Katholieke Universiteit Leuven

Departement Elektrotechniek

ESAT-SISTA/TR 04-161

Constrained Linear MPC with Time-Varying Terminal Cost using Convex Combinations [1]

B. Pluymers, L. Roobrouck, J. Buijs, J.A.K. Suykens and B. De Moor [2]

May 2005
Published in *Automatica* 41 (2005) p. 831-837

[1]This report is available by anonymous ftp from *ftp.esat.kuleuven.ac.be* in the directory *pub/sista/pluymers/reports/aut05-ccmpc.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@esat.kuleuven.ac.be*. Bert Pluymers is a research assistant with the I.W.T. (Flemish Institute for Scientific and Technological Research in Industry). Jeroen Buijs was a research assistant with the Katholieke Universiteit Leuven at the time the research was initiated. Johan Suykens is a postdoctoral researcher with the FWO Flanders and a professor at the Katholieke Universiteit Leuven. Dr. Bart De Moor is a full professor at the Katholieke Universiteit Leuven, Belgium. Research supported by Research Council KUL: GOA-Mefisto 666, 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.0499.04 (robust statistics), G.0211.05 (nonlinear identification), G.0080.01 (collective behaviour), research communities (ICCoS, ANMMM); AWI: Bil. Int. Collaboration Hungary/ Poland; IWT: PhD Grants, Soft4s (softsensors), Belgian Federal Government: DWTC (IUAP IV-02 (1996-2001) and IUAP V-22 (2002-2006)), PODO-II (CP/40: TMS and Sustainability); EU: CAGE; ERNSI; Eureka 2063-IMPACT; Eureka 2419-FliTE; Contract Research/agreements: Data4s, Electrabel, Elia, LMS, IPCOS, VIB

Abstract

Recent papers have introduced dual-mode MPC algorithms using a time-varying terminal cost and/or constraint. The advantage of these methods is the enlargement of the admissible set of initial states whithout sacrificing local optimality of the controller, but this comes at the cost of a higher computational complexity. This paper delivers two main contributions in this area. First, a new MPC algorithm with a time-varying terminal cost and constraint is introduced. The algorithm uses convex combinations of off-line computed ellipsoidal terminal constraint sets and uses the associated cost as a terminal cost. In this way a significant on-line computational advantage is obtained. The second main contribution is the introduction of a general stability theorem, proving stability of both the new MPC algorithm and several existing MPC schemes. This allows a theoretical comparison to be made between the different algorithms. The new algorithm using convex combinations is illustrated and compared with other methods on the example of an inverted pendulum.

Constrained Linear MPC with Time-Varying Terminal Cost using Convex Combinations ⋆

B. Pluymers, L. Roobrouck, J. Buijs, 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, luc.roobrouck, johan.suykens, bart.demoor}@esat.kuleuven.ac.be, jeroen.buijs@groept.be

Abstract

Recent papers [3,4,13] have introduced dual-mode MPC algorithms using a time-varying terminal cost and/or constraint. The advantage of these methods is the enlargement of the admissible set of initial states whithout sacrificing local optimality of the controller, but this comes at the cost of a higher computational complexity. This paper delivers two main contributions in this area. First, a new MPC algorithm with a time-varying terminal cost and constraint is introduced. The algorithm uses convex combinations of off-line computed ellipsoidal terminal constraint sets and uses the associated cost as a terminal cost. In this way a significant on-line computational advantage is obtained. The second main contribution is the introduction of a general stability theorem, proving stability of both the new MPC algorithm and several existing MPC schemes [3,4]. This allows a theoretical comparison to be made between the different algorithms. The new algorithm using convex combinations is illustrated and compared with other methods on the example of an inverted pendulum.

Key words: Modelbased predictive control, Linear matrix inequalities, Stability, Optimality, Time-varying terminal cost

1 Introduction

Stability of MPC has been intensively studied in the last decade, resulting in the identification of three ingredients to impose stability : a locally stabilizing terminal feedback controller, a terminal state constraint and a terminal state cost. Several different stabilizing MPC-schemes where shown to fit in this framework in [10]. Recent publications have introduced the use of a time-varying terminal cost to impose stability for linear time-varying systems [6,8,13] or to achieve improved feasibility and optimality for linear time-invariant systems [3,4].

The main disadvantage of MPC using a time-varying terminal cost is the increase in computational complexity induced by the on-line optimization of the terminal cost and constraint. In this paper a new MPC scheme using a time-varying terminal cost and constraint is introduced for linear, time-invariant systems, further improving the computational advantage of the method proposed in [3]. The method uses a discrete set of terminal costs and constraints that are calculated off-line in order to compute a time-varying terminal cost and constraint on-line. This can be considered as an extension to the use of convex combinations as introduced in [13] in the context of robust MPC. Compared with the method from [3], the new method does not need an explicit decomposition of the terminal state, which leads to a decreased number of additional optimization variables. As a consequence, this leads to a further reduction in on-line computational complexity.

A general stability theorem is formulated, unifying MPC with fixed terminal cost and several MPC schemes using time-varying terminal cost, including the method introduced in this paper. The new algorithm extends the well-known results presented in [10] and leads to additional insights in the different algorithms discussed.

This paper is organized as follows. In Section 2, the general notation used in this paper and some necessary background knowledge is briefly explained. Section 3 introduces the new MPC scheme and compares its computational complexity with the methods published in [3,4]. Section 4 introduces a unifying stability theorem and Section 5 demonstrates the new method on a simple example.

⋆ This paper was not presented at any IFAC meeting. Corresponding author Bert Pluymers, E-mail bert.pluymers@esat.kuleuven.ac.be, Tel. +32-16-321129

2 Background

2.1 Model predictive control

When referring to Model-Predictive Control (MPC) for controlling a linear time-invariant system defined by

$$x(k+1) = Ax(k) + Bu(k), \quad k = 0, \ldots, \infty, \quad (1)$$

we will refer to a control scheme that solves at each time step k, given a value for $x(k) \in \mathbb{R}^{n_x}$, the following optimization problem :

$$\min_{\mathbf{x},\mathbf{u}} \quad J_{n_h}(x(k), \mathbf{x}, \mathbf{u}) \quad (2a)$$
$$\text{s.t.} \quad x(k+i+1|k) = Ax(k+i|k) + Bu(k+i|k), \quad i = 0, \ldots, n_h - 1, \quad (2b)$$

with $J_{n_h}(x(k), \mathbf{x}, \mathbf{u}) = \sum_{i=0}^{n_h-1} \|u(k+i|k)\|_R^2 + \sum_{i=0}^{n_h} \|x(k+i|k))\|_Q^2$, where $\|x\|_Q^2 \triangleq x^{\mathrm{T}} Q x$, after which $u(0|0) \in \mathbb{R}^{n_u}$ is applied to the plant. The scalars n_x and n_u respectively denote the number of states and inputs. $\mathbf{x} \in \mathbb{R}^{n_h \cdot n_x}$ and $\mathbf{u} \in \mathbb{R}^{n_h \cdot n_u}$ denote respectively the stacked vectors of within-horizon states and inputs, with $x(k+i|k)$ and $u(k+i|k)$ respectively denoting the system state and inputs at time $k+i$ as predicted at time k. $Q \in \mathbb{R}^{n_x \times n_x}$ and $R \in \mathbb{R}^{n_u \times n_u}$ are positive definite matrices denoting the state and input weighting matrices, while n_h denotes the horizon. $A \in \mathbb{R}^{n_x \times n_x}$ and $B \in \mathbb{R}^{n_x \times n_u}$ define the linear state-space prediction model used by the controller. Additional state and input constraints are denoted by

$$x(k+i|k) \in \mathbb{X}, \quad u(k+i-1|k) \in \mathbb{U}, \quad i = 1, \ldots, n_h, \quad (2c)$$

with $\mathbb{X} \subset \mathbb{R}^{n_x}$ and $\mathbb{U} \subset \mathbb{R}^{n_u}$ convex sets. In the following sections, we call this the *standard MPC* algorithm.

2.2 Stability of MPC

In general, MPC stability is obtained by changing the cost term of the last state of the horizon into $F(x(k+n_h|k))$, where $F(\cdot)$ is a convex function ($\mathbb{R}^{n_x} \to \mathbb{R}$) and by imposing the terminal constraint $x(k+n_h|k) \in \mathbb{X}_{n_h}$. Asymptotic stability can be proven if there also exists a terminal state feedback controller $\kappa(\cdot)$ such that the following conditions are satisfied :

$$a)\ \kappa(x) \in \mathbb{U}, \quad \forall x \in \mathbb{X}_{n_h}; \quad (3a)$$
$$b)\ Ax + B\kappa(x) \in \mathbb{X}_{n_h}, \quad \forall x \in \mathbb{X}_{n_h}; \quad (3b)$$
$$c)\ \mathbb{X}_{n_h} \subset \mathbb{X}; \quad (3c)$$
$$d)\ F(x) - F(Ax + B\kappa(x)) \geq \|x\|_Q^2 + \|\kappa(x)\|_R^2, \quad \forall x \in \mathbb{X}_{n_h}. \quad (3d)$$

When dealing with a linear system (1), and in case the state and input constraints $\mathbb{X}$ and $\mathbb{U}$ are defined as

$$|x_i| \leq x_{i,\max}, \quad i = 1, \ldots, n_x, \quad (4a)$$
$$|u_i| \leq u_{i,\max}, \quad i = 1, \ldots, n_u, \quad (4b)$$

then $F(\cdot), \kappa(\cdot)$ and $\mathbb{X}_{n_h}$ can be chosen as

$$\kappa(x) = Kx, \quad (5a)$$
$$F(x) = x^{\mathrm{T}} Q_{n_h} x, \quad (5b)$$
$$\mathbb{X}_{n_h} = \{x | x^{\mathrm{T}} Z^{-1} x \leq 1\} \quad (5c)$$

and can be calculated by solving the LMIs stated in [7] :

$$\min_{\gamma, Z, Y, X} \gamma \quad (6a)$$

subject to

$$\begin{bmatrix} 1 & * \\ \bar{x} & Z \end{bmatrix} \geq 0, \quad \begin{bmatrix} Z & * & * & * \\ Q^{\frac{1}{2}} Z & \gamma I & * & * \\ R^{\frac{1}{2}} Y & 0 & \gamma I & * \\ AZ + BY & 0 & 0 & Z \end{bmatrix} \geq 0, \quad (6b)$$

$$\begin{bmatrix} X & * \\ Y^{\mathrm{T}} & Z \end{bmatrix} \geq 0, \quad \mathrm{diag}(u_{\max}^2) \geq X, \quad (6c)$$

$$\begin{bmatrix} x_{i,\max}^2 Z & * \\ C_i Z & 1 \end{bmatrix} \geq 0, \quad i = 1, \ldots, n_x, \quad (6d)$$

with $C_i = [0\ 0\ \ldots\ 1\ \ldots\ 0\ 0]$ (i-th component). Asterisks are used to denote the corresponding transpose of the lower block part of symmetric matrices. While in the method proposed in [7] $\bar{x}$ represents the current state measurement and the above LMIs are recalculated at each time instant, in this context $\bar{x} \in \mathbb{X}$ is a state to be chosen by the user (called *canonical state*) determining the size of the resulting terminal constraint $\mathbb{X}_{n_h}$. The feedback matrix K and the closed loop Lyapunov function $F(x) = \|x\|_{Q_{n_h}} = x^{\mathrm{T}} Q_{n_h} x$ can now be calculated as

$$K = YZ^{-1}, \qquad Q_{n_h} = \gamma Z^{-1}. \quad (6e)$$

In the following sections, MPC using a terminal cost and controller (5), which is off-line calculated with the above optimization problem, is called *MPC with fixed terminal cost* or *F-MPC*. A related method was proposed in [5].

2.3 MPC with time-varying terminal cost

As already indicated in [4], a trade-off has to be made between feasibility (large $\|\bar{x}\|$) and local optimality (small $\|\bar{x}\|$). To solve this problem, an MPC scheme using the

following optimization was introduced [4] :

$$\min_{\mathbf{u},\gamma',X,Y,Z,\gamma} \gamma'+\gamma \quad \text{s.t.} \begin{cases} \begin{bmatrix} \gamma' - f^{\mathrm{T}}\mathbf{u} & \mathbf{u}^{\mathrm{T}} \\ \mathbf{u} & H^{-1} \end{bmatrix} \geq 0, \\ A_{\mathrm{ineq}}\mathbf{u} \leq b_{\mathrm{ineq}}, \end{cases} \quad (7)$$

and subject to (6b)-(6d) with $\bar{x} = \Psi_{n_h}\mathbf{u} + \psi_{n_h}$.

$\Psi_{n_h} = [A^{n_h-1}B\ A^{n_h-2}B\ \dots\ B]$ and $\psi_{n_h} = A^{n_h}x(0|0)$ define the dependence of the terminal state on the within-horizon inputs, while H and f represent the within-horizon control cost $\mathbf{u}^{\mathrm{T}}H\mathbf{u} + f^{\mathrm{T}}\mathbf{u}$. A_{ineq} and b_{ineq} denote the within-horizon input and state constraints. *In the following sections we will refer to this method as MPC with full time-varying terminal cost or FTV-MPC.* See [4] for details.

In [3] it was recognized that performing some of the optimization off-line and using on-line interpolation to calculate a time-varying terminal cost, results in a significant computational advantage. The method proposed there uses a set of n off-line calculated terminal constraints, costs and controllers defined by $Z_i, Q_{n_h,i}, K_i$, satisfying (3a)-(3d). At each time step the terminal state $x(k+n_h|k)$ is decomposed into n different components $\hat{x}_i$ in the following way :

$$x(k+n_h|k) = \sum_{i=1}^{n} \hat{x}_i, \begin{cases} \sum_{i=1}^{n} \lambda_i = 1; \lambda_i \geq 0, & \forall i; \\ x_i^{\mathrm{T}} Z_i^{-1} x_i \leq 1, & \forall i; \\ \hat{x}_i = \lambda_i x_i, & \forall i. \end{cases} \quad (8)$$

These components are then subjected to the respective control laws K_i, resulting in the following control law

$$u(k+n_h+j|k) = \sum_{i=1}^{n} K_i(A+BK_i)^j \hat{x}_i, \quad j \geq 0, \quad (9)$$

for which, through the use of an off-line convex optimization problem that is discussed in detail in [3], a quadratic cost function $T(x) = \tilde{x}^{\mathrm{T}}V\tilde{x}$ with $\tilde{x} = [\hat{x}_1^{\mathrm{T}}\ \hat{x}_2^{\mathrm{T}}\ \dots\ \hat{x}_n^{\mathrm{T}}]^{\mathrm{T}}$ can be calculated. By using this cost function as a terminal cost, by using constraints (8) as a terminal constraint and by adding the $\hat{x}_i$ and λ_i as optimization variables, an MPC scheme with time-varying terminal cost, but with reduced computational complexity is obtained. *In the following sections we refer to this method as MPC with time-varying terminal cost using state decomposition or SD-MPC.* See [3] for details.

3 Convex Combinations

In this section a different approach towards constructing a time-varying terminal cost and constraint is presented. We assume that we have a set of canonical states $\bar{x}_i, i = 1, \dots, n$, with $n > 0$ an integer chosen by the user. Assuming that we have solutions γ_i, X_i, Y_i and Z_i to (6b)-(6d) for these canonical states, it can be easily proven (based on the convexity of LMI's) that any convex combination

$$(\gamma, X, Y, Z) \equiv \sum_{i=1}^{n} \lambda_i(\gamma_i, X_i, Y_i, Z_i) \quad (10)$$

of these solutions, with $\lambda_i \geq 0, \sum \lambda_i \leq 1$ is a solution to (6b)-(6d) for $\bar{x} = \sum_{i=1}^{n} \lambda_i \bar{x}_i$, implying that the corresponding terminal controller, cost and constraint (using (6e) and (5a)-(5c)) also satisfies the MPC stability constraints (3a)-(3d). Note that in the strictest sense, convex combinations only allow $\sum_{i=1}^{n} \lambda_i = 1$, but since one can always find an arbitrarily small solution γ_0, X_0, Y_0, Z_0 to (6a)-(6d) for $\bar{x}_0 = 0$, the above is still valid, assuming this solution is incorporated in the convex combination with weight $\lambda_0 = 1 - \sum_{i=1}^{n} \lambda_i$. The idea is to calculate this discrete set of solutions off-line, while making the convex combinations on-line, thus eliminating the LMI's (6b)-(6d) from the on-line optimization problem. This leads to the following off-line algorithm for calculating the terminal constraints and terminal costs :

Algorithm 1 ***[CC-MPC (off-line)] :*** *Given a model* (1)*, state and input constraints* (4)*, weighting matrices* Q *and* R*, horizon* n_h *and a positive integer* n*, calculate* $X_i, Y_i, Z_i, \gamma_i, i = 1, \dots, n$ *using any one of the following methods :*

a) choose $\bar{x}_i \in \mathbb{X}$ *and solve* (6a)-(6d) *for* $i = 1, \dots, n$.
b) choose $\bar{x} \in \mathbb{X}$*, for example* $\bar{x} = [0 \dots 1 \dots 0]^{\mathrm{T}}$ *(*j*-th component, with* $j \in \{1, \dots, n_x\}$*), choose* $\bar{x}_i = c_i\bar{x}$ *with* $c_{1\dots n} \in \mathbb{R}_0^+$ *and* $c_i < c_{i+1}$ *and solve* (6a)-(6d) *for* $i = 1, \dots, n$.
c) calculate $a_{1\dots n_x} \in \mathbb{R}_0^+$ *as* $\max_{a_i} a_i$ *subject to the constraints of* (7) *and* (6b)-(6d)*, with* $x(0|0) = [0 \dots a_i \dots 0]^{\mathrm{T}}$ *(*i*-th component). Choose values* $c_{1\dots n} \in (0, 1)$ *with* $c_i < c_{i+1}$*. Calculate the* $X_i, Y_i, Z_i, \gamma_i, i = 1, \dots, n$ *as follows :*

$$\min_{\substack{\{u_j(k)\in\mathbb{R}^{n_u}\}_{j=1\dots n, k=0\dots n_h-1} \\ X,Y,Z,\gamma}} \gamma \quad (11a)$$

subject to ($\forall k = 0, \dots, n_h - 1, \forall j = 1, \dots, n$*) :*

$$x_j(k+1) = Ax_j(k) + Bu_j(k), \quad (11b)$$
$$u_j(k) \in \mathbb{U}, \quad (11c)$$
$$x_j(k) \in \mathbb{X}, \quad (11d)$$

subject to ($\forall j = 1, \dots, n$*) :*

$$x_j(n_h)^{\mathrm{T}} Z^{-1} x_j(n_h) \leq 1, \quad (11e)$$
$$x_j(0) = [0\ \dots\ c_i c_{\max} a_j\ \dots\ 0]^{\mathrm{T}} \quad (11f)$$

and subject to (6b)-(6d), *with* $c_{\max} \in (0,1)$ *assigned the largest value resulting in a feasible optimization problem* (11) *and* (6b)-(6d) *with* $c_i = 1$.

Method a) allows the user to choose the $\bar{x}_i$ freely, while method b) reduces the choice of the $\bar{x}_i$ to the choice of $n+1$ positive scalars, which is more transparant to the user than variant 1. Method c) is a refinement of b) and imposes that the terminal constraints have to be reachable in n_h time steps from n_x different initial states that have norms proportional to the c_i. Since in all three cases, the $X_i, Y_i, Z_i, \gamma_i, i = 1, \ldots, n$ satisfy (6b)-(6d), they are valid to be used in the on-line part of the algorithm :

Algorithm 2 ***[CC-MPC (on-line)] :*** *Given a set of terminal costs and constraints defined by* $X_i, Y_i, Z_i, \gamma_i, i = 1, \ldots, n$, *computed using Algorithm 1, solve at each time step* k, *given the current state* $x(k|k)$, *the following optimization problem*

$$\min_{\boldsymbol{u},\gamma,\lambda} \gamma + \sum_{i=1}^{n} \lambda_i \gamma_i \tag{12a}$$

subject to

$$\begin{bmatrix} \gamma - f^{\mathrm{T}}\boldsymbol{u} & \boldsymbol{u}^{\mathrm{T}} \\ \boldsymbol{u} & H^{-1} \end{bmatrix} \geq 0, \tag{12b}$$

$$A_{\mathrm{ineq}}\boldsymbol{u} \leq b_{\mathrm{ineq}}, \tag{12c}$$

$$\begin{bmatrix} 1 & (\Psi_{n_h}\boldsymbol{u} + \psi_{n_h})^{\mathrm{T}} \\ \Psi_{n_h}\boldsymbol{u} + \psi_{n_h} & \sum_{i=1}^{n} \lambda_i Z_i \end{bmatrix} \geq 0, \tag{12d}$$

$$\lambda_{1\ldots n} \geq 0, \tag{12e}$$

$$\sum_{i=1}^{n} \lambda_i \leq 1, \tag{12f}$$

and apply $u(k)$ *to the system.*

Compared to SD-MPC, only $n+1$ additional variables (the variables γ and λ_i) have to be added to the standard MPC optimization problem, instead of $(n-1)(n_x+1)$ additional variables (the variables $\hat{x}_i$ and λ_i). In the following sections we refer to this method as *MPC using convex combinations* or *CC-MPC*. Asymptotic stability of this method will be proven in Section 4. Two further refinements to this algorithm that at each time step only considers a subset of the total set of terminal constraints can be found in [11,12].

Remark 1 *It can easily be proven that the resulting terminal admissable set* $\mathbb{X}_{n_h}$ *is equal to the convex hull of the different terminal constraints defined by the* Z_i, *which is also the case for SD-MPC. The difference between the two methods will become clear in Section 4.*

Remark 2 *Using techniques from [9] both SD-MPC and CC-MPC can be reformulated as a* Second-Order Cone Program (SOCP), *which can be solved more efficiently. CC-MPC has the disadvantage that additional variables, similar to the terminal state components of SD-MPC, have to be introduced in order to construct the SOCP, but SD-MPC still has an additional* $n \cdot n_x$*-dimensional SOC constraint compared with CC-MPC. We refer to the extended internal report version of this paper [11] for more details.*

4 Stability proof

Theorem 1 *Consider a linear state-space model given by* $x(k+1) = Ax(k) + Bu(k)$, *state and input constraints* $x(k) \in \mathbb{X}$ *and* $u(k) \in \mathbb{U}$ *and positive definite weighting matrices* Q *and* R. *Furthermore, consider a parameterized set* $\kappa_\theta(\cdot) : \mathbb{R}^{n_x} \to \mathbb{R}^{n_u}, \mathbb{X}_{n_h,\theta} \subset \mathbb{X}$ *and* $F_\theta(\cdot) : \mathbb{R}^{n_x} \to \mathbb{R}$ *with parameter* θ, *that is defined over a user specified set* Θ *and assume that* $\forall \theta \in \Theta$ *stability conditions* (3a)-(3d) *are satisfied. The following MPC scheme, if feasible for* $k = 0$, *now ensures asymptotic stability of the closed-loop system if the time-varying set* $\Theta(k) \subset \Theta$, *chosen by the controller at each time step* k *before starting the optimization, satisfies* $\theta^{\mathrm{o}}(k-1) \in \Theta(k)$, *with* $\theta^{\mathrm{o}}(k-1)$ *the optimal value of* θ *at time step* $k-1$ *:*

$$\min_{\boldsymbol{x},\boldsymbol{u},\theta} J^{\star}_{n_h}(x(k), \boldsymbol{x}, \boldsymbol{u}, \theta) = \sum_{i=0}^{n_h-1} \|u(k+i|k)\|_R^2 + \sum_{i=0}^{n_h-1} \|x(k+i|k)\|_Q^2 + F_\theta(x(k+n_h|k)) \tag{13a}$$

subject to

$$x(k+i+1|k) = Ax(k+i|k) + Bu(k+i|k), \quad i = 0, \ldots, n_h-1, \tag{13b}$$

$$u(k+i|k) \in \mathbb{U}, \quad i = 0, \ldots, n_h-1, \tag{13c}$$

$$x(k+i|k) \in \mathbb{X}, \quad i = 1, \ldots, n_h-1, \tag{13d}$$

$$x(k+n_h|k) \in \mathbb{X}_{n_h,\theta}, \tag{13e}$$

$$\theta \in \Theta(k). \tag{13f}$$

PROOF. The main idea is to construct a feasible solution to the optimization problem at time step $k+1$ with associated cost $J^{\star,\mathrm{f}}_{n_h}(x(k+1)) \equiv J^{\star}_{n_h}(x(k+1), \mathbf{x}^{\mathrm{f}}(k+1), \mathbf{u}^{\mathrm{f}}(k+1), \theta^{\mathrm{f}}(k+1))$ using the optimal solution at time step k, with associated cost $J^{\star,\mathrm{o}}_{n_h}(x(k)) \equiv J^{\star}_{n_h}(x(k), \mathbf{x}^{\mathrm{o}}(k), \mathbf{u}^{\mathrm{o}}(k), \theta^{\mathrm{o}}(k))$. We will prove that $J^{\star,\mathrm{f}}_{n_h}(x(k+1)) < J^{\star,\mathrm{o}}_{n_h}(x(k)), \forall x(k) \neq 0$, which then leads to $J^{\star,\mathrm{o}}_{n_h}(x(k+1)) < J^{\star,\mathrm{o}}_{n_h}(x(k)), \forall x(k) \neq 0$ and implies that $J^{\star,\mathrm{o}}_{n_h}(x(k))$ is a Lyapunov function for the closed loop system, thus establishing asymptotic stability. The feasible solution to (13) is constructed as

		$n=3$		$n=9$		$n=17$	
	FTV-MPC	SD-MPC	CC-MPC	SD-MPC	CC-MPC	SD-MPC	CC-MPC
control cost for $b=2$	52.4	53.5	55.3	52.5	52.8	52.5	52.7
control cost for $b=4$	293.2	296.2	307.1	294.4	296.3	294.3	295.8
CPU-t. (LMI)	0.84	0.80	0.59	1.63	0.70	3.92	0.84
CPU-t. (SOCP)	/	0.16	0.16	0.18	0.17	0.53	0.17

Table 1
Control performance (expressed as total simulation control cost) and computational complexity (expressed as maximum CPU-time per iteration in seconds as measured on a P-4 2GHz PC using MATLAB 6.5, SEDUMI 1.05R5 [2] and MOSEK 3.0.1.18 [1]) of FTV-MPC [4], SD-MPC [3], CC-MPC and SCC-MPC for different initial state values $x(0) = [b\ 0\ 0\ 0]^{\mathrm{T}}$. All algorithms used a horizon of $n_h = 3$.

follows :

$$\mathbf{x}^{\mathrm{f}}(k+1) = \big[(x^{\mathrm{o}}(k+2|k))^{\mathrm{T}}\ \ldots\ (x(k+n_h|k)^{\mathrm{o}})^{\mathrm{T}},\ f(x(k+n_h|k)^{\mathrm{o}}, \kappa_{\theta^{\mathrm{o}}(k)}(x(k+n_h|k)^{\mathrm{o}}))^{\mathrm{T}}\big]^{\mathrm{T}} \quad (14a)$$

$$\mathbf{u}^{\mathrm{f}}(k+1) = \big[(u^{\mathrm{o}}(k+1|k))^{\mathrm{T}}\ \ldots\ (u(k+n_h-1|k)^{\mathrm{o}})^{\mathrm{T}},\ \kappa_{\theta^{\mathrm{o}}(k)}(x(k+n_h|k)^{\mathrm{o}})^{\mathrm{T}}\big]^{\mathrm{T}} \quad (14b)$$

$$\theta^{\mathrm{f}}(k+1) = \theta^{\mathrm{o}}(k). \quad (14c)$$

Due to the restriction that $\Theta(k)$ has to be chosen in such a way that $\theta^{\mathrm{o}}(k) \in \Theta(k+1)$, the above solution is indeed feasible to (13) if (3a)-(3c) hold, which is the case $\forall\theta \in \Theta(k) \subset \Theta$. (3d), which also holds $\forall\theta \in \Theta$, then establishes $J^{\star,\mathrm{o}}_{n_h}(x(k+1)) < J^{\star,\mathrm{o}}_{n_h}(x(k)), \forall x(k) \neq 0$, proving the theorem. □

This unifying theorem can be used to prove asymptotic stability of FTV-MPC, F-MPC, SD-MPC, CC-MPC and SCC-MPC, as follows :

- **FTV-MPC:** In case $\bar{x}$ is used as a parameter θ to parameterize the terminal cost and constraint, the terminal cost and constraint can be defined as $Q_{n_h,\theta} = \gamma Z^{-1}$ and $\mathbb{X}_{n_h,\theta} = \mathbb{X} \cap \{x | x^{\mathrm{T}} Z^{-1} x \leq 1\}$. When choosing the parameter admissible set as $\Theta = \{\bar{x} | (6b)\text{-}(6d) \text{ is feasible}\}$, it can be shown that $\bar{x}$ may be substituted with $x(k+n_h|k)$ without influencing the optimization results and the above theorem then reduces to the FTV-MPC algorithm as introduced in [4].
- **F-MPC:** F-MPC can be seen as a special case of FTV-MPC with $\Theta(k)$ defined as the singleton $\Theta(k) = \Theta = \{\bar{x}\}$. In this way stability of this method is trivially proven.
- **SD-MPC:** A slight reformulation also enables this method to fit into the above stability theorem. While the actual implementation uses both $\lambda_{i\ldots n}$ and $\hat{x}_{i\ldots n}$ as θ, in this reformulation we define $\theta = [\lambda_1; \ldots; \lambda_n]$ and define the terminal cost and constraint as $F_\theta(x) = \{\min_{\hat{x}_{1\ldots n}} \tilde{x}^{\mathrm{T}} V \tilde{x} \text{ s.t. } (8)\}$ and $\mathbb{X}_{n_h,\theta} = \{x(k+n_h|k) | \exists \hat{x}_{1\ldots n} \text{ satisfying } (8)\}$. This constraint can easily be verified to be equivalent to $\mathbb{X}_{n_h,\theta} = \{x | x^{\mathrm{T}} \left(\sum_{i=1}^n \lambda_i Z_i\right)^{-1} x \leq 1\}$. The admissible parameter set used in this method can be defined as $\Theta(k) = \Theta = \{\theta | \sum_i \lambda_i = 1\}$. The controller obtained in this way can be written as a linear state feedback controller in terms of the terminal state components $\hat{x}_{1\ldots n}$, and can be shown [3] to satisfy the stability conditions (3a)-(3d) for all $\theta \in \Theta$ in this generalized definition of the state, which validates the use of the above theorem to claim stability.
- **CC-MPC:** This method also corresponds to the choice $\theta = [\lambda_1; \ldots; \lambda_n]$, but uses as terminal cost and constraint $Q_{n_h,\theta} = \left(\sum_{i=1}^n \lambda_i \gamma_i\right) \left(\sum_{i=1}^n \lambda_i Z_i\right)^{-1}$ and $\mathbb{X}_{n_h,\theta} = \{x | x^{\mathrm{T}} \left(\sum_{i=1}^n \lambda_i Z_i\right)^{-1} x \leq 1\}$, while the parameter admissible set is defined as $\Theta(k) = \Theta = \{\lambda_{1\ldots n} | \lambda_{1\ldots n} \geq 0, \sum_i \lambda_i \leq 1\}$. It was shown in Section 3 that this terminal cost and constraint satisfies (3a)-(3d) for all $\theta \in \Theta$, and thus stability is guaranteed.

5 Example

This example deals with the control of an inverted pendulum, which consists of a cart that is driven by an electrical motor. A rod is connected to the cart by a joint that can rotate freely. The aim is to steer the cart to the desired position while keeping the rod in the upward position. The input of this system is the motor voltage V, the states are the cart position x_{cart}, the rod angle α and their time derivatives v_x and v_α. State and input constraints are defined by peak bounds $x_{\max} = [20\ 30\pi/180\ 10\ 10]^{\mathrm{T}}$ and $u_{\max} = 5$ with $x = [x_{\mathrm{cart}}\ \alpha\ v_x\ v_\alpha]^{\mathrm{T}}$ and $u = V$. The standard MPC algorithm with $n_h = 3, Q = \mathrm{diag}([1\ 0.001\ 0.001\ 0.001])$ and $R = 0.01$ resulted in unstable behaviour for the initial states shown in the examples. An FTV-MPC controller and SD-MPC and CC-MPC controllers with ellipsoidal terminal constraint sets computed with Algorithm 1.c ($n = 3, 9, 17$ and $c_i \in \{0.01, 0.02, 0.03, 0.04, 0.06, 0.07, 0.08, 0.1, 0.12, 0.14, 0.16, 0.18, 0.20, 0.22, 0.24, 0.26\}$) were also applied to the problem. Table 1 shows the results obtained when applying the different MPC algorithms to this system. FTV-MPC, SD-MPC and CC-MPC have very similar

performance, especially for larger values of n, but CC-MPC still has a lower computational complexity than the two other algorithms. Figure 1 shows that CC-MPC chooses a smaller terminal constraint and hence a more optimal terminal controller at each time step. Figure 2 shows the cart position and input voltage for $b = 2$ for FTV-MPC, CC-MPC, SD-MPC and F-MPC controllers. The first three algorithms have an almost identical control behaviour, that is superior compared to the F-MPC controller (computed with $\bar{x} = [2\ 0\ 0\ 0]^{\mathrm{T}}$). Note that an F-MPC controller with a smaller terminal constraint (and hence a locally more optimal terminal controller) resulted in on-line infeasibilities.

6 Conclusion

In this paper a new MPC algorithm with time varying-terminal cost was introduced and demonstrated. The algorithm uses convex combinations of a set of precalculated terminal costs and constraints to calculate the terminal cost and has efficient LMI and SOCP formulations. The scheme is proven to be stabilizing within a stability framework unifying MPC with fixed terminal cost and other MPC schemes with time-varying terminal cost. The algorithm has a reduced computational complexity compared to similar algorithms, while achieving similar control performance levels. Due to the use of ellipsoidal invariant sets, the method can be extended to deal with model uncertainty, which is the subject of future research.

Acknowledgements

Research supported by KUL: GOA-Mefisto 666, GOA-AmbioRics; FWO: G.0240.99, G.0407.02, G.0197.02, G.0141.03, G.0491.03, G.0120.03, G.0800.01, G.0452.04, G.0499.04, G.0211.05, G.0080.01, research communities (ICCoS, ANMMM); IWT: PhD Grants, BFSPO: IUAP P5/22; PODO-II (CP/40: TMS and Sustainability); EU: FP5-CAGE; FP5-Quprodis; ERNSI; FP6-BioPattern; Eureka 2419-FliTE; Contract Research/agreements: Data4s, Electrabel, Elia, LMS, IPCOS, VIB; Bert Pluymers is a research assistant with the I.W.T. Jeroen Buijs is a teaching assistant at the department of Energy of the Group T Leuven Hogeschool. Dr. Johan Suykens is an associate professor and Dr. Bart De Moor is a full professor at the Katholieke Universiteit Leuven, Belgium

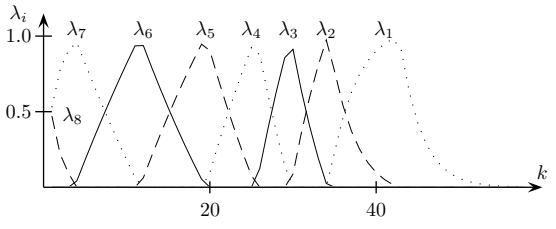


Fig. 1. λ_i values resulting from CC-MPC (with $n = 9$) for the simulation with $b = 4$ (see table 1). The controller chooses a smaller terminal constraint as the state approaches the origin.

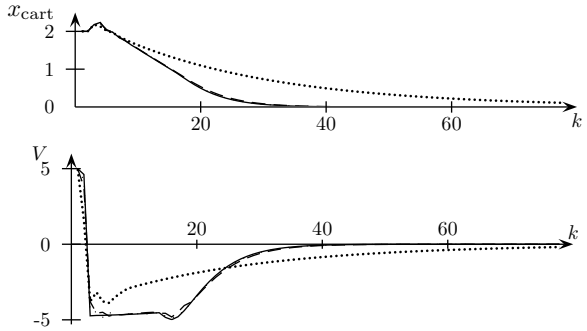


Fig. 2. Cart position (top) and input voltage (bottom) trajectories for FTV-MPC (solid), SD-MPC (dotted, $n = 17$), CC-MPC (dashed, $n = 17$) and MPC with fixed terminal cost (thick dotted, total simulation control cost : 71.7) for $a = 2$ (see Table 1). All algorithms used a horizon of $n_h = 3$.

References

[1] Mosek 3.0 MATLAB optimization toolbox user's manual. http://www.mosek.com/documentation.html.

[2] Sedumi 1.05R5 MATLAB SDP optimization toolbox. http://fewcal.kub.nl/sturm/software/sedumi.html.

[3] M. Bacic, M. Cannon, Y. I. Lee, and B. Kouvaritakis. General interpolation in MPC and its advantages. *IEEE Transactions on Automatic Control*, 48(6):1092–1096, June 2003.

[4] H. H. J. Bloemen, T. J. J. van den Boom, and H. B. Verbruggen. Optimizing the end-point state-weighting matrix in model-based predictive control. *Automatica*, 38:1061–1068, 2002.

[5] H. Chen and F. Allgöwer. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*, 34:1205–1217, 1998.

[6] K. B. Kim. Implementation of stabilizing receding horizon controls for time-varying systems. *Automatica*, 38:1705–1711, 2002.

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

[8] J.-W. Lee, W. H. Kwon, and J. Choi. On stability of constrained receding horizon control with finite terminal weighting matrix. *Automatica*, 34:1607–1612, 1998.

[9] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret. Applications of second-order cone programming. *Linear Algebra and Its Applications*, 284:193–228, 1998.

[10] 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.

[11] B. Pluymers, L. Roobrouck, J. Buijs, J. A. K. Suykens, and B. De Moor. Model-predictive control with time-varying terminal cost using convex combinations. *Internal Report 04-028, ESAT-SISTA, K. U. Leuven (Leuven, Belgium), http://www.esat.kuleuven.ac.be/sista/.*

[12] B. Pluymers, J. A. K. Suykens, and B. De Moor. Linear model predictive control with time-varying terminal cost using sparse convex combinations and bisection searching. *Internal Report 04-27, ESAT-SISTA, K. U. Leuven (Leuven, Belgium), Submitted for publication*, 2004.

[13] Z. Wan and M. V. Kothare. Efficient robust constrained model predictive control with a time varying terminal constraint set. *Syst. and Contr. Letters*, 48:375–383, 2003.