Available online at www.sciencedirect.com
SCIENCE DIRECT®
Systems & Control Letters 54 (2005) 1143–1148
SYSTEMS & CONTROL LETTERS
www.elsevier.com/locate/sysconle

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"☆

B. Pluymers*, J.A.K. Suykens, B. De Moor

Department of Electrical Engineering, ESAT-SCD-SISTA, Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, B-3001 Heverlee (Leuven), Belgium

Received 3 June 2004; received in revised form 22 November 2004; accepted 7 April 2005
Available online 26 May 2005

Abstract

In this paper a robust MPC scheme using a time-varying terminal constraint set for input-constrained systems with a polytopic uncertainty description is proposed. The new scheme is a refinement of the algorithm published in "Efficient robust constrained model predictive control with a time-varying terminal constraint set" [Z. Wan, M.V. Kothare, Systems Control Lett. 48 (2003) 375–383]. The original result contains an error in the stability proof which is solved in this paper by introducing within-horizon feedback [P.O.M. Scokaert, D.Q. Mayne, IEEE Trans. Automat. Control 43(8) (1998) 1136–1142]. in the on-line algorithm. The new scheme is proven to be robustly asymptotically stabilizing but at the cost of an increased computational complexity.
© 2005 Elsevier B.V. All rights reserved.

Keywords: Model predictive control; Robust stability; Time-varying terminal constraint set

1. Introduction

We consider the problem of controlling a linear time-varying (LTV) system, described by

$$x(k+1) = A(k)x(k) + B(k)u(k), \tag{1}$$

with $x(k) \in \mathbb{R}^n$ denoting the state of the system at time k, $u(k) \in \mathbb{R}^m$ denoting the input of the system, subject to componentwise input constraints $|u_r(k)| \leqslant u_{r,\max}$ and $[A(k)\ B(k)] \in \Omega = \mathrm{Co}\{[A_1\ B_1], \ldots, [A_L\ B_L]\}$, with $\mathrm{Co}\{\cdot\}$ denoting the convex hull. No state constraints are considered and exact state measurements are assumed. By assumption, the desired state and input values correspond to the origins of $\mathbb{R}^n$ and $\mathbb{R}^m$, respectively. A nominal model $[\hat{A}\ \hat{B}]$, representing the most likely model of the true system is also assumed to be known. The aim is to construct a model-predictive controller to robustly asymptotically stabilize

☆ [Systems and Control Letters 48 (2003) 375–383].

* Corresponding author. Tel.: +32 1632 17 09; fax: +32 1632 19 70.
E-mail addresses: bert.pluymers@esat.kuleuven.ac.be (B. Pluymers), johan.suykens@esat.kuleuven.ac.be (J.A.K. Suykens), bart.demoor@esat.kuleuven.ac.be (B. De Moor).

0167-6911/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.sysconle.2005.04.001

(1) with an L_2-cost with state and input weighting matrices Q and R as optimality criterium.

The algorithm proposed in [4] addresses this issue by using a finite horizon length, within which a tree of robust state predictions is constructed based on a deterministic sequence of inputs, combined with a time-varying terminal constraint set that is imposed on the terminal states of the state prediction tree. The stability proof of the algorithm is based on the assertion that, due to the fact that the terminal constraint set is invariant with respect to a robustly stabilizing terminal controller, each terminal state is driven further inside the terminal constraint set. However, due to the choice of a deterministic input sequence, proving stability requires the construction of a single input vector that simultaneously drives all terminal states further inside the terminal constraint set. For unstable systems this cannot be guaranteed based upon the invariance property, that only guarantees that such an input vector exists for each terminal state individually. For this reason the algorithm proposed in [4] cannot be proven to be asymptotically stabilizing, neither can it be guaranteed to be feasible if it is initially feasible.

In this paper we present a new algorithm with a time-varying terminal constraint set that constructs a tree of input vectors, as was originally introduced in [3]. Due to this modification, the invariance property of the terminal constraint becomes a sufficient condition for stability of the controller. A simple numerical example is also provided, illustrating the flaw in the initially proposed algorithm as well as the improvement obtained with the new algorithm.

2. Time-varying terminal constraint set

In this section a brief description is given of the off-line part of the new algorithm, which is identical to that of the original algorithm. For the sake of brevity we will refer to equations of the original paper [4] with the notation $(\cdot)^*$. Similar notations will be used to refer to theorems, corollaries and algorithms.

We make use of the classical ingredients (a) *a terminal controller* $\kappa_f(\cdot) : \mathbb{R}^n \to \mathbb{R}^m$, (b) *a terminal cost* $\mathscr{F}(\cdot) : \mathbb{R}^n \to \mathbb{R}$ and (c) *a terminal constraint* $\mathscr{X}_f \subset \mathbb{R}^n$. The latter two elements can be interpreted as an upper bound to the (in this case nominal) control cost of the terminal controller, when applied at the end of the control horizon of the MPC controller. The terminal constraint represents a feasible invariant set associated with the terminal controller, given the forementioned input constraints. See [2,4] for details. Theorem $1^\star$ allows the construction of a set $(\mathscr{X}_f, \kappa_f(\cdot), \mathscr{F}(\cdot))$ parameterized by variables (γ, Q, X, Y) as $\mathscr{X}_f = \{x \in \mathbb{R}^n | x^T Q^{-1} x \leqslant 1\}$, $k_f(x) = YQ^{-1}x$ and $\mathscr{F}(x) = x^T \gamma Q^{-1} x$, with (γ, Q, X, Y) satisfying $(5)^\star$, $(6)^\star$, $(9)^\star$ and $(11)^\star$ for a given $r > 0$, denoting the radius of a hyperball inscribed in $\mathscr{X}_f$. $\mathscr{X}_f$ is an invariant ellipsoid with respect to $\kappa_f(\cdot)$, meaning that

$$x \in \mathscr{X}_f \Rightarrow A(k)x + B(k)\kappa_f(X) \in \mathscr{X}_f \quad \forall [A(k)\ B(k)] \in \Omega. \tag{2}$$

Corollary $1^\star$ then allows the construction of a continuum $(\mathscr{X}_f(\theta), \kappa_f(\theta, \cdot), \mathscr{F}(\theta, \cdot))$ based on two sets of parameters $(\gamma_1, Q_1, X_1, Y_1)$ and $(\gamma_0, Q_0, X_0, Y_0)$, obtained with Theorem $1^\star$ for two values r_0 and r_1 with $r_1 > r_0$, by considering the linear combination

$$(\gamma(\theta), Q(\theta), X(\theta), Y(\theta)) = \theta(\gamma_1, Q_1, X_1, Y_1) + (1-\theta)(\gamma_0, Q_0, X_0, Y_0), \tag{3}$$

and constructing the corresponding $(\mathscr{X}_f(\theta), \kappa_f(\theta, \cdot), \mathscr{F}(\theta, \cdot))$ as in Theorem $1^\star$. Algorithm $1^\star$ makes use of Theorem $1^\star$ and Corollaries $1^\star$ and $2^\star$ to construct $(\gamma_0, Q_0, X_0, Y_0)$ and $(\gamma_1, Q_1, X_1, Y_1)$ and the corresponding continuum of terminal constraint sets in a practical way.

3. Recursive feasibility

In general, in order to prove stability of an MPC algorithm, one first has to prove recursive feasibility of the optimization problem. This means that, given a solution to the optimization problem at time k, it should be possible to construct a feasible solution for the problem at time $k+1$.

The proof of Theorem $2^\star$ asserts that, given optimal solutions $U^\circ(k), X^\circ(k), \theta^\circ(k)$ at time k, one can find a feasible input sequence

$$U^f(k+1) = [u^\circ(k+1|k)^T, \ldots, u^\circ(k+N-1|k)^T, u^f(k+N|k+1)]^T, \tag{4}$$

for the optimization problem at time $k+1$, such that the corresponding state prediction set $\mathscr{X}^{\mathrm{f}}(k+N+1|k+1)$ satisfies $\mathscr{X}^{\mathrm{f}}(k+N+1|k+1) \subset \mathscr{X}_{\mathrm{f}}(\theta^{\circ}(k))$, by applying the terminal controller $\kappa_{\mathrm{f}}(\theta^{\circ}(k), \cdot)$ to the terminal state $x(k+N|k)$.

However, this terminal state is not uniquely determined due to the unknown coefficients c_{i,j_i}, $j_i = 1, \ldots, L$, $i = 1, \ldots, N$ (cf. expressions between (3)* and (4)*). It is therefore unclear what value to choose for $u^{\mathrm{f}}(k+N|k+1)$. Neither can it be guaranteed that there exists any value for $u^{\mathrm{f}}(k+N|k+1)$, such that $\mathscr{X}^{\mathrm{f}}(k+N+1|k+1) \subset \mathscr{X}_{\mathrm{f}}(\theta^{\circ}(k))$. This would require that

$$Ax + Bu^{\mathrm{f}}(k+N|k+1) \in \mathscr{X}_{\mathrm{f}}(\theta^{\circ}(k)) \quad \forall [A\ B] \in \Omega,\ \ \forall x \in \mathscr{X}_{\mathrm{f}}(\theta^{\circ}(k)), \tag{5}$$

while set invariance of $\mathscr{X}_{\mathrm{f}}(\theta^{\circ}(k))$ only guarantees that

$$Ax + B\kappa_{\mathrm{f}}(\theta^{\circ}(k), x) \in \mathscr{X}_{\mathrm{f}}(\theta^{\circ}(k)) \quad \forall [A\ B] \in \Omega,\ \ \forall x \in \mathscr{X}_{\mathrm{f}}(\theta^{\circ}(k)), \tag{6}$$

which means that a different input vector is used for each $x \in \mathscr{X}_{\mathrm{f}}(\theta^{\circ}(k))$, which conflicts with condition (5). Therefore, it is not always possible to find an appropriate value for $u^{\mathrm{f}}(k+N|k+1)$ and hence $\theta^{\circ}(k)$ is not necessarily non-decreasing, neither is recursive feasibility guaranteed.

However, in the case that $\mathscr{X}_{\mathrm{f}}(\theta^{\circ}(k))$ is invariant with respect to the open-loop model $x_{k+1} = A(k)x_k$ with $A(k) \in \mathrm{Co}\{A_1, \ldots, A_L\}$, one can choose $u^{\mathrm{f}}(k+N|k+1) = 0$. This suggests an explanation why the example in [1] does not result in unstable closed-loop behaviour. This also suggests the possible successful application of the original algorithm to certain classes of stable systems.

4. Feedback MPC formulation

In this section we present the on-line part of the new algorithm and the different modes of operation that are employed. The focus is on the introduction of feedback within the horizon as was proposed in [3], in order to be able to guarantee recursive feasibility.

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

$$\mathbf{u}_j(k+p|k) \triangleq \begin{bmatrix} u_{1,\ldots,1,j}(k+p|k) \\ u_{2,\ldots,1,j}(k+p|k) \\ \vdots \\ \underbrace{u_{L,\ldots,L,j}}_{p}(k+p|k) \end{bmatrix}, \quad p = 0, \ldots, N-1, \tag{7}$$

$\mathbf{u}(k+p|k) \triangleq [\mathbf{u}_1(k+p|k)^{\mathrm{T}}, \ldots, \mathbf{u}_L(k+p|k)^{\mathrm{T}}]^{\mathrm{T}}$ and $\mathbf{u}_N(k) \triangleq [\mathbf{u}(k|k)^{\mathrm{T}}, \ldots, \mathbf{u}(k+N-1|k)^{\mathrm{T}}]^{\mathrm{T}}$. This tree of inputs implicitly defines a control policy for all possible combinations of $[A(k+p)\ B(k+p)] \in \Omega$, $p = 0, \ldots, N-1$, since these can be described as convex combinations of the nodes of the polytope Ω^{N-1}. A state prediction tree $x_{j_p,\ldots,j_1}(k+p|k)$ with $j_{1\ldots p} = 1, \ldots, L$ and $p = 1, \ldots, N$ is constructed as

$$x_{j_p,j_{p-1},\ldots,j_1}(k+p|k) = A_{j_p} x_{j_{p-1},\ldots,j_1}(k+p-1|k) + B_{j_p} u_{j_{p-1},\ldots,j_1}(k+p-1|k). \tag{8}$$

Similar notations $\mathbf{x}_j(k+p|k)$ and $\mathbf{x}(k+p|k)$ with $p = 0, \ldots, N$ and $\mathbf{x}_N(k)$ will be used in further sections.

In an initial phase (*mode* 1), an optimization is performed over the inputs $\mathbf{u}_N(k)$ in order to minimize the size of the terminal constraint set, characterized by the parameter θ:

$$\min_{\mathbf{u}_N(k),\theta} \theta \tag{9a}$$

subject to

$$|u_{j_p,\ldots,j_1}(k+p|k)| < u_{\max}, \quad j_{1\ldots p} = 1, \ldots, L,\ \ p = 0, \ldots, N-1, \tag{9b}$$

$$x_{j_N,\ldots,j_1} \in \mathscr{X}_{\mathrm{f}}(\theta), \quad j_{1\ldots N} = 1, \ldots, L, \tag{9c}$$

$$0 \leqslant \theta \leqslant 1. \tag{9d}$$

The optimal value of θ at time k is denoted as $\theta^{\circ}(k)$.

A second phase (*mode* 2) is initiated when at the previous time step $k-1$ the smallest terminal constraint $\theta^{\circ}(k-1) = 0$ is obtained. In this phase the horizon length is reduced with 1 at each time step ($N := N-1$) and an MPC problem with a nominal cost objective is solved. For the sake of clarity and without loss of generality, we assume that the nominal system model is

given by $[\hat{A}\ \hat{B}]=[A_1\ B_1]$. The nominal state and input predictions now become $\hat{x}(k+p|k)=x_{1,\ldots,1}(k+p|k)$ and $\hat{u}(k+p|k)=u_{1,\ldots,1}(k+p|k)$. The following optimization problem is obtained:

$$\min_{\mathbf{u}_N(k)} \sum_{i=0}^{N-1} \|\hat{u}(k+i|k)\|_R + \sum_{i=0}^{N-1} \|\hat{x}(k+i|k)\|_Q + \|\hat{x}(k+N|k)\|_{\gamma_0 Q_0^{-1}}, \tag{10}$$

subject to (9b) and (9c) with $\theta=0$.

If at the previous time step $k-1$ a mode 2 optimization problem was solved with $N=1$, a third and final phase (*mode* 3) is initiated. In this mode, the current state $x(k)$ is guaranteed to be positioned within the smallest terminal constraint $\mathscr{X}_f(0)$ and therefore the corresponding terminal control law $u(k)=\kappa_f(0,x(k))=Y_0Q_0^{-1}x(k)$ is applied to further drive the system to the origin.

These 3 modes of operation can be summarized in the following new algorithm:

Algorithm 1. *Initialize* $N:=N_0$ *and mode* $:=1$. *Given a system description* (1), *input constraints* $u_{\max}$, *state and input weighting matrices* Q *and* R *and two sets* $(\gamma_0, Q_0, X_0, Y_0)$ *and* $(\gamma_1, Q_1, X_1, Y_1)$ *calculated using Algorithm* $1^\star$ *and given the current state* $x(k)$, *perform at each time step k the following steps*:

- *If mode* $=1$, *solve optimization problem* (9) *and apply* $u(k|k)$ *to the system. If* $\theta^\circ(k)=0$ *set mode* $:=2$. *Wait until time step* $k+1$.
- *If mode* $=2$, *set* $N:=N-1$, *solve optimization problem* (10) *subject to* (9b) *and* (9c) *with* $\theta=0$ *and apply* $u(k|k)$ *to the system. If* $N=1$, *set mode* $:=3$. *Wait until time step* $k+1$.
- *If mode* $=3$, *apply* $u(k)=Y_0Q_0^{-1}x(k)$ *to the system. Wait until time step* $k+1$.

5. Feasibility and asymptotic stability

The following theorem is proven as a corrected version to Theorem $2^\star$.

Theorem 1. *Given a system description* (1), *input constraints* $u_{\max}$, *state and input weighting matrices* Q *and* R *and an initial horizon length* N_0, *if optimization problem* (9) *is feasible at time* $k=0$ *for the initial state* $x(0)$ *and* $N=N_0$, *then Algorithm* 1 *is also feasible for* $k>0$ *and robustly, asymptotically stabilizes* (1).

Proof. We prove that under the given assumptions modes 1 and 2 are feasible and terminate in a finite number of time steps. Therefore robust asymptotic stability is proven if mode 3 is robustly asymptotically stable.

First, we prove by induction that mode 1 is feasible and terminates in a finite amount of time. By assumption a feasible mode 1 solution $\mathbf{u}_N^\circ(k-1)$, $\mathbf{x}_N^\circ(k-1)$ and $\theta^\circ(k-1)$ exists for each $k>0$. We will now construct a feasible mode 1 solution $\mathbf{u}_N^f(k)$, $\mathbf{x}_N^f(k)$ and $\theta^f(k)$ with $\theta^f(k)<\theta^\circ(k-1)$. Since $[A(k)\ B(k)]\in\Omega$, it is possible to find values $c_1,\ldots,c_L$, with $c_i\geqslant 0$ and $\sum_{i=1}^L c_i=1$, such that $x(k|k)=\sum_{i=1}^L c_i x_i^\circ(k|k-1)$. Using these values $c_{1,\ldots,L}$ it is possible to construct a new set of feasible inputs

$$\mathbf{u}^f(k+p|k)=\sum_{i=1}^{L} c_i\mathbf{u}_i^\circ(k+p|k-1),\quad p=0,\ldots,N-2, \tag{11a}$$

and

$$u^f_{j_{N-1,\ldots,j_1}}(k+N-1|k) = \kappa_f(\theta^\circ(k-1), x^f_{j_{N-1,\ldots,j_1}}(k+N-1|k)), \tag{11b}$$

with $j_{1\ldots N-1}=1,\ldots,L$. A set of feasible states can be constructed likewise:

$$\mathbf{x}^f(k+p|k)=\sum_{i=1}^{L} c_i\mathbf{x}_i^\circ(k+p|k-1),\quad p=0,\ldots,N-1, \tag{12}$$

and $x^f_{j_N,\ldots,j_1}(k+N|k)$ can be defined using (8) for $p=N$. It is clear that through construction these sets of inputs and states satisfy (8) for $p=1,\ldots,N$. Due to the convexity of the terminal constraint set, it is also clear that by construction all the states from $\mathbf{x}^f(k+N-1|k)$ also lie within the terminal constraint set $\mathscr{X}_f(\theta^\circ(k-1))$. Due to the invariance of this terminal constraint set, the strict inequality in $(5)^\star$ and the construction of $\mathbf{u}^f(k+N-1|k)$, all terminal states from $\mathbf{x}^f(k+N|k)$ lie in the strict interior of $\mathscr{X}_f(\theta^\circ(k-1))$. Therefore $\inf_k(\theta^\circ(k-1)-\theta^\circ(k))>0$, which proves

the fact that mode 1 terminates in a finite amount of time.

In mode 2 the horizon length is decreased by 1 in each time step, so it is trivial to prove that the condition $N=1$ will be satisfied in a finite number (i.e. N_0-1) of time steps. By constructing a feasible mode 2 solution $\mathbf{u}^{\mathrm{f}}_{N-1}(k), \mathbf{x}^{\mathrm{f}}_{N-1}(k)$ using (11a) and (12) feasibility is also trivially guaranteed.

By means of a similar argument one can easily see that if $N=1$ at time $k-1$, the current state $x(k)$ will lie in the terminal constraint set $\mathscr{X}_{\mathrm{f}}(0)$, which legitimates the use of the terminal controller $\kappa_{\mathrm{f}}(0,\cdot)$ in this mode, since it robustly asymptotically stabilizes (1) for all initial states that belong to its invariant set $\mathscr{X}_{\mathrm{f}}(0)$. See [4,1] for details. This, combined with the above arguments, proves robust asymptotic stability of Algorithm 1. □

6. Example

In this section we present a numerical example that clearly illustrates the flaw in Algorithm 2* and shows the improvement obtained with Algorithm 1. Consider a system with $L=2, m=1, n=2$, described by

$$A_1=\begin{bmatrix}1 & 0\\ -0.3 & 1.3\end{bmatrix}, \quad A_2=\begin{bmatrix}1 & 0\\ -0.1 & 1.1\end{bmatrix}, \tag{13a}$$

$$B_1=[0.2\ \ 0]^{\mathrm{T}}, \qquad B_2=[1\ \ 0]^{\mathrm{T}}, \tag{13b}$$

with input constraint $|u|\leqslant 1$. Initial horizons $N_0 \in \{4,5\}$ and input and state cost matrices $Q=\mathrm{diag}(1,0.1)$ and $R=0.001$ were chosen. The radius of the largest inscribed ball was chosen as $r_1\in\{0.34,0.35\}$. The radius of the smallest inscribed ball was found to be $r_0=0.27411$. Simulations were performed on a time-invariant system described by $[A_2\ B_2]$ with initial state $x_0=[1\ 1]^{\mathrm{T}}$. The resulting total simulation control costs and computation times are given in Table 1, the sequence of $\theta^\circ(k)$-values for $N_0=4$ is depicted in Fig. 1. Algorithm 2* exhibits non-monotonically decreasing values of $\theta^\circ(k)$ in mode 1 for all forementioned values of r_1 and N_0 and becomes infeasible in mode 1 for $r_1=0.34$, $N_0=4$ due to (9d) and infeasible in mode 2 for $N_0=5$ due to (9c). Algorithm 1 exhibits monotonically decreasing values of $\theta^\circ(k)$ in mode 1 and is feasible and asymptotically stable for all forementioned values of r_1 and N_0, but at the cost of an increase in computational complexity.

Table 1
Total simulation control cost for Algorithm 2* and Algorithm 1 for $r_1\in\{0.34,0.34\}$ and $N_0\in\{4,5\}$

	$r_1=0.34$	$r_1=0.35$	
Alg. 2*, $N_0=4$	inf. at $k=2$	42.58	(0.39s)
Alg. 2*, $N_0=5$	inf. at $k=29$	inf. at $k=27$	(0.83s)
Alg. 1, $N_0=4$	30.74	30.73	(0.59s)
Alg. 1, $N_0=5$	30.79	30.78	(2.08s)

The maximum computation time per iteration (P4-2GHz, MATLAB 6.5, LMI LAB 1.0.8) for $r_1=0.35$ is indicated between brackets.

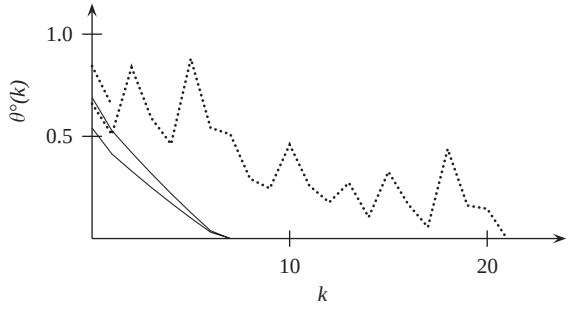

Fig. 1. Values of $\theta^\circ(k)$ for Algorithm 2* (dotted) and the new Algorithm 1 (solid) for $r_1=0.34$ (larger θ-values) and $r_1=0.35$ (smaller θ-values) and $N_0=4$. Note that Algorithm 2* becomes infeasible for $r_1=0.34$ at time $k=2$.

7. Conclusion

In this paper a corrected version of the algorithm in [4] is presented. The new algorithm can be proven to be robustly asymptotically stabilizing, but this goes at the cost of an increased computational complexity due to an increased number of optimization variables.

No attempt has been made to improve other apparent deficiencies of the original algorithm. One example is the fact that mode 1 of the algorithm does not take the within-horizon cost into account and can therefore lead to suboptimal control behaviour when the input weighting matrix R is given a relatively large value; another deficiency is the large maximum computational complexity per iteration of the algorithm. These issues form the subject of future research.

Acknowledgements

The authors would like to thank Jeroen Buijs (Group T Engineering School, Leuven) for the interesting and valuable discussions.

Research supported by Research Council KUL: GOA-Mefisto 666, GOA-Ambiorics, several Ph.D./postdoc & fellow grants; Flemish Government: FWO: Ph.D./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: Ph.D. 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 Sustainability); 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 IWT (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.

References

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

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

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