpful suggestions in the early stage of this work and anonymous reviewers for useful comments.

A Complements to Section 3

A.1 Closed-loop system matrices

The closed-loop matrices Λ and Γ are as follows:

$$\Lambda = \begin{bmatrix} \Lambda_{11} & \Lambda_{12} & \Lambda_{13} & \Lambda_{14} \\ \Lambda_{21} & \Lambda_{22} & \Lambda_{23} & \Lambda_{24} \\ \Lambda_{31} & \Lambda_{32} & \Lambda_{33} & \Lambda_{34} \\ \Lambda_{41} & \Lambda_{42} & \Lambda_{43} & \Lambda_{44} \end{bmatrix}, \qquad \Gamma = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} \\ \Gamma_{21} & \Gamma_{22} \\ \Gamma_{31} & \Gamma_{32} \\ \Gamma_{41} & \Gamma_{42} \end{bmatrix},$$

in which

$$\begin{aligned}
\Lambda_{11} &= A + B(K_x L_x + \alpha L_d)C \\
\Lambda_{12} &= B\left(K_x(I - L_x C) - \alpha L_d C\right) \\
\Lambda_{13} &= B\left(-K_x L_x C_d + \alpha(I - L_d C_d)\right) \\
\Lambda_{14} &= B K_u \\
\Lambda_{21} &= (A_m + B_m K)L_x C + (B_d + B_m \alpha)L_d C \\
\Lambda_{22} &= (A_m + B_m K)(I - L_x C) - (B_d + B_m \alpha)L_d C \\
\Lambda_{23} &= -(A_m + B_m K)L_x C_d + (B_d + B_m \alpha)(I - L_d C_d) \\
\Lambda_{24} &= B_m K_u \\
\Lambda_{31} &= L_d C \\
\Lambda_{32} &= -L_d C \\
\Lambda_{33} &= I - L_d C_d \\
\Lambda_{34} &= 0 \\
\Lambda_{41} &= (K_x L_x + \alpha L_d)C \\
\Lambda_{42} &= K_x(I - L_x C) - \alpha L_d C \\
\Lambda_{43} &= -K_x L_x C_d + \alpha(I - L_d C_d) \\
\Lambda_{44} &= K_u \,,
\end{aligned}$$

$$\begin{aligned}
&\Gamma_{11} = B\beta && \Gamma_{12} = B\gamma R_s && \Gamma_{21} = B_m\beta && \Gamma_{22} = B_m\gamma R_s \\
&\Gamma_{31} = 0 && \Gamma_{32} = 0 && \Gamma_{41} = \beta && \Gamma_{42} = \gamma R_s \,,
\end{aligned}$$

$$\alpha = (I - K_u)(\theta_{23}B_d - \theta_{24}HC_d) - K_x(\theta_{13}B_d - \theta_{14}HC_d)$$
$$\beta = (I - K_u)\theta_{24} - K_x\theta_{14}$$
$$\gamma = (I - K_u)\theta_{22} - K_x\theta_{12} \, ,$$

$$\Theta^{-1} = \begin{bmatrix} 0 & 0 & I - A_m^T & C^T H^T \\ 0 & R_s & -B_m^T & 0 \\ I - A_m & -B_m & 0 & 0 \\ HC & 0 & 0 & 0 \end{bmatrix} \Rightarrow \Theta = \begin{bmatrix} \theta_{11} & \theta_{12} & \theta_{13} & \theta_{14} \\ \theta_{21} & \theta_{22} & \theta_{23} & \theta_{24} \\ \theta_{31} & \theta_{32} & \theta_{33} & \theta_{34} \\ \theta_{44} & \theta_{42} & \theta_{43} & \theta_{44} \end{bmatrix} .$$

A.2 True closed-loop objective function

The penalty matrix $\mathcal{P}$ in (14) is the solution of the following Lyapunov equation:

$$\mathcal{P} = \mathcal{Q} + \mathcal{A}^T \mathcal{P} \mathcal{A} \, ,$$

in which

$$\mathcal{A} = \begin{bmatrix} \Lambda & \Gamma \\ 0 & I \end{bmatrix} , \quad \mathcal{Q} = \begin{bmatrix} \mathcal{C}^T Q \mathcal{C} + \mathcal{E}^T R \mathcal{E} & \mathcal{D}^T Q \mathcal{C} + \mathcal{F}^T R \mathcal{E} \\ \mathcal{C}^T Q \mathcal{D} + \mathcal{E}^T R \mathcal{F} & \mathcal{D}^T Q \mathcal{D} + \mathcal{F}^T R \mathcal{F} \end{bmatrix}$$
$$\mathcal{C} = \begin{bmatrix} HC & 0 & 0 & 0 \end{bmatrix} , \quad \mathcal{D} = \begin{bmatrix} I & 0 \end{bmatrix} , \quad \mathcal{E} = \begin{bmatrix} \Lambda_{41} & \Lambda_{42} & \Lambda_{43} & \Lambda_{44} - I \end{bmatrix} , \quad \mathcal{F} = \begin{bmatrix} \Gamma_{41} & \Gamma_{42} \end{bmatrix} .$$

Notice that $\mathcal{P}$ is function of the nominal model matrices (A_m, B_m), of the disturbance model matrices (B_d, C_d), of the actual plant matrices (A, B). Also, notice that the matrix $\mathcal{P}$ is defined if and only if the matrix Λ is strictly stable.

References

[1] J. Richalet, J. Rault, J. L. Testud, J. Papon, Model predictive heuristic control: applications to industrial processes, Automatica 14 (1978) 413–1554.

[2] C. R. Cutler, B. L. Ramaker, Dynamic matrix control – a computer algorithm, in: AIChE 86th National Meeting, Houston, TX, 1979.

[3] M. Morari, J. H. Lee, Model predictive control: the good, the bad and the ugly, in: Fourth International Conference on Chemical Process Control, Elsevier, 1991, pp. 419–444.

[4] J. H. Lee, M. Morari, C. E. Garcia, State-space interpretation of model predictive control, Automatica 30 (1994) 707–717.

[5] K. R. Muske, J. B. Rawlings, Model predictive control with linear models, AIChE J. 39 (1993) 262–287.

[6] D. W. Clarke, C. Mohtadi, P. S. Tuffs, Generalized predictive control - Part I. The basic algorithm, Automatica 23 (1987) 137–148.

[7] K. R. Muske, T. A. Badgwell, Disturbance modeling for offset-free linear model predictive control, J. Proc. Cont. 12 (2002) 617–632.

[8] G. Pannocchia, J. B. Rawlings, Disturbance models for offset-free model predictive control, AIChE J. in press.

[9] F. G. Shinskey, Feedback Controllers for the Process Industries, McGraw-Hill, Inc, 1994.

[10] P. Lundström, J. H. Lee, M. Morari, S. Skogestad, Limitations of dynamic matrix control, Comput. Chem. Eng. 19 (1995) 409–421.

[11] D. W. Clarke, C. Mohtadi, Properties of generalized predictive control, Automatica 25 (1989) 859–875.

[12] B. Kouvaritakis, J. A. Rossiter, A. O. T. Chang, Stable generalised predictive control: an algorithm with guaranteed stability, IEE Proc.–D 139 (1992) 349–362.

[13] T.-W. Yoon, D. W. Clarke, Observer design in receding-horizon predictive control, Int. J. Control 61 (1995) 171–191.

[14] C. Prada, J. Serrano, P. Vega, M. A. Piera, A comparative study of DMC and GPC controllers, in: D. W. Clarke (Ed.), Model-Based Predictive Control, Oxford, 1994, pp. 38–52.

[15] D. J. Cloud, B. Kouvaritakis, Characteristic decomposition and the multivariable generalization of predictive self-tuning control, IEE Proc.–D 135 (1988) 161–181.

[16] C. Mohtadi, S. L. Shah, D. G. Fisher, Frequency response characteristic of MIMO GPC, Int. J. Control 55 (1992) 877–900.

[17] S. Skogestad, M. Morari, Implication of large RGA-elements on control performance, Ind. Eng. Chem. Res. 26 (1987) 2323–2330.

[18] A. Brambilla, L. D'Elia, Multivariable controller for distillation columns in the presence of strong directionality and model errors, Ind. Eng. Chem. Res. 31 (1992) 536–543.

[19] M. Hovd, S. Skogestad, Simple frequency-dependent tools for control system analysis, structure selection and design, Automatica 28 (1992) 989–996.

[20] D. Semino, G. Pannocchia, Robust multivariable inverse-based controllers: Theory and application, Ind. Eng. Chem. Res. 38 (1999) 2375–2382.

[21] M. Morari, E. Zafiriou, Robust Process Control, Prentice Hall, 1989.

[22] J.-W. Chang, C.-C. Yu, Failure conditions for inverse-based multivariable controllers: Internal Model Control structure, Chem. Eng. Sci. 47 (1992) 2047–2056.

[23] D. Semino, C. Scali, A. Brambilla, Design of double filter IMC controllers for ill-conditioned distillation columns, Comput. Chem. Eng. 17 (1993) 1041–1046.

[24] G. Pannocchia, D. Semino, Optimal modified models for robust predictive controllers, in: 14th IFAC World Congress, Vol. N-7, Beijing, China, 1999, pp. 157–162.

[25] S. J. Qin, T. A. Badgwell, An overview of industrial model predictive control technology, in: J. C. Kantor, C. E. Garcia, B. Carnahan (Eds.), Fifth International

Conference on Chemical Process Control, AIChE Symposium Series 316, 93, 1997, pp. 232–256.

[26] E. J. Davison, H. W. Smith, Pole assignment in linear time-invariant multivariable systems with constant disturbances, Automatica 7 (1971) 489–498.

[27] M. V. Kothare, V. Balakrishnan, M. Morari, Robust constrained model predictive control using linear matrix inequalities, Automatica 32 (1996) 1361–1379.

[28] J. B. Rawlings, Tutorial: Model predictive control technology, in: American Control Conference, San Diego, CA (USA), 1999, pp. 662–676.

[29] D. Chmielewski, V. Manousiouthakis, On constrained infinite-time linear quadratic optimal control, Sys. Cont. Let. 29 (1996) 121–129.

[30] P. O. M. Scokaert, J. B. Rawlings, Constrained linear quadratic regulation, IEEE Trans. Auto. Cont. 43 (1998) 1163–1169.

[31] J. Nocedal, S. J. Wright, Numerical Optimization, Springer, 1999.

Figure captions

Figure 1: Unconstrained closed-loop responses (outputs)

Figure 2: Unconstrained closed-loop responses (inputs)

Figure 3: Constrained closed-loop responses (outputs)

Figure 4: Constrained closed-loop responses (inputs)

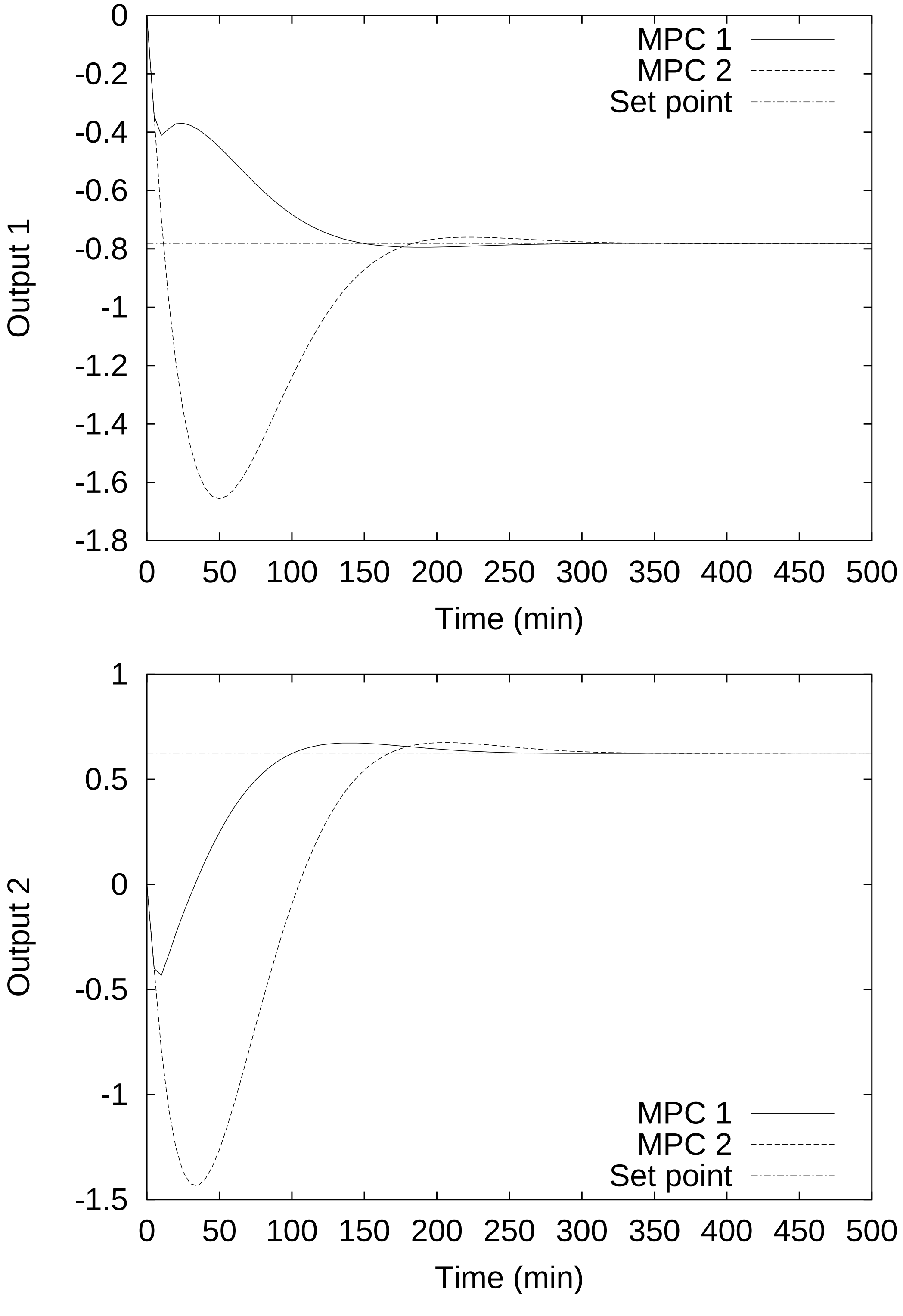
0
-0.2
-0.4
-0.6
-0.8
-1
-1.2
-1.4
-1.6
-1.8
Output 1
MPC 1
MPC 2
Set point
0
50
100
150
200
250
300
350
400
450
500
Time (min)
1
0.5
0
-0.5
-1
-1.5
Output 2
MPC 1
MPC 2
Set point
0
50
100
150
200
250
300
350
400
450
500
Time (min)

Figure 1

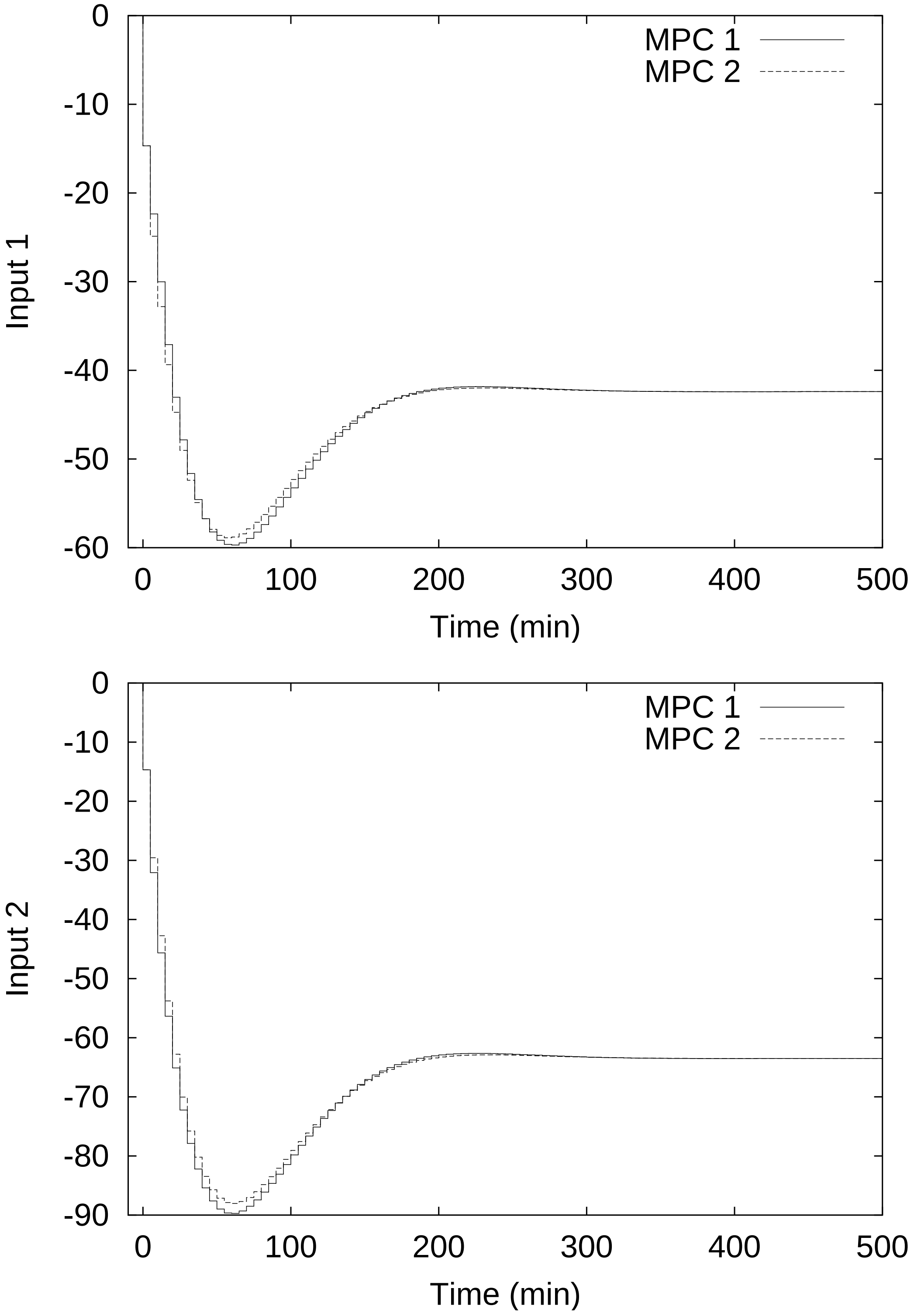
Input 1
0
-10
-20
-30
-40
-50
-60
0
100
200
300
400
500
Time (min)
MPC 1
MPC 2
Input 2
0
-10
-20
-30
-40
-50
-60
-70
-80
-90
0
100
200
300
400
500
Time (min)
MPC 1
MPC 2

Figure 2

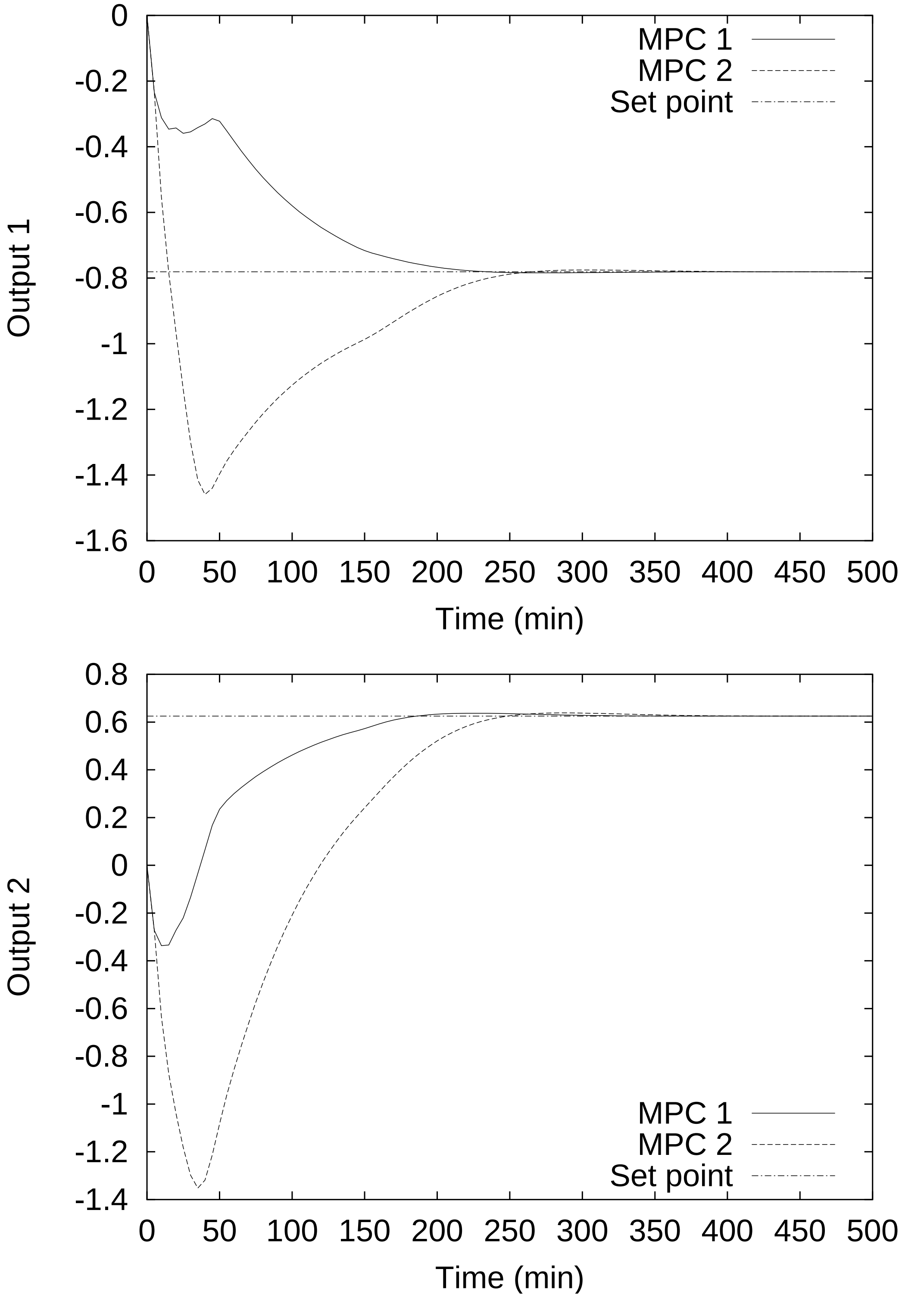
Output 1
0
-0.2
-0.4
-0.6
-0.8
-1
-1.2
-1.4
-1.6
0 50 100 150 200 250 300 350 400 450 500
Time (min)
MPC 1
MPC 2
Set point
Output 2
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1.2
-1.4
0 50 100 150 200 250 300 350 400 450 500
Time (min)
MPC 1
MPC 2
Set point

Figure 3

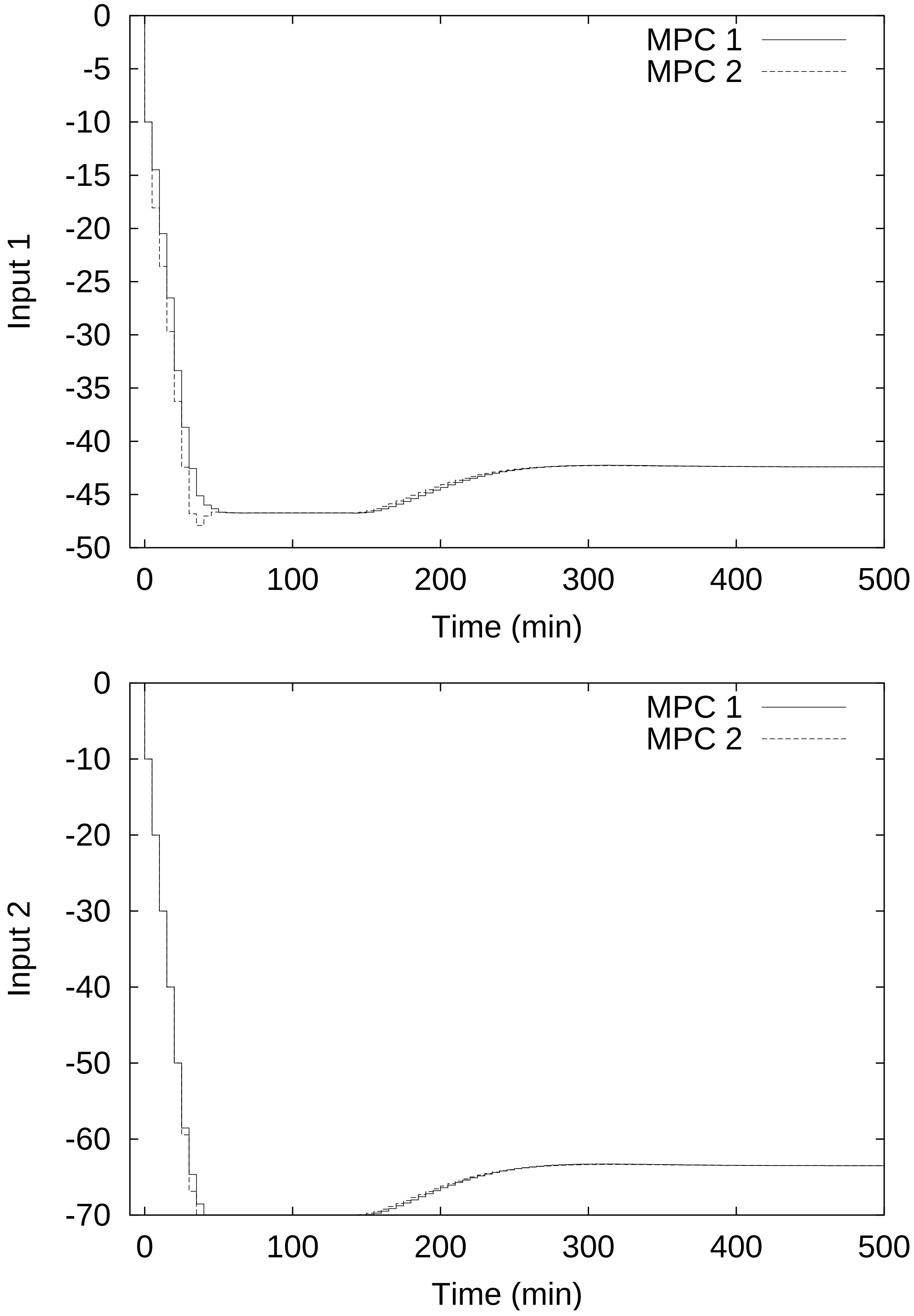
MPC 1
MPC 2
Input 1
0
-5
-10
-15
-20
-25
-30
-35
-40
-45
-50
0
100
200
300
400
500
Time (min)
MPC 1
MPC 2
Input 2
0
-10
-20
-30
-40
-50
-60
-70
0
100
200
300
400
500
Time (min)

Figure 4