Katholieke Universiteit Leuven

Departement Elektrotechniek

ESAT-SISTA/TR 04-136

A Simple Algorithm For Robust MPC [1]

B. Pluymers[2] , J.A. Rossiter[3] , J. Suykens[2] and B. De Moor[2]

March 2005

Proceedings of the IFAC World Congress 2005, Prague, Czech Republic.

[1]This report is available by anonymous ftp from *ftp.esat.kuleuven.ac.be* in the directory *pub/sista/pluymers/reports/ifac05a_rmpc.pdf*

[2]K.U.Leuven, Dept. of Electrical Engineering (ESAT), Research group SCD-SISTA, Kasteelpark 10, 3001 Leuven, Belgium, Tel. 32/16/32 10 35, Fax 32/16/32 19 70, WWW: *http://www.esat.kuleuven.ac.be/scd.* E-mail: {*bert.pluymers,johan.suykens,bart.demoor*}*@esat.kuleuven.ac.be.*

Research supported by Research Council KUL: GOA-Mefisto 666, GOA-Ambiorics, several PhD/postdoc & fellow grants; Flemish Government: FWO: PhD/postdoc grants, projects, G.0240.99 (multilinear algebra), G.0407.02 (support vector machines), G.0197.02 (power islands), G.0141.03 (Identification and cryptography), G.0491.03 (control for intensive care glycemia), G.0120.03 (QIT), G.0452.04 (QC), G.0499.04 (robust SVM), research communities (ICCoS, ANMMM, MLDM); AWI: Bil. Int. Collaboration Hungary/ Poland; IWT: PhD Grants, GBOU (McKnow) Belgian Federal Government: Belgian Federal Science Policy Office: IUAP V-22 (Dynamical Systems and Control: Computation, Identification and Modelling, 2002-2006), PODO-II (CP/01/40: TMS and Sustainibility); EU: FP5-Quprodis; ERNSI; Eureka 2063-IMPACT; Eureka 2419-FliTE; Contract Research/agreements: ISMC/IPCOS, Data4s, TML, Elia, LMS, IPCOS, Mastercard; Bert Pluymers is a research assistant with the I.W.T. (Flemish Institute for Scientific and Technological Research in Industry). Dr. Johan Suykens is an associate professor at the Katholieke Universiteit Leuven. Dr. Bart De Moor is a full professor at the Katholieke Universiteit Leuven, Belgium.

[3]University of Sheffield, Department of Automatic Control and Systems Engineering, Mappin Street, Sheffield S1 3JD, United Kingdom, E-mail: *j.a.rossiter@sheffield.ac.uk*

Abstract

The majority of recent works on robust MPC either require very burdensome online computations or are restricted to relatively small feasible regions. This paper builds on a recent work which demonstrated that one could compute the maximal admissible set for a linear parameter varying system and shows how this set can be used as the terminal region and hence allows the definition of an MPC algorithm requiring only quadratic programming, but with a maximal region of attraction and guaranteed convergence for the robust case.

A SIMPLE ALGORITHM FOR ROBUST MPC

B. Pluymers * [1] **J.A. Rossiter** ** **J.A.K. Suykens** *
B. De Moor * [2]

* *Research Group SCD-SISTA, Department Of Electrotechnical Engineering (ESAT), Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, 3001 Leuven, Belgium*
{bert.pluymers,johan.suykens,bart.demoor}@esat.kuleuven.ac.be
** *Department of Automatic Control and Systems Engineering, Mappin Street, University of Sheffield, S1 3JD, UK*
j.a.rossiter@sheffield.ac.uk

Abstract: The majority of recent works on robust MPC either require very burdensome online computations or are restricted to relatively small feasible regions. This paper builds on a recent work which demonstrated that one could compute the maximal admissible set for a linear parameter varying system and shows how this set can be used as the terminal region and hence allows the definition of an MPC algorithm requiring only quadratic programming, but with a maximal region of attraction and guaranteed convergence for the robust case.
Copyright© 2005 IFAC

Keywords: Robust MPC, maximal admissible sets, LPV systems

1. INTRODUCTION

Linear model predictive control (MPC) is, by now, well understood (Mayne *et al.*, 2000; Rossiter, 2003) and the remaining issues are more to do with implementation than the underlying algorithm, that is how can we speed up, simplify or other the online implementation. Robust MPC however still attracts a great deal of attention in the literature. Partially this is due to the large number of possible scenarios: for instance how to classify parameter uncertainty, or disturbance uncertainty. However of more significance is the computational complexity that arises from a desire to give guarantees of feasibility and convergence; this is evident even from the seminal paper (Kothare *et al.*, 1996). Some authors have sought to reduce the complexity of the online computations (Kouvaritakis *et al.*, 2000; Kothare *et al.*, 2003) by performing more offline analysis, however with a significant restriction to the volume of the feasible region, as one needs to fall back to ellipsoidal invariant regions; it is known, at least for the certain case, that the maximal volume feasible region is polyhedral in general. To the authors' knowledge no-one has yet proposed the computation and use of the maximal admissible set (MAS) (Gilbert *et al.*, 1991) for the uncertain case, perhaps because a simplistic extension of the original algorithm in (Gilbert *et al.*, 1991) is intractable in general.

This situation is now changing. For instance the work of (Kerrigan, 2000) and co-workers has demonstrated that one can compute invariant polyhedral sets for the case of linear parameter varying (LPV) systems with disturbances, although the focus there was very general. More recently (Pluymers *et al.*, 2005) it was shown that for the specific case of a fixed control law, which is usually assumed for the terminal region in MPC (Scokaert *et al.*, 1998), one can compute the MAS for an LPV system efficiently. The main contribution here, therefore, is show how these

[1] B. Pluymers is a research assistant with the I.W.T. Flanders.
[2] Research partially supported by Research Council KULeuven : GOA-Mefisto 666, FWO projects : G.0240.99, G.0197.02, G.0491.03, Belgian Federal Government : IUAP V-22 (2002-2006).

robust invariant sets can be used to formulate a robust MPC algorithm. Of particular note is that the proposed algorithm requires only a quadratic program (QP) online, and therefore is far simpler than the typical offerings in the literature.

It should be emphasised that the algorithm proposed here gives a guarantee of feasibility and convergence for the uncertain system and notably for the largest volume feasible region possible, assuming a given terminal control law. This we believe has not appeared in the literature before where the online algorithm makes use only of a QP optimisation. Moreover, it should be emphasised that the algorithm of this paper can cater for non-symmetric constraints which ellipsoidal (Kothare *et al.*, 1996) and low complexity polyhedral methods (Cannon *et al.*, 2002) cannot.

The paper is organised as follows. Section 2 gives some background to MPC and invariant sets. Section 3 presents the proposed algorithm and its properties and section 4 gives some numerical illustrations.

2. BACKGROUND

2.1 Notation, prediction, cost and control

This paper will assume discrete state space models and state feedbacks of the form

$$x_{k+1} = Ax_k + Bu_k; \quad u_k = -Kx_k \tag{1}$$

Performance, either predicted or actual, will be assessed by the cost

$$J = \sum_{k=0}^{\infty} x_k^T Q x_k + u_k^T R u_k \tag{2}$$

More generally, let the 'predicted' control law (Rossiter *et al.*, 1998; Scokaert *et al.*, 1998) be:

$$\begin{array}{ll} u_k = -Kx_k + c_k & k = 0, ..., n_c - 1 \\ u_k = -Kx_k & k \geq n_c \end{array} \tag{3}$$

where c_k are d.o.f. available for constraint handling. Substituting (1,3) into (2) one can show, for K the optimal ((Rossiter, 2003) chap. 6,7), that J takes the form

$$J = \sum_{j=0}^{i-1} c_j^T W c_j + p; \quad p = x^T V x \tag{4}$$

and $W = B^T \Sigma B + R,\ \Sigma - \Phi^T \Sigma \Phi = Q + K^T R K$; as p is not dependent on the d.o.f., it is convenient to omit it and rephrase the objective function as:

$$J = C^T W_D C; \quad C = [c_0^T, ..., c_{n_c-1}^T]^T \tag{5}$$

W_D =diag$(W, ..., W)$.

2.1.1. Constraints and MAS Assume that the process is subject to constraints:

$$\underline{u} \leq u_k \leq \overline{u}; \quad \underline{x} \leq x_k \leq \overline{x}, \qquad k = 1, \ldots, \infty \tag{6}$$

The MAS (Gilbert *et al.*, 1991) is defined as the region within which the state and control evolutions (in this case from the closed-loop system $x_{k+1} = [A - BK]x_k$) satisfies constraints (6). Let the MAS be defined as

$$S_0 = \{x : M_0 x \leq d_0\} \tag{7}$$

The maximal control admissible set (MCAS) is the region within which the d.o.f. C (from (3)) are sufficient to ensure constraint satisfaction of the predictions [1]. One can show (if S_0 has sufficient redundant rows) that the MCAS takes the form:

$$S_c = \{x : \exists C \text{ s.t. } M_0 x + N_0 C \leq d_0\} \tag{8}$$

2.1.2. MPC for the nominal case A typical paradigm for MPC minimises the cost of (5) subject to the transient and terminal constraints which are implicit in (8).

Algorithm 2.1. **Nominal MPC:** At each sampling instant, perform the optimisation:

$$\min_C \quad J = C^T W_D C \quad \text{s.t.} \quad M_0 x + N_0 C \leq d_0 \tag{9}$$

Use the first block element of C in control law (3).

Remark 2.1. This algorithm (Scokaert *et al.*, 1998) has a guarantee of recursive feasibility and convergence, for the nominal case. It can also be shown that for n_c large enough and K the optimal control law, the solution is the same as the optimum constrained control law.

2.2 MPC for the uncertain case

This paper extends MPC to deal with parameter uncertainty (extensions to deal with disturbances form ongoing work). In particular we will consider the case of LPV systems, that is

$$\begin{array}{c} x_{k+1} = A(k)x_k + B(k)u_k; \\ (A(k), B(k)) \in Co\{(A_1, B_1), ..., (A_m, B_m)\} \end{array} \tag{10}$$

Typical MPC algorithms which deal with this case suffer a number of limitations. MPC is based on predictions, but authors have generally assumed that one cannot form the entire class of predictions due to the combinatorial explosion in the required number of terms (Schuurmanns *et al.*, 2000). This limitation also meant that people were unsure how to formulate the MAS for this system and hence an MPC algorithm such as 2.1 could not be constructed.

As a consequence the vast majority of work has made use of low complexity invariant sets, in particular ellipsoidal sets (Kothare *et al.*, 1996) or very simple polyhedral sets (Cannon *et al.*, 2002). In this case one can form conditions which apply to the LPV system, but the restriction to low complexity invariant sets implies a corresponding restriction to the applicability of the associated algorithms, that is to the feasible region.

[1] Use C to satisfy constraints over the first n_c steps and to ensure that $x_{k+n_c} \in S_0$.

Some authors have tackled this limitation by augmenting the state dimension (Kouvaritakis *et al.*, 2000; Imsland *et al.*, 2004) and in effect allowing a time varying control strategy. However, the restriction to ellipsoidal sets remains in those works. Other examples of algorithms using time-varying terminal ellipsoidal regions are (Pluymers *et al.*, 2004; Park *et al.*, 1999; Kothare *et al.*, 2003), but also in these algorithms the need for SDP directly induces a significantly larger online computational burden than QP-based methods.

2.3 Polyhedral invariant sets for LPV systems

This paper proposes to make use of a recent development (Pluymers *et al.*, 2005) which demonstrates how, in some cases, one can indeed formulate the MAS for an LPV system. The key idea used is not dissimilar to the one-step sets popularised in (Kerrigan, 2000), that is to use backwards prediction rather than forwards prediction. This simple change eliminates the combinatorial explosion in the possible number of prediction terms and hence creates a tractable problem.

A brief summary of the key results is given next without details, for which the reader is referred to (Pluymers *et al.*, 2005). First define the closed-loop system matrices

$$\Phi_i = A_i - B_i K, \qquad i = 1, \ldots, m. \tag{11}$$

The MAS for the uncertain system (10), constraints (6) and control law $u = -Kx$ is:

$$S_u = \{x : M_u x \leq d_u\}. \tag{12}$$

By definition S_u is invariant so $x \in S_u \Rightarrow \Phi_i x \in S_u,\ i = 1, \ldots, m$. This can be shown to be equivalent with $S_u \subset S_u^-$, with the $^-$-operator defined as $S^- = \{x : \Phi_i x \in S, i = 1, \ldots, m\}$. The following algorithm starts with the initial set $S = \{x : \underline{x} \leq x \leq \overline{x}, \underline{u} \leq -Kx \leq \overline{u}\}$ and iteratively adds constraints from S^- until $S \subset S^-$. The resulting set is the MAS.

Algorithm 2.2. Robust Invariant Set

(1) Set $M_u := [I^{\mathrm{T}} \ -I^{\mathrm{T}} \ -K^{\mathrm{T}} \ K^{\mathrm{T}}]^{\mathrm{T}}, d_u = [\overline{x}^{\mathrm{T}} \ \underline{x}^{\mathrm{T}} \ \overline{u}^{\mathrm{T}} \ \underline{u}^{\mathrm{T}}]$ and $i := 1$.
(2) Select row i from (M_u, d_u) and check $\forall j$ whether $M_{u,i}\Phi_j x \leq d_{u,i}$ is redundant with respect to the constraint defined by (M_u, d_u). Add the non-redundant constraints to (M_u, d_u) by assigning $M_u := [M_u^{\mathrm{T}} \ (M_{u,i}\Phi_j)^{\mathrm{T}}]^{\mathrm{T}}$ and $d_u := [d_u^{\mathrm{T}} \ d_{u,i}^{\mathrm{T}}]^{\mathrm{T}}$ for all relevant j.
(3) Set $i := i + 1$. If i is strictly larger than the number of rows in (M_u, d_u) then terminate, otherwise continue with step 2).

The resulting set $S_u = \{x : M_u x \leq d_u\}$ is the MAS for the given system, constraints, and feedback controller. The algorithm is guaranteed to terminate in a finite number of iterations if the closed-loop system is quadratically stable :

$$\exists V > 0 \ \text{ s.t. } \ \Phi_i^{\mathrm{T}} V \Phi_i \leq V, \ \ \forall i \tag{13}$$

Remark 2.2. Constraints added in later iterations of the algorithm can render constraints added in earlier iterations redundant. Therefore it is advisable to check for redundant constraints regularly during the execution of the algorithm; this can decrease execution time considerably.

3. ROBUST MPC FOR LPV SYSTEMS

This section shows how to make use of the robust MAS (12) to develop an algorithm very similar to the conventional algorithm 2.1 but applicable to a LPV system. That is, one which minimises nominal predicted performance but subject to constraint satisfaction by the whole class of possible predictions during transients and moreover with the condition that all possible n_c-step ahead predictions are within the terminal set (12). Some lemmata and proofs demonstrate the properties of recursive feasibility and convergence.

3.1 Robust predictions and constraint handling

Define a nominal model $(A,\ B)$ and use this to form the nominal predictions over n_c steps. These predictions are used solely to give the expected value (5) of the cost function J which will be the target Lyapunov function for stability analysis. Nominal predictions used to formulate the MCAS in algorithm 2.1 are

$$x_{k+n} = \Phi^n x_k + \sum_{j=0}^{n-1} \Phi^{n-j-1} B c_j \tag{14}$$

For robust constraint handling, one needs to ensure that all possible predictions satisfy constraints and therefore one must formulate all possible predictions for the model (10). The n-step ahead prediction is given as

$$x_{k+n} = \prod_{i=0}^{n-1} \Phi(i) x_k + \sum_{j=0}^{n-1} \{ \prod_{l=0}^{n-j-2} \Phi(n-l) \} B c_j \tag{15}$$

where $\Phi(i) \in Co\{\Phi_1, ..., \Phi_m\}$. Clearly one must allow for all possible combinations of $\Phi(i)$ to capture all the predictions; this procedure is only tractable for small n, as the number of possible vertices for x_{k+n} grows at approximately m^n.

3.1.1. Transient constraints The constraint equations (6) must also be applied to all possible $x_{k+i}, i = 0, ..., n_c - 1$ (note these constitute only the transient constraints). Omitting the algebra as tedious, but straightfoward, these constraints are satisfied, for all possible predictions (15), if the following inequalities are satisfied:

$$M_t x_k + N_t C \leq d_t \tag{16}$$

3.1.2. Terminal constraints One requires all possible n_c-step ahead predictions to lie within the terminal region S_u of (12), therefore each possible prediction must satisfy $M_u x_{k+n_c} \leq d_u$. These constraints can be represented by:

$$M_v x_k + N_v C \leq d_v; \tag{17}$$

Hence combining constraints (16,17) the overall feasible region is given by:

$$\begin{aligned} S_r &= \{x : \exists C \text{ s.t. } M_r x_k + N_r C \leq d_r\}; \\ M_r &= \begin{bmatrix} M_t \\ M_v \end{bmatrix};\ N_r = \begin{bmatrix} N_t \\ N_v \end{bmatrix};\quad d_r = \begin{bmatrix} d_t \\ d_v \end{bmatrix} \end{aligned} \tag{18}$$

Remark 3.1. The reader may assume that the number of constraint equations implied here is rather large and this would lead to an intractable QP. However, many of these constraints would be redundant and could be removed offline. We are investigating equivalent procedures to that deployed in (Pluymers *et al.*, 2005) to compute the sets more efficiently but omit this due to lack of space.

3.2 The algorithm and its properties

This section now defines the QP optimisation defining the online algorithm and gives some discussion to its properties.

Algorithm 3.1. **Robust MPC**

At each sampling instant, minimise the performance index:

$$\min_C\ J = C^T W_D C \text{ s.t. } M_r x_k + N_r C \leq d_r \tag{19}$$

Use the first block element of C in control law (3).

Lemma 3.1. Algorithm 3.1 has a guarantee of recursive feasibility within the set S_r of (18).

Proof: Recursive feasibility requires that

$$x_k \in S_r \quad \Rightarrow \quad x_{k+1} \in S_r \tag{20}$$

Now the definition of (16,17) and hence (18) allowed for all possible predictions deploying n_c control perturbations c_i, so the one step ahead predictions implicit in this must lie in a region for which all possible predictions deploying $n_c - 1$ perturbations are feasible; this by necessity therefore must be a subset of S_r. □

Lemma 3.2. The cost J is continuous for all points outside the terminal region and inside S_r, it is piecewise quadratic with minimum on the boundary on S_u and therefore can be deployed as a potential lyapunov function.

Proof: By definition (17), the unconstrained control law is feasible and gives convergence to the origin, that is, the choice $C = 0$ (which implies $J = 0$) is feasible within S_u. As algorithm 3.1 is a QP, the piecewise quadratic nature of J outside S_u is well known (Bemporad *et al.*, 2002). □

Lemma 3.3. The cost J reduces each sample until the state enters S_u and hence is Lyapunov.

Proof: This proof follows a well accepted route of showing that the tail of the optimum from sampling instant k is a valid choice at sampling instant $k+1$; as this choice implies a reduction in J, the new optimum also gives a reduction in J. The cost function is specified in terms of the control perturbations as $J_k = \sum_{j=0}^{i-1} c_{j|k}^T W c_{j|k}$. A possible choice at the next sampling instant is $c_{j|k+1} = c_{j|k}$ and hence $J_{k+1} \leq J_k - c_{0|k}^T W c_{0|k}$. Hence J is a lyapunov function because one can only have that $J_{k+1} = J_k, \forall k$ if $c_k = 0,\ \forall k$ which implies that the state is already inside the terminal region and hence $J = 0$. Because J is Lyapunov, the perturbation terms c must converge to zero which implies the state has entered S_u. □

Theorem 3.1. Algorithm 3.1 has a guarantee of convergence for all initial points inside the feasible region S_f.

Proof: The proof needs three key threads; (i) a guarantee of recursive feasibility (established in lemma 3.1); (ii) a proof that once inside the terminal region S_u the state converges (implicit in (12)) and (iii) a proof that states outside the terminal region converge to the terminal region (established in lemma 3.3). □

4. NUMERICAL ILLUSTRATIONS

It is perhaps obvious how the proposed algorithm improves upon those algorithms already in the literature, however for completeness we present some numerical illustrations to demonstrate the efficacy. For simplicity we will illustrate with a single example whose model and constraints are:

$$A_1 = \begin{bmatrix} 1 & 0.1 \\ 0 & 1 \end{bmatrix}, \qquad B_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \tag{21}$$

$$A_2 = \begin{bmatrix} 1 & 0.2 \\ 0 & 1 \end{bmatrix}, \qquad B_2 = \begin{bmatrix} 0 \\ 1.5 \end{bmatrix}, \tag{22}$$

$$\overline{u} = -\underline{u} = 1, \quad \overline{x} = -\underline{x} = [10,\ 10]^{\mathrm{T}}. \tag{23}$$

The nominal model is chosen as $A = 0.5(A_1 + A_2), B = 0.5(B_1 + B_2)$. The feedback controller $K = [0.4558\ 0.3698]^{\mathrm{T}}$ is the LQ-optimal for $Q = \mathrm{diag}(1, 0.01), R = 3$.

4.1 Improved volume for terminal region

The common alternative to a polyhedral set is an ellipsoidal set. Hence here we overlay, in figure 1, the maximum volume invariant ellipsiodal set (dotted line) and the robust MAS (solid line). The enlargement of the target region is obvious.

For completeness it is also worth demonstrating the impact of non-symmetric constraints, e.g.:

$$\overline{u} = 1; \quad \underline{u} = -0.5; \quad \overline{x} = -\underline{x} = [10,\ 10]^T. \tag{24}$$

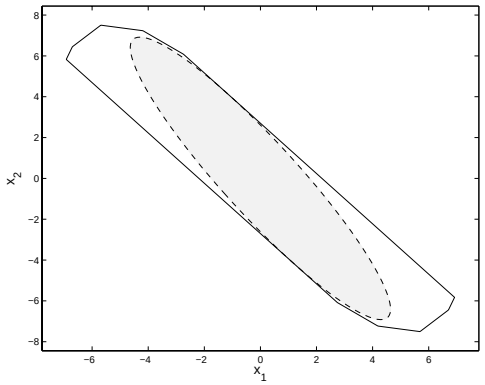


Fig. 1. Ellipsoidal and polyhedral terminal invariant sets with symmetric constraints (23).

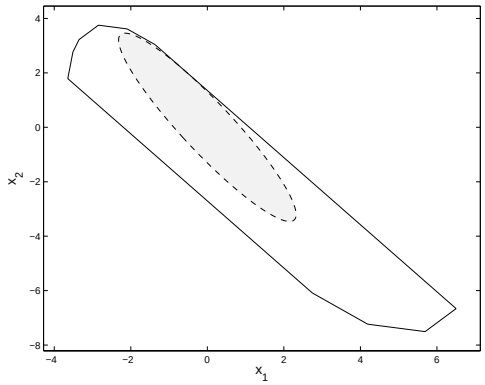


Fig. 2. Ellipsoidal and polyhedral terminal invariant sets with non-symmetric constraints (24).

We overlay, in figure 2, the maximum volume invariant ellipsiodal set and the robust MAS. The enlargement of the target region is unsurprisingly very significant in this case.

4.2 Improved region of attraction

The region of attraction once degrees of freedom have been added, is S_r as defined in (18). However, in the case of ellipsiodal target sets, the corresponding calculations are either hard to do because they do not involve LP or QP, but rather require SDP (Cannon *et al.*, 2001) (and hence would not form a fair comparison due to the excessive online computation implied) or use simplifications as in (Kouvaritakis *et al.*, 2000) but are therefore still restricted to ellipsoids, with the obvious limitations illustrated in figures 1 and 2. So here we simply demonstrate in figure 3 that the feasible region for algorithm 3.1, increases in volume as more d.o.f. are added.

4.3 Closed-loop performance and robustness

The motivation for this algorithm was to give a guarantee of stability and convergence for the uncertain case. To demonstrate how the proposed algorithm performs, we take a number of initial points near the boundary of the feasible region and allow the model to be arbitrarily time-varying within the allowable sets of parameters (see (10)). Some of the resulting trajectories are plotted in figure 4. They all converge to the origin!

4.4 Improved robustness

What perhaps is of equal interest is a comparison between the robust algorithm and the nominal algorithm 2.1. Both have equivalent behaviour within S_u, the terminal region, but this is not true outside S_u where different choices of C are implied by the differing constraints in the optimisations.

Case 1: One can conjecture a value x such that $x \in S_0$ but $x \notin S_u$. In this case algorithm 2.1 will suggest $C = 0$ whereas algorithm 3.1 would give $C \neq 0$ and so one could clearly demonstrate that J is Lyapunov for algorithm 3.1 but not so for algorithm 2.1 (due to the model uncertainty the state may subsequently move outside S_0). Without this Lyapunov property the proof of convergence is invalid. For instance take the initial state $[-4.0285\ 3.5259]^{\mathrm{T}}$ situated on the boundary of S_0, then it suffices only to show that there exists $(A(k), B(k))$ such that $\max_i(e_i^T[M_0x - d_0]) = 0$ implies that $\max_i(e_i^T[M_0\Phi(k)x - d_0]) > 0$. This is the case for both (A_1, B_1) as (A_2, B_2).

Case 2: An even worse scenario could be where algorithm 2.1 selects a move that leads to infeasibility; in some cases this can lead to unavoidable instability and at best an undefined control strategy. An illustration of this would be points near the boundary of S_r; for points outside this we know there must exist a set of parameter variations such that one may not be able to ensure convergence and constraint satisfaction. Hence it is sufficient to show, that the Algorithm 2.1 may select an optimum $\mathbf{C}$ such that $\max_i(e_i^T[Mx - d]) = 0$ implies that $\max_i(e_i^T[M\{\Phi(k)x + B(k)c\} - d]) > 0$. One example is the initial state $[6.9992\ -8.3825]$, lying on the boundary of S_r and leading to $c = 0.1602$ for Algorithm 2.1. This results for both $(A_1\ B_1)$ and $(A_2\ B_2)$ in a next state outside of S_r, leading to infeasibilities in future iterations for at least some possible uncertainty realisations.

4.5 Computational load

A key contribution of this paper is to develop an algorithm that makes use only of a QP optimiser. In fact algorithms 2.1 and 3.1, are identical apart from the implied inequalities. Hence, one comparison of importance is the number of inequalities implied in each algorithm; this section summarises such a comparison for the model of (21).

Table 1 shows that up to $n_c = 3$ the number of constraints for both the nominal and robust algorithm isn't prohibitively large. Theoretically, the number of constraints for algorithm 3.1 is expected to at least double for each increment of

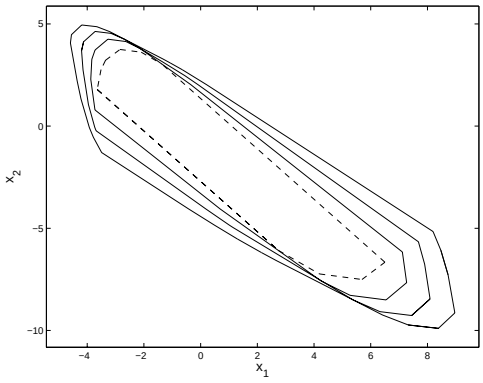


Fig. 3. Target region (dashed) and feasible regions for $n_c = 1, 2, 3$ degrees of freedom (solid) in the non-symmetric case.

	Number of inequalities	
n_c	Alg. 2.1	Alg. 3.1
1	13	17
2	15	31
3	19	56

Table 1. Comparison of the number of inequalities for both algorithms.

n_c, but due to the off-line elimination of redundant constraints, the increase actually is less than twofold for each increment.

5. CONCLUSIONS

This paper has made one major contribution, that is to develop a robust MPC algorithm for LPV systems that requires only a QP optimiser on line but nevertheless is guaranteed feasible and convergent in the largest possible region for a given number of free control moves and fixed terminal control law. This paper relies on the recent advance of (Pluymers *et al.*, 2005). It has been shown through numerical illustrations and argument that: (i) the algorithm outperforms nominal MPC algorithms due to improved robustness and (ii) outperformns other robust algorithms due to the increased volume of the feasible region.

REFERENCES

Bemporad, A., M. Morari, V. Dua and E.N. Pistokopoulos, The explicit linear quadractic regulator for constrained systems, *Automatica*, 38(1):3–20, 2002.

Blachini, F., 1999, Set invariance in control, *Automatica*, 35:1747–1767.

Cannon, M., B. Kouvaritakis and J.A. Rossiter, Efficient active set optimisation in triple mode MPC, *IEEE Trans. AC*, 46(8):1307–1313, 2001.

Cannon, M., Deshmukh, V., Lee, Y.I., and B. Kouvaritakis, 2002, Enlargement of polytopic terminal region in NMPC by interpolation and partial invariance, CDC.

Gilbert, E.G. and K. T. Tan, 1991, Linear systems with state and control constraints: the theory and application of maximal output admissable sets, *IEEE Trans. AC*, 36(9):1008–1020.

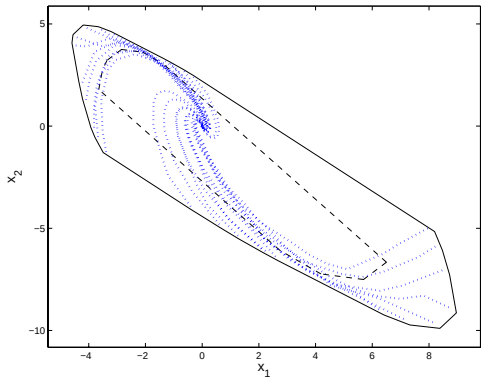


Fig. 4. Closed-loop simulations with $n_c = 3$ for different initial states near the boundary of the feasible region.

Imsland, L., Bar, N. and B. Foss, 2004, A new algorithm for efficient MPC and a comparison with competing schemes, ACC.

Kerrigan, E., Robust Constraint Satisfaction: Invariant Sets and Predictive Control, Ph.D Thesis, 2000, Cambridge

Kothare, M.V., Balakrishnan, V. and Morari, M., 1996, Robust constrained model predictive control using linear matrix inequalities, *Automatica*, 32:1361–1379.

Kouvaritakis, B., J.A. Rossiter and J. Schuurmans, Efficient robust predictive control, *IEEE Trans. AC*, 45(8):1545–1549, 2000.

Mayne, D.Q., Rawlings, J.B., Rao, C.V. and Scokaert, P.O.M., 2000, Constrained model predictive control: stability and optimality, *Automatica*, 36:789–814.

Byung-Gun Park, Jae-Won Lee, and Wook Hyun Kwon. Robust one-step receding horizon control for constrained systems. *Int. Jour. of Rob.and Nonlinear Cont.*, 9:381–395, 1999.

Pluymers, B., Suykens, J.A.K. and De Moor, B., 2004, Robust Finite-Horizon MPC using Optimal Worst-Case Closed-Loop Predictions, Accepted CDC'05.

Pluymers, B., Rossiter, J.A., Suykens, J. and De Moor, B., The efficient computation of polyhedral invariant sets for linear systems with polytopic incertainty, submitted ACC

Rossiter, J.A., M.J. Rice, and B. Kouvaritakis. A numerically robust state-space approach to stable predictive control strategies. *Automatica*, 38(1):65–73, 1998.

Rossiter, J.A., 2003, Model based predictive control, CRC press

Schuurmanns, J., and J.A. Rossiter, Robust piecewise linear control for polytopic systems with input constraints, *IEE Proc. Cont. Theory and Appl.*, 2000, 147(1):13–18

Scokaert, P.O.M. and Rawlings, J.B., 1998, Constrained linear quadratic regulation, *IEEE Trans on AC*, 43(8):1163–1168.

Wan, Z. and M.V. Kothare, 2003, Efficient robust constrained model predictive control with a time varying terminal constraint set, *Sys. and Cont. Let.*, 48:375–383