# Comments on "Min-Max Predictive Control Strategies for Input-Saturated Polytopic Uncertain Systems"

B. Pluymers[†], M.V. Kothare[‡], J.A.K. Suykens[†], B. De Moor[†]

† *Katholieke Universiteit Leuven*
*Department of Electrical Engineering, ESAT-SCD-SISTA*
*Kasteelpark Arenberg 10, B-3001 Heverlee (Leuven), Belgium*
*E-Mail : {bert.pluymers, johan.suykens, bart.demoor}@esat.kuleuven.be*

‡ *Chemical Process Modeling and Control Research Center*
*Lehigh University, Bethlehem, PA 18015, USA*
*E-mail : mayuresh.kothare@lehigh.edu*

**Abstract**

This paper reports an error in the recursive feasibility and stability proofs of the paper [Casavola *et al.*, Automatica 36, 125–133]. A counterexample is provided, indicating that the algorithm proposed in that paper is not guaranteed to be recursively feasible. A correction to the original algorithm is formulated to solve the feasibility problem and is illustrated using the same example.

*Key words:* Model predictive control, min-max feedback MPC

## 1 Introduction

This paper discusses the MPC algorithms introduced in [1], developed for input-constrained linear systems with polytopic uncertainty description. The authors state that the algorithms are recursively feasible and asymptotically stable. In order to obtain these two properties, the authors make use of a terminal cost and a corresponding constraint set [2] that is recalculated at each time step using the results of [3]. Due to the lack of imposing the terminal constraint on the set of terminal states in the optimization problem of Algorithm 1 of [1] (in further sections referred to as Algorithm 1$^\star$, with similar notation for Algoritm 2 of [1]), it is not guaranteed that the terminal controller will satisfy the input constraints for these terminal states and hence recursive feasibility is not guaranteed. Furthermore, explicitly imposing this terminal constraint in the on-line optimization without other modifications also doesn't guarantee feasibility. This is due to the fact that the algorithm optimizes over a deterministic input sequence, i.e. an input sequence that does not explicitly depend on the state evolution within the horizon, and hence in general there are not enough degrees of freedom over which to optimize in order to keep all possible values of the terminal state within the terminal constraint set. Algorithm 2$^\star$ suffers from a similar deficiency, but for reasons of brevity we will focus on Algorithm 1$^\star$ in this paper.

This paper is organized as follows. Section 2 discusses the deficiencies in Algorithm 1$^\star$ and gives an example that leads to violation of the assumed recursive feasibility property. Section 3 then provides a correction to this algorithm and proves its recursive feasibility and asymptotic stability. The same example as in Section 2 is used to demonstrate the efficacy of the proposed correction.

In the rest of this paper $^\star$-notation is used to denote theorems, equations and algorithms of the original article [1]. We refer to the original article for further details about other notations.

## 2 Original algorithm

### *2.1 Algorithm deficiencies*

Algorithm 1$^\star$ extends the results of [3] by adding $N$ free control moves $u(t|t), \ldots, u(t+N-1|t)$ to the formulation. These control moves are meant to drive the system inside the terminal constraint set $\mathscr{E}(t)$, which is taken to be the invariant ellipsoid corresponding to a terminal controller $F(t)$ constructed using [3]. At each time step

$t \geq 0$, the terminal controller and its invariant ellipsoid, are recomputed to take into account the (assumed) fact that the system has been driven closer to the origin. For reasons of brevity we refrain from entirely restating the algorithm, but refer to [1] for the details.

Two main deficiencies exist in Algorithm $1^\star$ :

- The terminal constraint $\mathscr{E}(t)$ is not explicitly imposed on the terminal states in optimization problem $(28)^\star$-$(29)^\star$, which corresponds to step 1 of Algorithm $1^\star$ and as a result feasibility of the terminal controller is not guaranteed.
- The input sequence $u(t|t), \ldots, u(t+N-1|t)$ is deterministic, which means that it does not explicitly depend on the state evolutions within the horizon. Therefore there are not enough degrees of freedom in the optimization to be able to guarantee recursive feasibility or asymptotic stability, even if the controller is initially feasible.

As a result of these deficiencies, Lemma $3^\star$ does not hold in general. Given an optimal input sequence $u^{\mathrm{o}}(\cdot|t)$ to $(28)^\star$-$(29)^\star$ at time $t$, expression $(37)^\star$ is used as a candidate feasible input sequence to $(28)^\star$-$(29)^\star$ at time $t+1$. However, due to the fact that the terminal constraint $z \in \mathscr{E}(t), \forall z \in \mathrm{vert}\{\mathscr{X}_{u^{\mathrm{o}}(\cdot|t)}^{t+N|t}(x(t))\}$ is not explicitly imposed in $(28)^\star$-$(29)^\star$, this input sequence $\bar{u}^*(\cdot|t+1)$ is not guaranteed to be feasible, since $F(t)x^{\mathrm{o}}(t+N|t)$ is not guaranteed to be feasible $\forall x^{\mathrm{o}}(t+N|t) \in \mathscr{X}_{u^{\mathrm{o}}(\cdot|t)}^{t+N|t}(x(t))$, which contradicts with the paragraph between $(36)^\star$ and $(38)^\star$.

Assume that it would be guaranteed $\forall t$ that $\mathscr{X}_{u^{\mathrm{o}}(\cdot|t)}^{t+N|t}(x(t)) \subseteq \mathscr{E}(t)$ and therefore that $F(t)x^{\mathrm{o}}(t+N|t) \in \Omega_{\mathrm{u}}$, even then monotonicity of the cost is not guaranteed. The expression right after $(38)^\star$ assumes that the terminal cost value at time $t+1$ resulting from input sequence $\bar{u}^*(\cdot|t+1)$ can be upper bounded by

$$\max_{i \in \{1,\ldots,l\}, z \in \mathrm{vert}\{\mathscr{X}_{u^{\mathrm{o}}(\cdot|t)}^{t+N|t}(x(t))\}} \|\Phi_{F(t),i} z\|_{Q(t+1)}^2, \qquad (1)$$

which in turn is based on the incorrect assumption that in general

$$\mathscr{X}_{\bar{u}^*(\cdot|t+1)}^{t+N+1|t+1}(x(t+1)) \subseteq \{\Phi_{F(t),i} z | i \in \{1,\ldots,l\}, z \in \mathscr{X}_{u^{\mathrm{o}}(\cdot|t)}^{t+N|t}(x(t))\}. \qquad (2)$$

Due to invariance of $\mathscr{E}(t)$ it can be shown that the RHS of the above expression is a subset of $\mathscr{E}(t)$. On the other hand, due the specific choice of $\bar{u}^*(\cdot|t+1)$, it can be seen that

$$\mathscr{X}_{\bar{u}^*(\cdot|t+1)}^{t+N+1|t+1}(x(t+1)) = \{\Phi_i z + G_i F(t) x^{\mathrm{o}}(t+N|t), i \in \{1,\ldots,l\}, z \in \mathscr{X}_{u^{\mathrm{o}}(\cdot|t)}^{t+N|t}(x(t))\}, \qquad (3)$$

which, in general, is not a subset of $\mathscr{E}(t)$, since one cannot necessarily find a fixed control move (i.e. $F(t)x^{\mathrm{o}}(t+N|t)$), that steers all terminal states further into the invariant ellipsoid $\mathscr{E}(t)$. This wrong assumption also lies at the basis of the error found in [4] and corrected in [5,6]. The main consequence is that the inclusion (2) does not hold in general, that the terminal cost at time $t+1$ in general cannot be bounded by (1) and that therefore monotonicity of $W(t)$ is not guaranteed. Similarly, recursive feasibility cannot be guaranteed anymore, since in general it is not guaranteed that $\mathscr{E}(t+1) \subset \mathscr{E}(t)$.

By means of similar arguments one can also invalidate both claims of Lemma $2^\star$.

### *2.2 Counterexample*

In order to illustrate these findings, we consider a system of the form $(1)^\star$-$(3)^\star$, with $l = 2$, defined as follows :

$$\Phi_1 = \begin{bmatrix} 1 & 0 \\ -0.3 & 1.3 \end{bmatrix}, \quad \Phi_2 = \begin{bmatrix} 1 & 0 \\ -0.1 & 1.1+\delta \end{bmatrix}, \qquad (4a)$$

$$G_1 = [0.2 \quad 0]^{\mathrm{T}}, \qquad G_2 = [1 \quad 0]^{\mathrm{T}}, \qquad (4b)$$

with $\delta \in [0,1]$ a parameter that is fixed in time and known during the controller synthesis. For $\delta = 0$, this system is identical to the one used in [5]. The system is subject to input constraint $|u(t)| \leq 1, \forall t \geq 0$. Cost matrices are chosen as $\Psi_x = \mathrm{diag}(1, 0.1)$ and $\Psi_u = 0.001$. The controller horizon is chosen as $N = 2$.

Figure 1 shows simulation results starting from initial state $[1; 1]$ for 4 different values of $\delta$. The algorithm is initially feasible for all values of $\delta$, but becomes infeasible after a few time steps for the two largest values of $\delta$. Also the function $W(t)$ is not monotonically decreasing. The original example in [1] does not exhibit these problems because the system is already stable in open-loop.

## 3 Corrected algorithm

### *3.1 Within-horizon feedback*

In order to correct the deficiencies of Algorithm $1^\star$, two modification are proposed :

- explicitly impose the terminal constraint set in the optimization problem $(28)^\star$-$(29)^\star$,

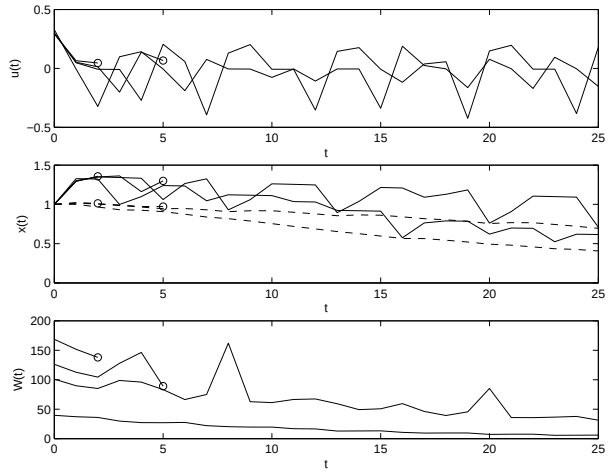


Fig. 1. Application of Algorithm 1$^\star$ to system (4) for initial state $[1;1]$ and $\delta \in \{0, 0.016, 0.018, 0.020\}$. The algorithm does not result in a monotonically decreasing Lyapunov function $W(t)$ and becomes infeasible for the latter two values of $\delta$ at time steps $t = 3$ and $t = 6$ respectively.

- use a state-dependent input sequence, similar to the one proposed in [7], allowing within-horizon feedback.

The modified algorithm will use a tree of within-horizon inputs $u_{j_p,\dots,j_1}(t+k|t)$ with $j_{1\dots p} = 1,\dots,l$ and $p = 0,\dots,N-1$, that can be interpreted as the inputs to be applied to the system at time $t+k$ if the real system is described by the models $[\Phi_{j_i}\ G_{j_i}]$ at times $t+i-1$ with $i = 1,\dots,p$. For notational brevity we now define

$$\mathbf{u}_j(t+k|t) \triangleq \begin{bmatrix} u_{1,\dots,1,j}(t+k|t) \\ u_{2,\dots,1,j}(t+k|t) \\ \vdots \\ u_{l-1,\dots,l,j}(t+k|t) \\ \underbrace{u_{l,\dots,l,j}}_{k}(t+k|t) \end{bmatrix}, \quad k = 0,\dots,N-1, \tag{5}$$

$\mathbf{u}(t+k|t) \triangleq [\mathbf{u}_1(t+k|t)^{\mathrm{T}} \dots \mathbf{u}_l(t+k|t)^{\mathrm{T}}]^{\mathrm{T}}$ and $\mathbf{u}_N(t) \triangleq [\mathbf{u}(t|t)^{\mathrm{T}} \dots \mathbf{u}(t+N-1|t)^{\mathrm{T}}]^{\mathrm{T}}$. A corresponding state prediction tree $x_{j_k,\dots,j_1}(t+k|t)$ with $j_{1\dots k} = 1,\dots,l$ and $k = 1,\dots,N$ is constructed as

$$x_{j_k,j_{k-1},\dots,j_1}(t+k|t) = \Phi_{j_k} x_{j_{k-1},\dots,j_1}(t+k-1|t) + G_{j_k} u_{j_{k-1},\dots,j_1}(t+k-1|t), \tag{6}$$

This leads to a new definition of the state prediction polytope :

$$\mathscr{X}^{t+N|t}_{\mathbf{u}_N(t)}(x(t)) = \mathrm{Co}\{x_{j_N,\dots,j_1}(t+N|t), j_{1\dots N} = 1,\dots,l\}. \tag{7}$$

### *3.2 Algorithm formulation*

We can now state the corrected algorithm in a straightforward way :

**Algorithm 1 :** *At time $t = 0$, given $x(0)$ solve the initialization step of Algorithm 1$^\star$, with the updated definitions of the input sequence $\mathbf{u}_N(t)$ (instead of $u^*(\cdot|t)$) and state prediction polytope* (7)*. At every time $t \geq 0$ execute the following steps :*

*(1) Given $x(t), P(t), \rho(t)$, solve the following optimization :*

$$\mathbf{u}^{\mathrm{o}}_N(t) = \underset{J_i, \mathbf{u}_N(t)}{\operatorname{argmin}} \sum_{i=0}^{j} J_i, \tag{8}$$

*subject to*

$$\begin{bmatrix} J_N & * \\ z & \rho^{-1}(t)P(t) \end{bmatrix} \geq 0, \quad \forall z \in \mathrm{vert}\{\mathscr{X}^{t+i|t}_{\mathbf{u}_N(t)}(x(t))\}, \tag{9}$$

$$J_N \leq \rho(t), \tag{10}$$

$$u_{j_k,\dots,j_1}(t+k|t) \in \Omega_{\mathrm{u}}, \quad j_{1\dots k} = 1,\dots,l, \quad k = 0,\dots,N-1, \tag{11}$$

$$\begin{bmatrix} 1 & * & * \\ \Psi_{\mathrm{x}}^{\frac{1}{2}} x_{j_k,j_{k-1},\dots,j_1}(t+k|t) & J_i I & * \\ \Psi_{\mathrm{u}}^{\frac{1}{2}} u_{j_k,j_{k-1},\dots,j_1}(t+k|t) & 0 & J_i I \end{bmatrix} \geq 0, \quad j_{1\dots k} = 1,\dots,l, \quad k = 0,\dots,N-1. \tag{12}$$

*(2) Apply $u^{\mathrm{o}}(t|t)$ to the plant.*
*(3) Calculate $[P(t+1), Y(t+1), \rho(t+1)]$ using step 3 of Algorithm 1$^\star$, with the updated definition of the state prediction polytope* (7)*.*

**Theorem 1 (Feasibility and Stability)** *Algorithm 1 is recursively feasible if the initialization step is feasible, in which case it also asymptotically stabilizes the system.*

**Proof :** Since step 1 is identical to the initialization step, except for the fixation of the terminal constraint and cost, it is straightforward that the former is feasible if the latter is feasible. This shows that step 1 is feasible at $t = 0$. If step 1 is feasible at time $t$, then step 3 is also feasible at time $t$. This is due to the fact that all terminal states lie inside $\mathscr{E}(t)$, since this is imposed by constraint (10). Therefore it is also guaranteed, due to invariance, that all states $\Phi_{F(t),i} z$ used in step 3 of Algorithm 1 also

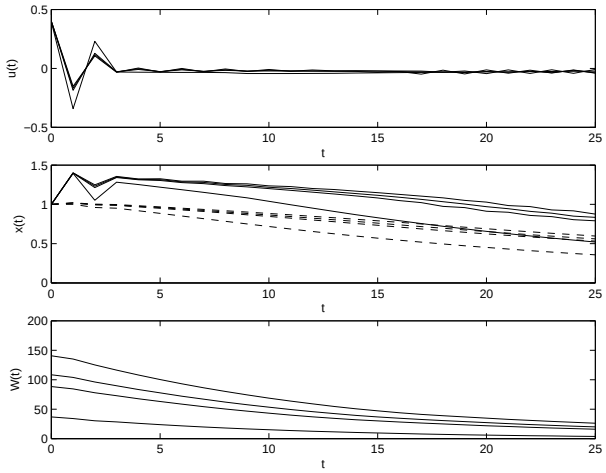


Fig. 2. Application of Algorithm 1 to system (4) for initial state $[1; 1]$ and $\delta \in \{0, 0.016, 0.018, 0.020\}$. The algorithm results in a monotonically decreasing Lyapunov function $W(t)$ and results in stable behaviour for all 4 values of $\delta$.

lie within $\mathscr{E}(t)$, which indicates that $P(t), Y(t), \rho(t)$ are feasible solutions to the optimization in this step. As a consequence the property $\rho(t+1) \leq \rho(t)$ also holds.

We now show that it is possible to construct a feasible solution to step 1 at time $t+1$. Because of $(2)^\star$, it is possible to find values $c_1, \ldots, c_l$, such that $x(t+1) = \sum_{i=1}^{l} c_i x_i^{\mathrm{o}}(t+1|t)$. A feasible solution $\mathbf{u}_N^{\mathrm{f}}(t+1)$ can now be constructed as

$$\mathbf{u}^{\mathrm{f}}(t+k|t+1) = \sum_{i=1}^{l} c_i \mathbf{u}_i^{\mathrm{o}}(t+k|t), \quad k = 1, \ldots, N-1,$$

and

$$u^{\mathrm{f}}_{j_{N-1},\ldots,j_1}(t+N|t+1) = F(t) x^{\mathrm{f}}_{j_{N-1},\ldots,j_1}(t+N|t+1)), \quad j_{1\ldots N-1} = 1, \ldots, l.$$

The corresponding state predictions can be expressed similarly as

$$\mathbf{x}^{\mathrm{f}}(t+k|t+1) = \sum_{i=1}^{l} c_i \mathbf{x}_i^{\mathrm{o}}(t+k|t), \quad k = 1, \ldots, N,$$

and

$$x^{\mathrm{f}}_{j_N,\ldots,j_1}(t+N+1|t+1) = \Phi_{F(t),j_N} x^{\mathrm{f}}_{j_{N-1},\ldots,j_1}(t+N|t+1), \quad j_{1\ldots N} = 1, \ldots, l.$$

One can see that this input sequence satisfies $(7)^\star$ for $k = 1, \ldots, N-1$ and, because all possible states $x(t+N|t+1) \in \mathscr{E}(t)$, also for $k = N$. Due to the specific choice of $\mathbf{u}^{\mathrm{f}}(t+N|t+1)$ and the way $\mathscr{E}(t+1)$ is calculated, one can see that the terminal constraint is also satisfied for the candidate input sequence proposed above.

Therefore step 1 of Algorithm 1 is also feasible at time $t+1$, which then proves recursive feasibility. Furthermore, due to convexity and because the state predictions corresponding to $\mathbf{u}_N^{\mathrm{f}}(t+1)$ can be expressed as convex combinations, one can see that

$$\mathscr{X}^{t+k|t+1}_{\mathbf{u}_N^{\mathrm{f}}(t+1)}(x(t+1)) \subseteq \mathscr{X}^{t+k|t}_{\mathbf{u}_N^{\mathrm{o}}(t)}(x(t)), k = 1, \ldots, N,$$

Combined with the observation that $Q(t)$ satisfies $(19)^\star$ and that $\rho(t+1) \leq \rho(t)$ this then shows that $W(t+1) \leq W(t)$, which proves asymptotic stability, along the lines presented in [2]. □

Figure 2 shows the same simulation shown in Figure 1, now using Algorithm 1. Recursive feasibility is now obtained for all 4 values of $\delta$. Also, $W(t)$ now behaves monotonically.

**Acknowledgements.** Research supported by Research Council KUL: GOA AMBioRICS, CoE EF/05/006, several PhD/postdoc & fellow grants; Flemish Government: FWO: PhD/postdoc grants, projects, G.0407.02, G.0197.02, G.0141.03, G.0491.03, G.0120.03, G.0452.04, G.0499.04, G.0211.05, G.0226.06, G.0321.06, G.0553.06, research communities (ICCoS, ANMMM, MLDM); IWT: PhD Grants,GBOU (McKnow), Eureka-Flite2; Belgian Federal Science Policy Office: IUAP P5/22; PODO-II (CP/40); EU: FP5-Quprodis; ERNSI; Contract Research/agreements: ISMC/IPCOS, Data4s, TML, Elia, LMS, Mastercard; Bert Pluymers is a research assistant with the I.W.T. at the KULeuven, Belgium. Dr. Johan Suykens is an associate professor and Dr. Bart De Moor is a full professor at the KULeuven, Belgium.

## References

[1] A. Casavola, M. Gianelli, E. Mosca, Min-max predictive control strategies for input-saturated polytopic uncertain systems, Automatica 36 (2000) 125–133.

[2] D. Q. Mayne, J. B. Rawlings, C. V. Rao, P. O. M. Scokaert, Constrained model predictive control: Stability and optimality, Automatica 36 (2000) 789–814.

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

[4] Z. Wan, M. V. Kothare, Efficient robust constrained model predictive control with a time varying terminal constraint set, Systems and Control Letters 48 (2003) 375–383.

[5] B. Pluymers, J. A. K. Suykens, B. De Moor, Min-max feedback MPC using a time-varying terminal constraint set and comments on efficient robust constrained model predictive control with a time-varying terminal constraint set, To appear in Systems and Control Letters.

[6] Z. Wan, B. Pluymers, M. V. Kothare, Corrections to "efficient robust constrained model predictive control with a time varying terminal constraint set", submitted to Systems and Control Letters, january 2006.

[7] P. O. M. Scokaert, D. Q. Mayne, Min-max feedback model predictive control for constrained linear systems, IEEE Transactions on Automatic Control 43 (8) (1998) 1136–1142.