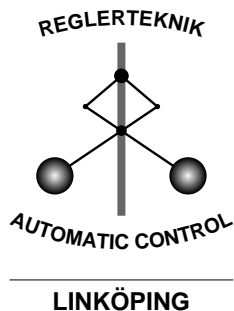


Minimax MPC for systems with uncertain input gain – revisited

Johan Löfberg

Department of Electrical Engineering
Linköpings universitet, SE-581 83 Linköping, Sweden
WWW: <http://www.control.isy.liu.se>
Email: johanl@isy.liu.se

June 20, 2001



Report no.: LiTH-ISY-R-2367

Technical reports from the Automatic Control group in Linköping are available by anonymous ftp at the address [ftp.control.isy.liu.se](ftp://ftp.control.isy.liu.se). This report is contained in the file 2367.pdf.

Minimax MPC for systems with uncertain input gain – revisited

Johan Löfberg

June 20, 2001

Abstract

Robust synthesis is one of the remaining challenges in model predictive control (MPC). One way to robustify an MPC controller is to formulate a minimax problems, i.e., optimize a worst-case performance measure. For systems modeled with an uncertain input gain, there are many results available. Typically, the minimax formulations have given intractable problems, or unorthodox performance measures have been used to obtain tractable problems. In this paper, we show how the standard quadratic performance measure can be used in a computationally tractable minimax MPC controller. The controller is developed in a linear matrix inequality framework that easily allows extensions and generalizations. Some of these extensions are pointed out.

Keywords: MPC, Robust control, Minimax, LMI

1 Introduction

Despite the tremendous amount of results and research on robust control during the last decades, model predictive control (MPC) still suffers from a lack of general and tractable results on robust synthesis. Many interesting approaches based on minimax (worst-case) optimization have been proposed, but they often come with some drawback such as computationally intractable [LC97, CGM99, SR00], conservativeness [KBM94], use of non-standard performance measures [CM87, AP91, OAFD00] or restriction to stable systems [Zhe95].

In this paper, we present an approach that solves minimax MPC with a traditional quadratic performance measure. The approach is not limited to stable systems, and can be implemented efficiently with semidefinite programming.

2 Problem formulation

A problem setup that has been used in many approaches to robust MPC is models with an uncertain input gain

$$x(k+1) = Ax(k) + B(k)u(k) \quad (1)$$

The time-varying uncertainty in $B(k)$ can be modeled in various ways. A common choice has been a polytopic model $B(k) \in \mathbf{Co}(B_1, \dots, B_q)$, i.e.,

$$B(k) = \sum_{i=1}^q \lambda_i B_i, \quad \sum_{i=1}^q \lambda_i = 1, \quad \lambda_j \geq 0 \quad (2)$$

In this work, we will turn our attention to a different model, a so called norm-bounded uncertainty model [BGFB94].

$$B(k) = B_0 + B_p \Delta(k) C_p, \quad \Delta(k) \in \mathbf{\Delta} \quad (3a)$$

$$\mathbf{\Delta} = \{\Delta : \|\Delta\| \leq 1\} \quad (3b)$$

Notice that a polytopic uncertainty can be approximated by a model of this type, see the appendix.

The goal in this paper is to solve a minimax MPC problem with quadratic performance measure (Q and R assumed positive definite)

$$\min_{u(\cdot|k)} \max_{\Delta^N} \sum_{j=0}^{N-1} \|x(k+j+1|k)\|_Q^2 + \|u(k+j|k)\|_R^2 \quad (4)$$

In the expression above, we introduced the total uncertainty along the future trajectory

$$\Delta^N = [\Delta(k), \dots, \Delta(k+N-1)] \in \mathbf{\Delta}^N = \mathbf{\Delta} \times \dots \times \mathbf{\Delta} \quad (5)$$

The paper is organized as follows. We begin in Section 3 with a review of some approaches to minimax MPC that have been proposed earlier in the literature. Section 4 introduces some central mathematical concepts. The main results are presented in Section 5 and we finally conclude the paper with a simple example.

3 Review of Available Results

The fundamental property that is exploited in minimax MPC for systems with an uncertain input gain is that with a convex uncertainty, the maximum of a convex performance measure will occur at the border of the uncertainty model [Ber99]. With a polytopic model of $B(k)$, it is thus possible to show that the maximum is found at a vertex of the uncertainty model.

Work on minimax MPC can be traced back to [CM87]. The uncertainty model was an uncertain FIR model,

$$y(k+1) = \sum_{i=0}^n g_i u(k-i) \quad (6)$$

where each impulse coefficient is subjected to a polytopic (time-invariant) uncertainty. By straightforward manipulations, this can be converted to a system with a polytopic uncertainty in the $B(k)$ matrix. The performance measure was chosen as the largest deviation (over a finite horizon) of the output $y(k+j|k)$ from some reference $r(k+j|k)$. Loosely speaking, this yields the problem (with Δ meaning the polytopic uncertainty)

$$\min_{u(\cdot|k)} \max_{\Delta} \max_j \|y(k+j|k) - r(k+j|k)\|_\infty \quad (7)$$

It was shown that this can be written as a linear programming (LP) problem. Unfortunately, the optimization problem had exponential complexity in the number of uncertain variables.

The complexity was improved in [AP91] where an equivalent LP problem with polynomial complexity was derived. It was also noted that the formulation with $\|\cdot\|_\infty$ easily could be extended to $\|\cdot\|_1$. Furthermore, the approach was extended to time-varying uncertainties.

Similar work on minimization of the worst-case deviation along a predicted trajectory, given a polytopic model of $B(k)$, can be found in, e.g., [Zhe95] and [OAFD00].

A quadratic minimax performance measure, such as (4) which we will address in this work, has not been studied to the same extent, at least not in the sense of efficient formulations.

Since the quadratic performance measure is convex, a polytopic uncertainty in the $B(k)$ matrix can be taken care of by just enumerating all the vertices of the uncertainty model along the future trajectory and solve a quadratic program for every possible combination. However, this has to be considered a non-constructive result since this will lead to problems with exponential complexity. If there are q uncertain parameters in the $B(k)$ matrix, there will be 2^q vertices of the uncertainty realization along the trajectory. Schemes based on straightforward enumeration can be found in, e.g., [LC97] and [CGM99].

An important approach to minimax MPC was introduced in [KBM94]. Basically, the performance measure used in the optimization is

$$x^T(k)P(k)x(k) \quad (8)$$

Of course, the matrix $P(k)$ is a part of the optimization problem and has to satisfy a number of constraints. In principle, the function $x^T(k)P(k)x(k)$ is an upper bound of the worst-case infinite horizon quadratic performance measure when the control sequence is parameterized as a linear feedback, a feedback which also is a part of the optimization problem. The obtained optimization problem can be solved in polynomial time with semidefinite programming (explained in the next section). The drawback with this approach is the possible conservativeness, since the feasible control sequence is limited to one generated by a linear feedback.

Stability of various minimax schemes has been studied in the literature. In [ZM93], it was shown that the original formulation in [CM87] could render the system unstable, and a remedy for this problem was introduced. Further work along these lines can be found in [Zhe95]. Various ideas to guarantee stability can also be found in [LC97]. A problem with the approaches in [Zhe95] and [LC97] is that they only hold for stable systems. Minimax formulations with guaranteed stability for unstable systems can be found in [KBM94] and, using the same basic principle, [CGM99]. A general result on minimax MPC with guaranteed stability can be found in [MRRS00].

4 Mathematical Preliminaries

The results in this paper are based on linear matrix inequalities, LMIs.

Definition 1 (LMI, [BGFB94]) *An LMI is an inequality, in the free scalar variables x_i , that for some fixed symmetric matrices F_i can be written*

$$F(x) = F_0 + x_1 F_1 + x_2 F_2 + \dots + x_n F_n \succeq 0$$

LMIs are used in semidefinite programming.

Definition 2 (SDP, [BGFB94]) *An SDP (semidefinite program), is an optimization problem that can be written*

$$\begin{aligned} \min_x \quad & c^T x \\ \text{subject to} \quad & F(x) \succeq 0 \end{aligned}$$

An SDP is a convex optimization problem that can be solved with high efficiency using solvers based on interior-point methods. The implementations needed for the examples in this paper have been solved with the freely available solver [VB].

The following lemma will be used repeatedly in this paper.

Lemma 1 (Schur complement, [Zha99]) *If $W \succ 0$, then for any $X \succeq 0$*

$$X - ZW^{-1}Z^T \succeq 0 \Leftrightarrow \begin{bmatrix} X & Z \\ Z^T & W \end{bmatrix} \succeq 0$$

The importance of this lemma is that it allows us to rewrite certain nonlinear matrix inequalities into linear matrix inequalities (LMIs). The lemma is a slight variation of the standard Schur complement which involves strict inequalities.

5 Main result

In this section, we show how to transform our original minimax problem to a semidefinite programming problem. The calculations will be done in a vector formulation, so we begin by defining the stacked state predictions and the future control sequence that we are trying to find

$$X = \begin{bmatrix} x(k+1|k) \\ x(k+2|k) \\ \vdots \\ x(k+N|k) \end{bmatrix}, \quad U = \begin{bmatrix} u(k|k) \\ u(k+1|k) \\ \vdots \\ u(k+N-1|k) \end{bmatrix} \quad (9)$$

The linear system and the uncertainty model $B(k) = B_0 + B_p \Delta(k) C_p$ allows us to write (for short notation $\Delta_j = \Delta(k+j)$)

$$X = Hx(k) + SU + \sum_{j=0}^{N-1} V_j \Delta_j W_j U, \quad \Delta_j \in \Delta \quad (10)$$

where

$$V_0 = \begin{bmatrix} B_p \\ AB_p \\ \vdots \\ A^{N-1}B_p \end{bmatrix}, \quad V_1 = \begin{bmatrix} 0 \\ B_p \\ \vdots \\ A^{N-2}B_p \end{bmatrix}, \quad \dots \quad (11)$$

$$W_0 = [C_p \quad 0 \quad 0 \quad \dots \quad 0], \quad W_1 = [0 \quad C_p \quad 0 \quad \dots \quad 0], \quad \dots \quad (12)$$

and H and S are defined in the standard way to account for the nominal part

$$H = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^N \end{bmatrix}, \quad S = \begin{bmatrix} B_0 & 0 & \dots & 0 \\ AB_0 & B_0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B_0 & A^{N-2}B_0 & \dots & B_0 \end{bmatrix} \quad (13)$$

For notational convinience, we redefine the weight matrices $Q := \text{diag}(Q, \dots, Q)$ and $R := \text{diag}(R, \dots, R)$. This makes it possible to write the minimax problem as

$$\begin{aligned} \min_{t, U} \quad & t \\ \text{subject to} \quad & \max_{\Delta^N} X^T Q X + U^T R U \leq t \end{aligned} \quad (14)$$

For the sake of a more compact notation, we define

$$\vartheta_i = \sum_{j=i}^{N-1} V_j \Delta_j W_j U \quad (15)$$

With this definition and Equation (10), our constraint in the optimization problem (14) can be rewritten with a Schur complement

$$\begin{bmatrix} t & x^T(k)H^T + U^T S^T + \vartheta_0^T & U^T \\ Hx(k) + SU + \vartheta_0 & Q^{-1} & 0 \\ U & 0 & R^{-1} \end{bmatrix} \succeq 0 \quad (16)$$

At this point, we would like to eliminate the uncertainties, and our tool to do this is the following theorem [GL97]

Theorem 1 (Robust LMI) *Robust satisfaction of the uncertain LMI*

$$F + L\Delta(I - D\Delta)^{-1}R + R^T(I - \Delta^T D^T)^{-1}\Delta^T L^T \succeq 0 \quad \forall \Delta \in \mathbf{\Delta}$$

is equivalent to the LMI

$$\begin{aligned} \begin{bmatrix} F & L \\ L^T & 0 \end{bmatrix} & \succeq \begin{bmatrix} R & D \\ 0 & I \end{bmatrix}^T \begin{bmatrix} \tau I & 0 \\ 0 & -\tau I \end{bmatrix} \begin{bmatrix} R & D \\ 0 & I \end{bmatrix} \\ \tau & \geq 0 \end{aligned}$$

We write our constraint in a form suitable for the above theorem by pulling out one uncertainty $\vartheta_0 = \vartheta_1 + V_0 \Delta_0 W_0 U$

$$\begin{bmatrix} t & x^T(k)H^T + U^T S^T + \vartheta_1^T & U^T \\ Hx(k) + SU + \vartheta_1 & Q^{-1} & 0 \\ U & 0 & R^{-1} \end{bmatrix} + \begin{bmatrix} U^T W_0^T \\ 0 \\ 0 \end{bmatrix} \Delta_0^T \begin{bmatrix} 0 & V^T & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ V \\ 0 \end{bmatrix} \Delta_0 \begin{bmatrix} W_0 U & 0 & 0 \end{bmatrix} \succeq 0 \quad (17)$$

Clearly, this is a special case of the structure in the theorem and we obtain the LMI

$$\begin{bmatrix} t & x^T(k)H^T + U^T S^T + \vartheta_1^T & U^T & U^T W_0^T \\ Hx(k) + SU + \vartheta_1 & Q^{-1} & 0 & 0 \\ U & 0 & R^{-1} & 0 \\ W_0 U & 0 & 0 & 0 \end{bmatrix} \succeq \begin{bmatrix} 0 & 0 \\ V_0 & 0 \\ 0 & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \tau_0 I & 0 \\ 0 & -\tau_0 I \end{bmatrix} \begin{bmatrix} 0 & V_0^T & 0 & 0 \\ 0 & 0 & 0 & I \end{bmatrix} \quad (18)$$

Simplification yields

$$\begin{bmatrix} t & x^T(k)H^T + U^T S^T + \vartheta_1^T & U^T & U^T W_0^T \\ Hx(k) + SU + \vartheta_1 & Q^{-1} - \tau_0 V_0 V_0^T & 0 & 0 \\ U & 0 & R^{-1} & 0 \\ W_0 U & 0 & 0 & \tau_0 I \end{bmatrix} \succeq 0 \quad (19)$$

The LMI above is still uncertain, due to the remaining term ϑ_1 . However, the structure is the same as the original LMI, so we can apply Theorem recursively until all uncertainties have been eliminated. The result will be a large LMI

$$\begin{bmatrix} t & x^T(k)H^T + U^T S^T & U^T & U^T W_0^T & U^T W_1^T & \dots \\ Hx(k) + SU & Q^{-1} - \sum_{j=0}^{N-1} \tau_j V_j V_j^T & 0 & 0 & 0 & 0 \\ U & 0 & R^{-1} & 0 & 0 & 0 \\ W_0 U & 0 & 0 & \tau_0 I & 0 & 0 \\ W_1 U & 0 & 0 & 0 & \tau_1 I & 0 \\ \vdots & 0 & 0 & 0 & 0 & \ddots \end{bmatrix} \succeq 0 \quad (20)$$

Working with such a large LMI might be inconvenient, but a Schur complement can be used to write it as a system of smaller LMIs, and we obtain our optimization problem

$$\begin{aligned} \min_{t, \tau, U} \quad & t_x + t_u + \sum_{j=0}^{N-1} t_j \\ \text{subject to} \quad & \begin{bmatrix} t_x & x^T(k)H^T + U^T S^T \\ Hx(k) + SU & Q^{-1} - \sum_{j=0}^{N-1} \tau_j V_j V_j^T \end{bmatrix} \succeq 0 \\ & \begin{bmatrix} t_u & U^T \\ U & R^{-1} \end{bmatrix} \succeq 0 \\ & \begin{bmatrix} t_j & U^T W_j^T \\ W_j U & \tau_j I \end{bmatrix} \succeq 0 \\ & \tau_j \geq 0 \end{aligned} \quad (21)$$

5.1 Connection to nominal MPC

One of the benefits with the proposed framework is that the obtained LMI can be analyzed to some extent. If we define the nominal state predictions $X_{nom} = Hx(k) + SU$, a Schur complement on the LMI (20) yields the equivalent constraint

$$X_{nom}^T (Q^{-1} - \sum_{j=0}^{N-1} \tau_j V_j V_j^T)^{-1} X_{nom} + U^T (R + \sum_{j=0}^{N-1} \frac{1}{\tau_j} W_j^T W_j) U \leq t \quad (22)$$

Recall that when we solve a nominal MPC problem, we have the constraint

$$X_{nom}^T Q X_{nom} + U^T R U \leq t \quad (23)$$

Hence, the difference is, to begin with, the additional $\sum_{j=0}^{N-1} \frac{1}{\tau_j} W_j^T W_j$ on the control weight. By recalling the definition of the matrices W_j , we see that the matrices $W_j^T W_j$ are blockdiagonal with zeros in the diagonal blocks except at the j th block. Hence, the extra weight on $u(k+j|k)$ will be proportional to τ_j^{-1} . The modified state weight is a bit less intuitive to analyse. However, if we assume τ is small, which simulations indicate, we can expand the expression using the matrix inversion lemma and then neglect higher order terms

$$(Q^{-1} - \sum_{j=0}^{N-1} \tau_j V_j V_j^T)^{-1} \approx Q + \sum_{j=0}^{N-1} \tau_j Q V_j V_j^T Q \quad (24)$$

We see that the state weight also will be increased, i.e. the robustification is not done by only increasing the control weight, and the adjustment of the state weight depends on both the original state weight and the uncertainty model.

6 Extensions

For the proposed framework to be interesting, it is important that standard extensions in nominal MPC can be applied also to our minimax controller. Indeed, this is the case as we will show here.

6.1 Control constraints

To begin with, we note that any standard convex constraint on U , such as amplitude or rate constraints, can be incorporated without any problems, since these constraints are unrelated to the uncertainty.

6.2 Linear state constraints

A typical situation in MPC is constraints on states and outputs. Also these can be dealt with in a robust manner. Let us assume that we can write the constraints as the element-wise constraint $MX \leq 1$. Inserting the definition of X yields

$$M(Hx(k) + SU + \sum_{j=0}^{N-1} V_j \Delta W_j U) \leq 1 \quad (25)$$

We define unit vectors e_i to pick out each single row

$$e_i^T M(Hx(k) + SU) + \sum_{j=0}^{N-1} V_j \Delta_j W_j U \leq 1 \quad (26)$$

Worst-case disturbances are found with the following lemma

Lemma 2

$$\max_{\|\Delta\| \leq 1} x^T \Delta y = \|x\| \|y\| \quad (27)$$

Proof: Follows from Schwarz inequality $(a^T b)^2 \leq \|a\|^2 \|b\|^2$. Equality when a and b are parallel. For us, this means that the equality holds when Δ is chosen so that x and Δy are parallel.

If we apply this to our uncertain predictions, we obtain the worst-case constraint

$$e_i^T M(Hx(k) + SU) + \sum_{j=0}^{N-1} \|e_i^T M V_j\| \|W_j U\| \leq 1 \quad (28)$$

We introduce the N bounds γ_j (so called second order cone constraints)

$$\|W_j U\| \leq \gamma_j \quad (29)$$

or equivalently

$$\begin{bmatrix} \gamma_j & U^T W_j^T \\ W_j U & \gamma_j I \end{bmatrix} \succeq 0 \quad (30)$$

With these bounds, we obtain the linear constraints

$$e_i^T M(Hx(k) + SU) + \sum_{j=0}^{N-1} \|e_i^T M V_j\| \gamma_j \leq 1 \quad (31)$$

Hence, the original linear constraints are taken care of by introducing N second order cone constraints and N new variables. Notice that despite how many constraints there are, the number of second order cone constraints and new variables will always be N .

6.3 Other uncertainty models

It is easy to realize that the approach straightforwardly extends to models of the type

$$B(k) = B_0 + \sum_{i=1}^q B_i \Delta_i(k) C_i \quad (32)$$

This means that some polytopic (parametric uncertainty) models can be used. For example,

$$B(k) = \begin{bmatrix} 1 + \Delta_1(k) & 2 + \Delta_1(k) + \Delta_2(k) \\ 3 & 4 + \Delta_2(k) \end{bmatrix}, \quad |\Delta_i(k)| \leq 1 \quad (33)$$

can be written as

$$B(k) = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \Delta_1(k) \begin{bmatrix} 1 & 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \Delta_2(k) \begin{bmatrix} 0 & 1 \end{bmatrix} \quad (34)$$

Unfortunately, general polytopic models cannot be modeled (using exactly the framework described here), a simple counter-example is the following model

$$B(k) = \begin{bmatrix} 1 + \Delta_1(k) & 2 + \Delta_1(k) + \Delta_2(k) \\ 3 + \Delta_2(k) & 4 + \Delta_2(k) \end{bmatrix}, \quad |\Delta_i(k)| \leq 1 \quad (35)$$

Basically, the parametric uncertainties must enter via rank 1 matrices.

6.4 State estimation and disturbances

The results in this work can be incorporated into the framework for minimax MPC for systems with state estimation errors and bounded external disturbances developed in [Löf01].

6.5 Contraction constraints

Contraction constraints is a simple way to guarantee stability for open-loop stable systems. The idea is to chose a quadratic Lyapunov function $x^T(k)Px(k)$, and force this to decay in every sample.

By assuming that the system is asymptotically stable, there exist a positive definite matrix P for any positive definite matrix S such that

$$A^T P A - P \preceq -S \quad (36)$$

The matrix P is used in the contraction constraint

$$\max_{\Delta(k)} x^T(k+1|k)Px(k+1|k) - x^T(k)Px(k) < 0 \quad (37)$$

The underlying idea is that this constraint always is feasible, since $u(k) = 0$ satisfies this, according to the definition of P . By applying a Schur complement, we see that the contraction constraint can be written as

$$\begin{bmatrix} x^T(k)Px(k) & x^T(k)A^T + u^T(k)B_0^T + C_p^T \Delta(k)B_p^T \\ Ax(k) + B_0 u(k) + B_p \Delta(k)C_p u(k) & P^{-1} \end{bmatrix} \succeq 0$$

Clearly, robust satisfaction of this LMI can be solved with Theorem 1 in the same way as we did for the quadratic performance cost.

The problem with a contraction constraint is that it only be used on stable systems since it assumes existence of a positive definite matrix P satisfying Equation (36). Such a matrix does not exist when the system is unstable. Another problem is that the constraint limits the degree of freedom in the optimization. This is related to the problem how S (i.e. P) should be chosen to obtain good performance.

6.6 Terminal state constraints

In the design of nominal MPC with guaranteed stability, many approaches use terminal state weights and constraints [MRRS00, L f01]. Most often, an ellipsoidal terminal state constraint $x^T(k + N|k)Sx(k + N|k) \leq 1$ and a quadratic terminal state weight $x^T(k + N|k)Px(k + N|k)$ is used.

The same idea is used, more or less explicitly, in most minimax MPC algorithms with guaranteed stability [KBM94, LC97, CGM99, BvdBV00].

A quadratic terminal state weight is easily incorporated into our framework by defining $Q := \text{diag}(Q, \dots, Q, P)$. Taking care of the terminal state constraint is done in the same way as for the contraction constraint, i.e., we apply a Schur complement to the terminal state constraint.

$$\begin{bmatrix} 1 & x^T(k + N|k) \\ x(k + N|k) & S^{-1} \end{bmatrix} \succeq 0$$

and use Theorem 1 to guarantee robust satisfaction. Finding suitable matrices P and S can be done by specializing the theory in [MRRS00] to our uncertainty model. The details are omitted for brevity.

7 Example

We study a sampled double integrator. In order to design a robust MPC controller, we create an uncertainty model which basically models an uncertain gain.

$$A = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad (38a)$$

$$B \in \mathbf{Co} \left(\begin{bmatrix} 1.50 \\ 0.55 \end{bmatrix}, \begin{bmatrix} 0.50 \\ 0.55 \end{bmatrix}, \begin{bmatrix} 1.50 \\ 0.45 \end{bmatrix}, \begin{bmatrix} 0.50 \\ 0.45 \end{bmatrix} \right) \quad (38b)$$

Our goal is to control the output $y(k) = x_2(k)$, under the control constraint $|u(k)| \leq 1$. A natural tuning is thus to only put weight on $x_2(k)$ in the performance measure. The chosen tuning variables were

$$Q = \begin{bmatrix} 0.01 & 0 \\ 0 & 1 \end{bmatrix}, \quad R = 0.01, \quad N = 5 \quad (39)$$

The polytopic model on $B(k)$ has to be converted to a norm-bounded model. This is done using the approach described in the appendix and we obtain

$$B_0 = \begin{bmatrix} 1 \\ 0.50 \end{bmatrix}, \quad B_p = \begin{bmatrix} 0.52 & 0 \\ 0 & 0.16 \end{bmatrix}, \quad C_p = \begin{bmatrix} 0.707 \\ 0.707 \end{bmatrix} \quad (40)$$

As a first experiment, we test a bad uncertainty realization (found by trial). The following uncertainty realization was used

$$B(k) = \begin{cases} \begin{bmatrix} 0.5 \\ 0.45 \end{bmatrix} & u(k) > 0 \\ \begin{bmatrix} 1.5 \\ 0.55 \end{bmatrix} & u(k) \leq 0 \end{cases} \quad (41)$$

The closed loop performance from the initial condition $x(0) = [0 \ 5]^T$ with this tuning and uncertainty realization can be seen in Figure 1. The nominal MPC controller has very poor performance, while the robust minimax MPC controller gives pretty good performance. Of course, the tuning is doomed to

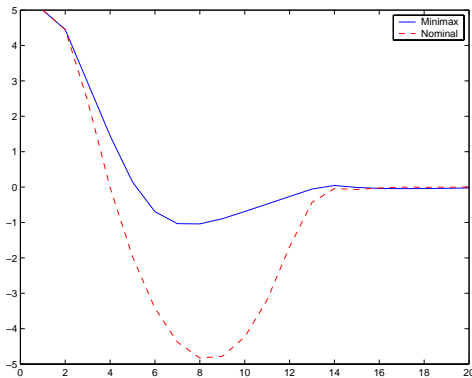


Figure 1: Aggressive tuning $R = 0.01$ and bad uncertainty.

give a nominal controller with poor robustness. A natural solution is to detune the controller, so we chose $R = 1$ instead and perform the same experiment. Surprisingly, this did actually not improve performance of the nominal controller

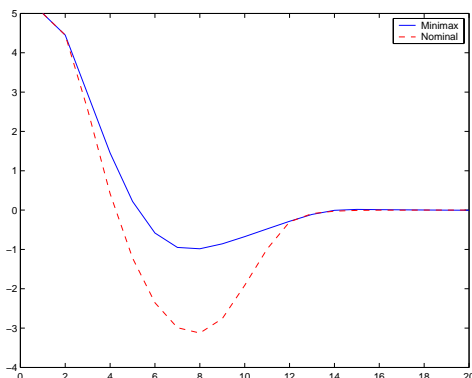


Figure 2: Detuned controllers $R = 1$ and bad uncertainty.

that much. The minimax controller gives pretty much the same response as before, see Figure 2.

By tuning the nominal controller carefully, it is possible to obtain better performance. However, the idea with robust control is that this should not be necessary. The tuning variables should reflect the actual performance criteria, and the robustness should be built-in. Given a new uncertainty model, it should not be necessary to retune the controller.

So what is the price? Conservativeness is the main problem with robust controllers, i.e., the performance in the “non-worst-case” situation can deteriorate. However, for this example, this is actually not the case. In Figure 3 we simulate the system with a random uncertainty using both the nominal and the minimax

controller. We see that the robust controller gives a slightly sluggish behaviour

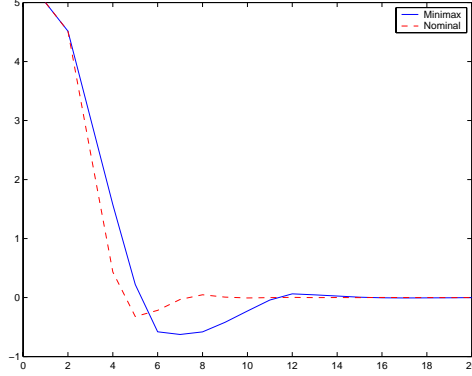


Figure 3: Aggressive tuning $R = 0.01$ and random uncertainty.

compared to the nominal controller. However, the robust controller does not seem to be overly conservative.

8 Discussion and Conclusions

There is one major drawback with the proposed minimax controller; feedback predictions cannot be applied. Feedback predictions is a standard trick in robust MPC in order to reduce the conservativeness. It is based on parameterizing the control sequence as

$$u(k+j) = -Lx(k+j) + v(k+j) \quad (42)$$

The reason that this cannot be used is that the predictions will be

$$x(k+j+1) = (A - B(k+j)L)x(k+j) + B(k+j)v(k+j) \quad (43)$$

Hence, the uncertainty in B will give uncertainty in the A matrices used for the predictions. Of course, this is not a problem only for our algorithm, this is the case for all approaches that are based on the linear relation between B and the prediction X .

Minimax MPC is not applicable to all systems. The example which we studied is of course chosen to point out the possible benefits of a minimax controller. However, the purpose of this paper has not been to advocate the use of minimax controllers, but to show that the problem at least can be solved efficiently and incorporated into the existing MPC framework.

An interesting question that needs further investigation is whether the connections to nominal MPC that we have found can be exploited in any sense, i.e., guidance in off-line tuning of Q and R .

References

- [AP91] J.C. Allwright and G. Papavasiliou. A reduced size L.P. formulation for robust M.P.C. using impulse-response models. In *Proceedings of the 30th Conference on Decision & Control*, pages 1939–1940, 1991.
- [Ber99] D.P. Bertsekas. *Nonlinear Programming*. Athena Scientific, second edition, 1999.
- [BGFB94] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. SIAM Studies in Applied Mathematics. SIAM, Philadelphia, Pennsylvania, 1994.
- [BvdBV00] H.H.J. Bloemen, T.J.J. van den Boom, and H.B. Verbruggen. Model-based predictive control for hammerstein systems. In *Proceedings of the 39th Conference on Decision & Control*, 2000.
- [CGM99] A. Casavola, M. Giannelli, and E. Mosca. Robust predictive control of input-saturated uncertain systems. In *Proceedings of the 38th Conference on Decision & Control*, pages 2768–2773, 1999.
- [CM87] P.J. Campo and M. Morari. Robust model predictive control. In *Proceedings of the American Control Conference*, pages 1021–1025, 1987.
- [GL97] L. El Ghaoui and H. Lebrete. Robust solutions to least-squares problems with uncertain data. *SIAM Journal on Matrix Analysis and Applications*, 18(4):1035–1064, 1997.
- [KBM94] M.V. Kothare, V. Balakrishnan, and M. Morari. Robust constrained model predictive control using linear matrix inequilities. In *Proceedings of the American Control Conference*, pages 440–444, 1994.
- [LC97] J.H. Lee and B.L. Cooley. Stable min-max control for state-space systems with bounded input matrix. In *Proceedings of the American Control Conference*, pages 2945–2949, 1997.
- [Löf01] J. Löfberg. Linear model predictive control: Stability and robustness. Licenciate thesis LIU-TEK-LIC-2001:03, Departement of Electrical Engineering, Linköpings universitet, Sweden, 2001.
- [MRRS00] D.Q. Mayne, J.B. Rawlings, C.V. Rao, and P.O.M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36:789–814, 2000.
- [OAFD00] G.H.C. Oliviera, W.C. Amaral, G. Favier, and G.A. Dumont. Constrained robust predictive controller for uncertain processes modeled by orthonormal series functions. *Automatica*, 36(4):563–572, 2000.
- [SR00] J. Schuurmans and J.A. Rossiter. Robust predictive control using tight sets of predicted sets. *Control Theory and Applications*, 147(1):13–18, 2000.

- [VB] L. Vandenberghe and S. Boyd. *SP - Software for Semidefinite Programming*. Information Systems Laboratory, Electrical Engineering Departement, Stanford University, version 1.0 edition.
- [Zha99] F. Zhang. *Matrix Theory: Basic results and techniques*. Springer-Verlag, New York, 1999.
- [Zhe95] A. Zheng. *Robust Control of Systems Subject to Constraints*. PhD thesis, California Institute of Technonlogy, Pasadena, California, 1995.
- [ZM93] A. Zheng and M. Morari. Robust stability of constrained model predictive control. In *Proceedings of the 4th IEEE Conference on Control Applications*, pages 379–383, 1993.

A Approximation of polytopic model

A polytopic models of $B(k)$ can be approximated by a norm-bounded model by using the following procedure [BGFB94]. In principle, we have an uncertain matrix B ,

$$B \in \mathbf{Co}(B_1, B_2, \dots, B_r) \quad (44)$$

which for a fixed x gives a set of possible y -values in the mapping

$$y = Bx \quad (45)$$

This set is a polytope defined by the vertices B_i (and the argument x of course). We want this reachable set to be contained in the reachable set when a norm-bounded matrix is used instead

$$y = (B_0 + B_p \Delta C_p)x \quad (46)$$

Define $\eta = \Delta C_p x$

$$y = B_0 x + B_p \eta \quad (47)$$

The polytopic set is contained if each of the vertices $B_i x$ is in the set defined by the norm-bounded uncertainty (which is convex). In other words, for each uncertainty η , it should be possible to write

$$B_p \eta = (B_j - B_0)x, \quad \eta^T \eta \leq x^T C_p^T C_p x \quad (48)$$

If we assume B_p to be invertible, this is equivalent to

$$(B_j - B_0)^T B_p^{-T} B_p^{-1} (B_j - B_0) \leq C_p^T C_p \quad (49)$$

Define $Z = B_p B_p^T$ och $Y = C_p^T C_p$ and a Schur complement yields

$$\begin{bmatrix} Y & (B_j - B_0)^T \\ (B_j - B_0) & Z \end{bmatrix} \succeq 0 \quad (50)$$

At this point, any “size-measure” ($\text{trace}(\cdot)$, $\|\cdot\|$, ...) on Z and Y can be minimized. Notice that the procedure does not uniquely define B_p och C_p . As an example, if C_p is a column-vector, Y will be a scalar and there will be infinitely many C_p such that $C_p^T C_p = Y$.