OPTIMAL CONTROL APPLICATIONS AND METHODS
Optim. Control Appl. Meth. (2015)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/oca.2188

MPC-based tracking for real-time systems subject to time-varying polytopic constraints

T. Manrique[1], M. Fiacchini[2], T. Chambrion[3,4] and G. Millerioux[1,*,†]

[1]*Université de Lorraine, Centre de Recherche en Automatique de Nancy, Vandœuvre-lès-Nancy, F-54506, France*
[2]*Gipsa-lab, Grenoble Campus, Saint-Martin-d'Hères, F-38402, France*
[3]*INRIA, Villers-lès-Nancy, F-54600, France*
[4]*Université de Lorraine, Institut Élie Cartan de Lorraine, Vandœuvre-lès-Nancy, F-54506, France*

SUMMARY

This paper presents a real-time MPC-based tracking strategy for linear systems subject to time-varying constraints. The framework is quite general because the time-varying constraints can apply both to the state and to the input. To handle the problem, a polytopic invariant set computed off-line is homogeneously dilated or contracted on-line to fit the polytopic time-varying constraints. The invariant set is used as an admissible terminal constraint set so that it guarantees stability and convergence in the tracking task. The on-line cost of the homothetic invariant set computation is low enough to cope with systems subject to stringent real-time constraints. Copyright © 2015 John Wiley & Sons, Ltd.

Received 20 February 2014; Revised 7 February 2015; Accepted 16 June 2015

KEY WORDS: model predictive control; time-varying constraints; invariant sets; real-time application

1. INTRODUCTION

Model predictive control (MPC) has been well-admitted, for the past 50 years, as a suitable solution to deal with multi-variable constrained control problems and also to deal with robustness in presence of model uncertainties and noise [1, 2]. Feasibility of the optimization problem and stability of the control law have been major issues. Regarding the feasibility, the inherent trade-off between finite-horizon and constraints of MPC-based techniques raises challenging problems [2, 3]. In particular, when it comes to MPC-based tracking, such a trade-off may prevent the reachability of the reference due to loss of feasibility of the MPC problem [4]. As far as stability is concerned, several approaches have been proposed to provide some guarantees. The consideration of invariant sets as terminal constraint is one of the most popular. Indeed, the strategies based on invariant sets allow to guarantee convergence towards the origin by implicitly extending the prediction horizon to infinity without any substantial increasing of the on-line computational cost. The computation of the invariant sets, which are usually polytopes or ellipsoids, is performed off-line [5–7].

Despite the effervescence of works dealing with MPC, only few ones address the problems under time-varying constraints. As some exceptions, we can mention the following contributions. Time-varying constraints that apply to the input and the output have been incorporated in the approach proposed by Bemporad *et al.* [8, 9], known under the name *explicit model predictive control*. However, the time-varying constraints are required to be fully known *a priori*. The controlling sequences must be computed off-line for all the possible variations of the constraints. Hence, the implementation of the MPC requires a massive storage capacity. Time-varying constraints have also been

*Correspondence to: G. Millerioux, Centre de Recherche en Automatique de Nancy, École Supérieure des Sciences et Technologies de l'Ingénieur de Nancy, 2 Rue Jean Lamour, Vandœuvre-lès-Nancy, F-54519, France.
†E-mail: gilles.millerioux@univ-lorraine.fr

considered in the work [10] wherein an extension of the *output feedback model predictive control* is proposed to cope with the time-varying case. This approach is based on a stable state estimator and a robust tube-based MPC which is solved on-line, unlike the previous aforementioned technique. However, the existence and the knowledge of a common invariant set satisfying the states and inputs constraints for all the time instants is required. More recently, Wada *et al.* [11] addressed the feasibility and stability issues related to an MPC-based tracking problem for LTI systems with time-varying constraints, more specifically time-varying input saturation levels. In [11], on-line LMIs optimization problems are involved in the design of a *gain-scheduled feedback control law*, that is, a feedback gain that depends on the time-varying saturation level of the input. However, solving on-line the LMIs may be unaffordable for real-time applications. Finally, let us stress that some well-admitted MPC-based tracking formulations can be very effective for time-invariant constraints, but cannot be suited to cope with time-varying ones. It is precisely the case for the MPC-based tracking strategy presented in [4], which benefits from a larger domain of attraction than the standard MPC formulation, but time-varying constraints are not allowed.

All in all, the consideration of time-varying constraints is still challenging when we are concerned with efficient MPC approaches compatible with real-time applications. This paper contributes to go further regarding such an issue. The objective is two-fold. It aims at providing a solution to the tracking problem for LTI systems subject to time-varying constraints. Besides, it aims at designing a control with low computational cost to allow for real-time applications. To enforce stability, the standard MPC-based tracking strategy [1, 3, 12] is used with the invariant terminal set constraint approach. The problem of the standard approach lies in that the computation of the invariant set becomes intricate when considering time-varying constraints. Indeed, if the constraints change in time, the invariant set is also time-varying and, so, must be recomputed on-line. And yet, when real-time applications are sought, it is far from being computationally affordable.

As a clue to tackle this problem, we propose in this paper an approach based on a homothetic transformation which consists in a contraction/dilatation of a predefined invariant set previously computed off-line. Such a transformation allows to fit on-line the time-varying constraints. The invariant set and the time-varying constraints are both admitting polytopic representations. The resulting parameter-dependent invariant set is an admissible terminal constraint for the MPC problem and guarantees asymptotic stability. Because the homothetic transformation is merely characterized by the dilating/contracting factor required to fit the constraints, the computational cost for the on-line procedure boils down to the calculation of such a factor, and thus, the MPC becomes suitable for real-time applications despite the time-varying constraints. Let us note that a very particular case of this approach has been presented in [13] for an application to low-consumption vehicles.

This paper is organized as follows. In Section 2, preliminaries on MPC tracking are recalled for LTI systems. In Section 3, the time-varying MPC algorithm is detailed and the conditions to guarantee stability are given. In Section 4, a numerical example is given to clarify the technical parts. In Section 5, the MPC tracking under time-varying constraints is developed for a real-time application involving an electric vehicle. Finally, Section 6 draws conclusions and proposes perspectives.

2. TRACKING MODEL PREDICTIVE CONTROL

2.1. Problem statement

Before addressing the time-varying case, let us recall some background on the issue of MPC-based tracking for LTI discrete-time systems under time-invariant constraints on the input and the state. Consider the linear system with state-space description

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

where $x(k) \in \mathbb{R}^n$ is the state, $u(k) \in \mathbb{R}^m$ is the control input, $\mathbf{A} \in \mathbb{R}^{n\times n}$ is the dynamical matrix and $\mathbf{B} \in \mathbb{R}^{m\times n}$ is the input matrix. The pair $(\mathbf{A}, \mathbf{B})$ is assumed to be stabilizable, that is, $\exists K$ s.t. $(\mathbf{A} + \mathbf{B}K)$ is Schur stable. The state is assumed to be accessible, that is, $x(k)$ is fully known at each time k. The system (1) is assumed to be subject to constraints on the input and the state, that is,

DOI: 10.1002/oca

$$
\begin{aligned}
x(k) &\in X,\ \forall k \geqslant 0, \\
u(k) &\in U,\ \forall k \geqslant 0,
\end{aligned} \tag{2}
$$

with X a convex and closed subset of $\mathbb{R}^n$ and U a convex and compact subset of $\mathbb{R}^m$.

Define the tracking errors $\Delta x(k) \in \mathbb{R}^n = x(k) - \bar{x}$ and $\Delta u(k) \in \mathbb{R}^m = u(k) - \bar{u}$ around the steady-state targets $\bar{x} \in \mathbb{R}^n$ and $\bar{u} \in \mathbb{R}^m$ ($\bar{x}$ and $\bar{u}$ are solutions of (1), i.e. $\bar{x} = \mathbf{A}\bar{x} + \mathbf{B}\bar{u}$). The dynamics of the tracking error is given by

$$
\Delta x(k+1) = \mathbf{A}\Delta x(k) + \mathbf{B}\Delta u(k), \tag{3}
$$

with tracking constraints defined $\forall k \geqslant 0$ as

$$
\begin{aligned}
\Delta x(k) &\in X_\Delta, \\
\Delta u(k) &\in U_\Delta,
\end{aligned} \tag{4}
$$

where X_Δ is a convex and closed subset of $\mathbb{R}^n$ and U_Δ is a convex and compact subset of $\mathbb{R}^m$. Both subsets are assumed to be polytopic sets containing the origin in their interior, that is, $0 \in int(X_\Delta)$ and $0 \in int(U_\Delta)$, hence $\bar{x} \in int(X)$ and $\bar{u} \in int(U)$ ($int(\cdot)$ denotes the interior of the set. The polytope X_Δ is described by

$$
X_\Delta = \{\Delta x \in \mathbb{R}^n : \mathbf{H}_{X_\Delta}\Delta x \leqslant \bar{1}\}, \tag{5}
$$

which is the hyperplane representation of the polytope X_Δ, with $\mathbf{H}_{X_\Delta} \in \mathbb{R}^{p_{X_\Delta} \times n}$ and $\bar{1} \in \mathbb{R}^{p_{X_\Delta}}$ is the ones vector $\bar{1} = [1, \ldots, 1]^T$. The scalar p_{X_Δ} is the number of facets of the polytope X_Δ and corresponds to the number of rows in $\mathbf{H}_{X_\Delta}$. The polytope U_Δ is also represented by a hyperplane representation as

$$
U_\Delta = \{\Delta u \in \mathbb{R}^m : \mathbf{H}_{U_\Delta}\Delta u \leqslant \bar{1}\}, \tag{6}
$$

with $\mathbf{H}_{U_\Delta} \in \mathbb{R}^{p_{U_\Delta} \times m}$ and $\bar{1} \in \mathbb{R}^{p_{U_\Delta}}$. The scalar p_{U_Δ} is the facets number of the polytope U_Δ. Let us note that polytopic representations are very handy to deal with linear constraints often encountered in practical applications [7].

The aim of the control law is to reach the steady-state targets $\bar{x}$ and $\bar{u}$ while fulfilling the constraints (4). That amounts to solving on-line the following open-loop problem:

$$
\begin{aligned}
\min_{\Delta u(k),\ldots,\Delta u(k+N_p-1)} & \sum_{i=0}^{N_p-1} \left(\Delta x^T(k+i)\mathbf{Q}\Delta x(k+i) + \Delta u^T(k+i)\mathbf{R}\Delta u(k+i)\right) \\
& \qquad\qquad + \Delta x^T(k+N_p)\mathbf{P}\Delta x(k+N_p), \\
s.t. \quad & \Delta x(k+i+1) = \mathbf{A}\Delta x(k+i) + \mathbf{B}\Delta u(k+i), \qquad \forall i = 0, \ldots, N_p - 1, \\
& \Delta x(k+i) \in X_\Delta, \qquad \forall i = 0, \ldots, N_p - 1, \\
& \Delta u(k+i) \in U_\Delta, \qquad \forall i = 0, \ldots, N_p - 1, \\
& \Delta x(k+N_p) \in X_f,
\end{aligned} \tag{7}
$$

with N_p as the prediction horizon and X_f the terminal (closed) set of feasible final states. The positive semi-definite weighting matrix $\mathbf{Q} \in \mathbb{R}^{n \times n}$ and the positive definite weighting matrix $\mathbf{R} \in \mathbb{R}^{m \times m}$ define the state and the input tracking costs, respectively. The matrix $\mathbf{P} \in \mathbb{R}^{n \times n}$ defines the terminal cost.

From [1–3], it is well known that to enforce stability and convergence towards the origin, it is sufficient to set the terminal cost positive definite matrix $\mathbf{P}$ as the solution of the Riccati equation that solves the infinite-horizon LQR problem for the system (1), with weighting matrices $\mathbf{Q}$ and $\mathbf{R}$. In addition, the tracking terminal set X_f must be designed to be a controlled invariant set in the neighbourhood of the origin, under the Linear-quadratic regulator (LQR) feedback gain denoted here with K. Indeed, the constraint on the terminal set ensures that, after N_p steps, the predicted state reaches the terminal set. Because such a set is invariant, if the infinite horizon LQR control

DOI: 10.1002/oca

gain is applied, the convergence towards the origin is guaranteed. Actually, the MPC consists in delivering, at each time k, the input $\Delta u(k)$, that is, the first sample of the optimal input sequence resulting from the solution of (7). At time $k+1$, a new open-loop optimal control problem is solved. The following subsection recalls some background on the design of the terminal invariant set X_f.

2.2. *Invariant terminal set design*

The following procedure corresponds to the standard procedure (e.g. [7]) to design the maximal invariant set but is particularized for the tracking problem (7). This procedure will become handy in the perspective of the tracking with time-varying constraints as it will be detailed in Section 3.

Let us define the closed-loop system

$$\Delta x(k+1) = (\mathbf{A} + \mathbf{B}K)\Delta x(k) \tag{8}$$

obtained from (3) and the stabilizing feedback control law $\Delta u = K\Delta x$. The gain K is derived from $\mathbf{P}$, that is, from the solution of the Riccati equation related to the infinite-horizon stabilization problem for system (3) with the quadratic cost weighting matrices $\mathbf{R}$ and $\mathbf{Q}$ involved in (7).

Let us now define the closed-loop constraints for (8) as

$$\mathcal{X}_\Delta = \{\Delta x \in \mathbb{R}^n : \Delta x \in X_\Delta,\ K\Delta x \in U_\Delta\}, \tag{9}$$

where X_Δ and U_Δ are given as in Subsection 2.1. The constraints (9) can be equivalently rewritten as

$$\mathcal{X}_\Delta = X_\Delta \cap X_{\Delta u}(U_\Delta), \tag{10}$$

where

$$X_{\Delta u}(U_\Delta) = \{\Delta x \in \mathbb{R}^n : K\Delta x \in U_\Delta\}. \tag{11}$$

The tracking constraints X_Δ and U_Δ in (4) are assumed to be polytopic. Hence, the closed-loop constraints (9) can be expressed as a convex polytopic set containing the origin in its interior. As a result, $\mathcal{X}_\Delta$ can be described by a hyperplane representation with complexity-index $p_{\mathcal{X}} \in \mathbb{N}$ (i.e. $p_{\mathcal{X}}$ is the number of facets of the polytope in the hyperplane representation):

$$\mathcal{X}_\Delta = \{\Delta x \in \mathbb{R}^n : \mathbf{H}_{\mathcal{X}_\Delta}\Delta x \leqslant \bar{1}\}, \tag{12}$$

where

$$\mathbf{H}_{\mathcal{X}_\Delta} = \begin{bmatrix} \mathbf{H}_{X_\Delta} \\ \mathbf{H}_{U_\Delta} K \end{bmatrix}, \tag{13}$$

with $\mathbf{H}_{\mathcal{X}_\Delta} \in \mathbb{R}^{p_{\mathcal{X}} \times n}$ and $\bar{1} \in \mathbb{R}^{p_{\mathcal{X}}}$ being the ones vector $\bar{1} = [1, \dots, 1]^T$. The matrices $\mathbf{H}_{X_\Delta} \in \mathbb{R}^{p_{X_\Delta} \times n}$ and $\mathbf{H}_{U_\Delta} \in \mathbb{R}^{p_{U_\Delta} \times m}$ are the matrices of the hyperplane representation of the tracking constraints X_Δ and U_Δ, respectively, with p_{X_Δ} and p_{U_Δ} as the number of facets in each one.

Now, let us recall the definition of an invariant set for the LTI discrete-time system (8). The definition is borrowed from the general definition given in [14] or [7] for example.

Definition 1

A closed and convex set $\Omega \subseteq \mathbb{R}^n$ with $0 \in int(\Omega)$ (i.e. Ω is a C-set) is said to be a positively invariant set for the system (8) under (tracking) constraints $\mathcal{X}_\Delta$ given in (9), if for all $\Delta x \in \Omega$ then $(\mathbf{A} + \mathbf{B}K)\Delta x \in \Omega$, being $\Omega \subseteq \mathcal{X}_\Delta$.

Otherwise stated, for all $\Delta x(k) \in \Omega$, that is, for any state $\Delta x(k)$ which has reached Ω, it holds that $\Delta x(k+i) \in \Omega$ for all $i \geqslant 0$ or equivalently that the state can no longer escape from Ω. The notion of *positively* refers to the fact that only the future states satisfy $x(k+i) \in \Omega$, that is, to $i \geqslant 0$. Hereafter, the positively invariant sets will be merely called *invariant sets* for brevity, because only the positive invariance will be considered.

DOI: 10.1002/oca

Because $\mathcal{X}_\Delta$ is a bounded polytope with the origin in its interior and $\mathbf{A} + \mathbf{B}K$ is Schur stable, then the maximal invariant set Ω in $\mathcal{X}_\Delta$ is also a convex polytope [5]. For convenience, the set Ω is considered to be described by its vertex representation

$$\Omega = \left\{ \Delta x \in \mathbb{R}^n : \Delta x = \mathbf{V}_\Omega \mathbf{w}, \sum_{i=1}^{p_\Omega} \mathbf{w}(i) = 1, \mathbf{w}(i) \geqslant 0, \forall i \right\}, \tag{14}$$

where $\mathbf{V}_\Omega \in \mathbb{R}^{n \times p_\Omega}$ is the matrix whose columns are the vertices of Ω, $\mathbf{w} \in \mathbb{R}^{p_\Omega}$, and p_Ω is the complexity index (number of vertices in the vertex representation) of the polytopic invariant set Ω. Each i-th column in $\mathbf{V}_\Omega$ is the i-th vertex of the polytope Ω.

To determine Ω, we can resort to the *backward iterative algorithm* (see [7, 14] for further details). It is an iterative procedure described by

$$\mathcal{X}_{\Delta_{-k-1}} = \{\Delta x \in \mathcal{X}_\Delta : (\mathbf{A} + \mathbf{B}K)\Delta x \in \mathcal{X}_{\Delta_{-k}}\}. \tag{15}$$

In other words, the pre-images of $\mathcal{X}_\Delta$ obtained from the inverse dynamics of (8) are successively backward computed and trimmed to get the largest polytopic invariant set included in $\mathcal{X}_\Delta$. To clarify such an algorithm, let us consider the following example.

Example 1
Consider the system $x(k+1) = \mathbf{A}x(k) + \mathbf{B}u(k)$ with

$$\mathbf{A} = \begin{bmatrix} 0.9 & 0.25 \\ -0.25 & 0.9 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} 0.5 \\ 2 \end{bmatrix}, \tag{16}$$

and tracking constraints X_Δ and U_Δ given by

$$\mathbf{H}_{X_\Delta} = \begin{bmatrix} 6.6667 & 0 \\ 0 & 20 \\ -5 & 0 \\ 0 & -12.5 \end{bmatrix}, \mathbf{H}_{U_\Delta} = \begin{bmatrix} 100 \\ -100 \end{bmatrix}. \tag{17}$$

The constraints X_Δ corresponds to $-0.2 \leqslant \Delta x_1 \leqslant 0.15$ and $-0.08 \leqslant \Delta x_2 \leqslant 0.05$, and the constraints U_Δ to $-0.01 \leqslant \Delta u \leqslant 0.01$. The weighting matrices of the MPC problem (7) are $\mathbf{Q} = [1\,0; 0\,1]$ and $\mathbf{R} = 30$. The solution of the infinite-horizon problem is $\mathbf{P} = [4.6534, 0.5613; 0.5613, 3.0237]$ and the corresponding gain is $K = [-0.0343, -0.1478]$. The matrix $\mathbf{H}_{\mathcal{X}_\Delta}$ defined as in (13), which characterizes the closed-loop system constraints $\mathcal{X}_\Delta$ given by (12), reads

$$\mathbf{H}_{\mathcal{X}_\Delta} = \begin{bmatrix} 6.6667 & 0 \\ 0 & 20 \\ -5 & 0 \\ 0 & -12.50 \\ -3.4304 & -14.7758 \\ 3.4304 & 14.7758 \end{bmatrix}, \tag{18}$$

with complexity-index $p_\mathcal{X} = 6$. The polytopic constraints X_Δ and $\mathcal{X}_\Delta$ are depicted on Figure 1. The figure clearly shows that $\mathcal{X}_\Delta$ is a subset of X_Δ, and it is in accordance with (10). The successive polytopes $\mathcal{X}_{\Delta_{-k-1}}$ are depicted in grey dotted lines in Figure 2. The largest invariant set is found after five steps and is portrayed in solid line in Figure 2. The resulting invariant set Ω is given by the following vertex representation, the complexity-index being $p_\Omega = 12$:

DOI: 10.1002/oca

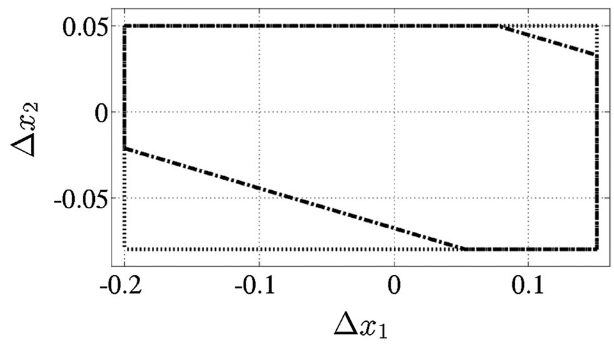

Figure 1. Dashed line, constraints X_Δ; dash-dotted line, constraints $\mathcal{X}_\Delta$.

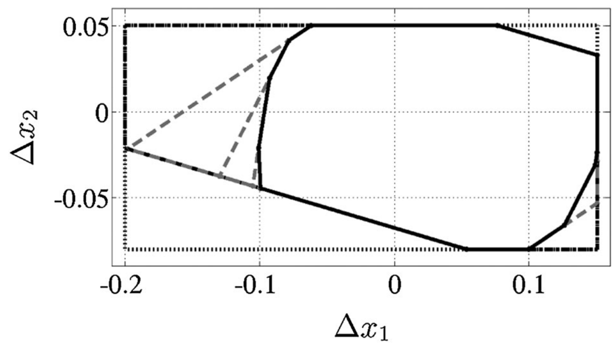

Figure 2. Grey dashed lines, pre-images of $\mathcal{X}_\Delta$; solid line, final invariant set Ω.

$$
\mathbf{V}_\Omega = \begin{bmatrix} 0.1500 & 0.0329 \\ 0.0993 & -0.0800 \\ -0.0928 & 0.0194 \\ 0.0531 & -0.0800 \\ 0.0761 & 0.0500 \\ -0.0621 & 0.0500 \\ -0.0785 & 0.0413 \\ -0.0996 & -0.0446 \\ -0.1010 & -0.0211 \\ 0.1257 & -0.0661 \\ 0.1485 & -0.0311 \\ 0.1500 & -0.0238 \end{bmatrix}^T . \tag{19}
$$

The calculation of the invariant set can be computationally demanding. However, for time-invariant constraints, the invariant set is computed only once and off-line [5–7]. On the other hand, if the constraints vary in time, the invariant terminal set must be recomputed on-line to solve the MPC problem (7). Let us stress that making such sets to be time varying can be a solution to cope, for example, with feasibility purposes. Indeed, the larger the sets, the greater the feasibility region. Nevertheless, the on-line recomputing of the invariant set is often restrictive if hard real-time constraints must be faced. The aim of the next section is to propose a solution to handle such a problem.

DOI: 10.1002/oca

3. TIME-VARYING TRACKING MODEL PREDICTIVE CONTROL

3.1. Problem statement

Let us consider the MPC-based optimal tracking problem (7) for the system (1) with tracking time-varying constraints verifying $\forall k \geqslant 0$

$$\begin{aligned} \Delta x(k) &\in X_\Delta(k) \subseteq \mathbb{R}^n, \\ \Delta u(k) &\in U_\Delta(k) \subseteq \mathbb{R}^m. \end{aligned} \tag{20}$$

The notation (k) reflects that the control input and the actual state are constrained within sets which might be changing in time.

The objective is to steer the tracking error to the origin while fulfilling the constraints (20) with an MPC-based strategy. The open-loop problem (7) turns into

$$\begin{aligned} \min_{\Delta u(k),\ldots,\Delta u(k+N_p-1)} \sum_{i=0}^{N_p-1} &\left(\Delta x^T(k+i)\mathbf{Q}\Delta x(k+i) + \Delta u^T(k+i)\mathbf{R}\Delta u(k+i)\right) \\ &+ \Delta x^T(k+N_p)\mathbf{P}\Delta x(k+N_p), \\ s.t. \quad & \Delta x(k+i+1) = \mathbf{A}\Delta x(k+i) + \mathbf{B}\Delta u(k+i), \quad \forall i = 0,\ldots,N_p-1, \\ & \Delta x(k+i) \in X_\Delta(k), \quad \forall i = 0,\ldots,N_p-1, \\ & \Delta u(k+i) \in U_\Delta(k), \quad \forall i = 0,\ldots,N_p-1, \\ & \Delta x(k+N_p) \in X_f(k), \end{aligned} \tag{21}$$

with N_p, $\mathbf{Q}$, $\mathbf{R}$ and $\mathbf{P}$ defined as in Subsection 2.1.

To ensure stability, convergence of the tracking error to zero and recursive constraints satisfaction, we propose to define the time-varying terminal (closed) set $X_f(k)$ of feasible final states as a parameter-dependent invariant set for the system (8) under time-varying constraints (20). It will be denoted hereafter $\bar{\Omega}(k)$. Having in mind the real-time efficiency as a main priority, the point is that simplicity of implementation must be preserved. And yet, as previously shown, obtaining an invariant set is computationally demanding. Let us note that, in particular, the complexity index of the polytopic constraints may also vary in time. A method to compute $\bar{\Omega}(k)$ is proposed in the subsequent discussion and benefits from an ease of real-time implementation while preserving the stability and convergence properties.

Following (9) for the time-invariant case, the input and the state have to satisfy $\Delta u = K\Delta x \in U_\Delta(k)$ and $\Delta x \in X_\Delta(k)$. This is equivalent to $\Delta x \in \mathcal{X}_\Delta(k)$ with

$$\mathcal{X}_\Delta(k) = \{\Delta x \in \mathbb{R}^n : \Delta x \in X_\Delta(k),\ K\Delta x \in U_\Delta(k)\} = X_\Delta(k) \cap X_{\Delta u}(U_\Delta(k)), \tag{22}$$

where $X_{\Delta u}(U_\Delta(k))$, similarly to (11), is given by

$$X_{\Delta u}(U_\Delta(k)) = \{\Delta x \in \mathbb{R}^n : K\Delta x \in U_\Delta(k)\}. \tag{23}$$

The following assumptions will be considered in the sequel.

Assumption 1
The steady-state targets $\bar{x}(k)$ and $\bar{u}(k)$ are assumed to be available on-line. Furthermore, the targets $\bar{x}(k)$ and $\bar{u}(k)$ are considered to remain constant within the prediction horizon N_p, that is, $\bar{x}(k+i) = \bar{x}(k)$ and $\bar{u}(k+i) = \bar{u}(k)$ for $i = 0,\ldots,N_p$.

Remark 1
The assumptions made on the available knowledge of the future constraints can affect substantially the feasibility of the solution. This work does not focus on the problem of ensuring recursive feasibility. To our opinion, this problem is not solvable for the general case with partial information of the future behaviour of the constraints. Thereby, we pose the following assumption (Assumption

DOI: 10.1002/oca

2) on the on-line available knowledge. The results could be easily adapted to the cases of different assumptions on the knowledge of future constraints. Further research efforts might be directed to the problem of characterizing the conditions under which feasibility can be guaranteed.

Assumption 2
The time-varying constraints sets $X_\Delta(k+i)$ and $U_\Delta(k+i)$ are available on-line for $i = 0, \dots, N_p$ and are considered to remain constant within the prediction horizon N_p.

Assumption 3
The time-varying constraints sets $X_\Delta(k)$ and $U_\Delta(k)$ are assumed to be convex and compact polytopes that contain the origin in their interior, for all k. $X_\Delta(k)$ and $U_\Delta(k)$ are represented by hyperplane representations given by $\mathbf{H}_{X_\Delta}(k)$ and $\mathbf{H}_{U_\Delta}(k)$, respectively.

As a result, similarly to the time-invariant constraints case, the polytopic constraints $\mathcal{X}_\Delta(k)$ are represented by the following hyperplane representation

$$\mathcal{X}_\Delta(k) = \{\Delta x \in \mathbb{R}^n : \mathbf{H}_{\mathcal{X}_\Delta}(k)\Delta x \leqslant \bar{1}\}, \tag{24}$$

where $\mathbf{H}_{\mathcal{X}_\Delta}(k) \in \mathbb{R}^{p_{\mathcal{X}}(k)\times n}$, $\bar{1} \in \mathbb{R}^{p_{\mathcal{X}}(k)}$, $p_{\mathcal{X}}(k)$ the complexity index of the polytopic constraints $\mathcal{X}_\Delta(k)$ at time k and

$$\mathbf{H}_{\mathcal{X}_\Delta}(k) = \begin{bmatrix} \mathbf{H}_{X_\Delta}(k) \\ \mathbf{H}_{U_\Delta}(k)K \end{bmatrix}. \tag{25}$$

3.2. Homothetic transformation of the invariant set

As a clue to tackle the aforementioned problem, we are given off-line a set $\hat{\Omega} \subseteq \mathbb{R}^n$ defined as an invariant set for the system (8) and admitting a polytopic description. The main idea consists in defining an on-line homothetic transformation, centred in the origin, to obtain a homothetic copy of the invariant set $\hat{\Omega}$ at each time k. This homothetic transformation is characterized by a factor $\alpha(k) \in \mathbb{R}$, such as the resulting convex invariant set $\bar{\Omega}(k) = \alpha(k)\hat{\Omega} \subseteq \mathbb{R}^n$ is an invariant set for the system (8) under convex polytopic constraints $\mathcal{X}_\Delta(k)$. Let us stress that, because $\mathcal{X}_\Delta(k)$ and $\hat{\Omega}$ are convex sets, the homothetic transformation always exists.

Remark 2
The stability does not depend on the feature of the invariant set $\hat{\Omega}$. It holds for any invariant set for the system (1). The feature can be chosen to meet specific characteristics all along the homothetic transformations. In practice, the choice can be made according to some heuristics specifying whether the invariant set should be centred or not around the origin, symmetric or not, admitting a prescribed number of vertices and so on. The invariant set $\hat{\Omega}$ can be designed from nominal constraints $\hat{X}_\Delta$ and $\hat{U}_\Delta$ conveniently chosen according to those specifications.

Similarly to the vertex representation (14), the set $\hat{\Omega}$ is given by

$$\hat{\Omega} = \left\{ \Delta x \in \mathbb{R}^n : \Delta x = \mathbf{V}_{\hat{\Omega}}\mathbf{w},\ \sum_{i=1}^{p_{\hat{\Omega}}} \mathbf{w}(i) = 1,\ \mathbf{w}(i) \geqslant 0,\ \forall i \right\}, \tag{26}$$

where $\mathbf{V}_{\hat{\Omega}} \in \mathbb{R}^{n\times p_{\hat{\Omega}}}$ is the vertices array

$$\mathbf{V}_{\hat{\Omega}} = \left[v_1\, v_2\, \dots\, v_{p_{\hat{\Omega}}} \right], \tag{27}$$

and each column $v_j \in \mathbb{R}^n$, $j = 1, \dots, p_{\hat{\Omega}}$, in $\mathbf{V}_{\hat{\Omega}}$ is the j-th vertex of the polytope $\hat{\Omega}$. Besides, $\mathbf{w} \in \mathbb{R}^{p_{\hat{\Omega}}}$, and $p_{\hat{\Omega}}$ is the complexity index (number of vertices in the polytope) of the invariant set $\hat{\Omega}$.

Consider $\mathcal{P}_i(k)$ as the i-th hyperplane given by the i-th facet of $\mathcal{X}_\Delta(k)$ in (24), that is,

$$\mathcal{P}_i(k) = \{\Delta x \in \mathbb{R}^n : H_i(k)\Delta x(k) = 1\}, \tag{28}$$

DOI: 10.1002/oca

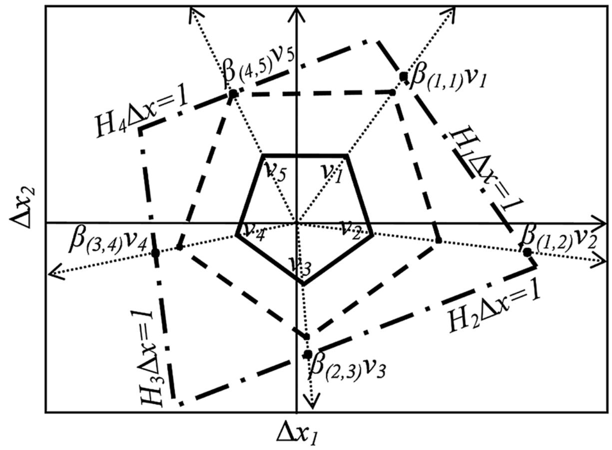

Figure 3. Solid line, polytopic invariant set $\hat{\Omega}$; dashed line, homothetic copy of $\hat{\Omega}$ at time k; dash-dotted line, polytopic constraints $\mathcal{X}_\Delta$ at time k; $\beta_{(i,j)}v_j$, homothetic vector of v_j (along the dotted vectors) with factor $\beta_{(i,j)}$ at time k (time index k not reported).

where $H_i(k) \in \mathbb{R}^{1\times n}$ is the i-th row in $\mathbf{H}_{\mathcal{X}_\Delta}(k)$, with $i = 1,\ldots,p_{\mathcal{X}}(k)$. Then, given a vertex $v_j \in \mathbb{R}^n$ $(j = 1,\cdots,p_{\hat{\Omega}})$ in (27), such that the scalar $H_i(k)v_j$ is non-zero, there exists a scalar $\beta \in \mathbb{R}$ such that βv_j belongs to the hyperplane $\mathcal{P}_i(k)$ (Figure 3). This non-zero scalar verifies

$$H_i(k)\beta v_j = 1. \tag{29}$$

Hence, it holds that β is indexed by i, j and k and satisfies

$$\beta_{(i,j,k)} = \frac{1}{H_i(k)v_j}. \tag{30}$$

The following remark guarantees the existence of such a quantity.

Remark 3
At each time k, because $\mathcal{X}_\Delta(k)$ is assumed to be convex and compact, then for each v_j, $j = 1,\ldots,p_{\hat{\Omega}}(k)$, there exists at least one $H_i(k)$, $i = 1,\ldots,p_{\mathcal{X}}(k)$, such that $\beta_{(i,j,k)}$ is positive.

For the polytopic sets $\hat{\Omega}$ and $\mathcal{X}_\Delta(k)$, define the following strategy

$$\alpha(k) = \min\left\{\max\{0,\beta_{(i,j,k)}\},\ i = 1,\ldots,p_{\mathcal{X}}(k),\ j = 1,\ldots,p_{\hat{\Omega}}\right\}. \tag{31}$$

Such a strategy delivers at each time k a contracting or a dilating factor, such as the set $\bar{\Omega}(k) = \alpha(k)\hat{\Omega} \subseteq \mathcal{X}_\Delta(k)$ is the largest 'copy' of $\hat{\Omega}$ contained in $\mathcal{X}_\Delta(k)$ up to an homothetic factor $\alpha(k)$.

Proposition 1
The convex polytope $\alpha(k)\hat{\Omega}$ under the strategy (31) is an invariant set for the system (8) under time-varying constraints $\mathcal{X}_\Delta(k)$.

Proof
Because $\hat{\Omega}$ is an invariant set for (8), then $\alpha(k)\hat{\Omega}$ is also an invariant set for (8) the in absence of constraints, in virtue of the invariance properties for linear systems ([7]). In addition, because $\alpha(k)\hat{\Omega} \subseteq \mathcal{X}_\Delta(k)$, then $\alpha(k)\hat{\Omega}$ is also an invariant set for (8) under constraints $\mathcal{X}_\Delta(k)$ in virtue of Definition 1. □

Then, we have the final result.

Corollary 1
The control law given by the solution of (21) with $X_f(k) = \alpha(k)\hat{\Omega}$ and $\alpha(k)$ defined as in (31), guarantees the convergence and the stability of Δx around the origin.

DOI: 10.1002/oca

Proof
It is a straightforward consequence of Proposition 1 which ensures that $X_f(k) = \alpha(k)\hat{\Omega}$ is an invariant set for (8) under constraints $\mathcal{X}_\Delta(k)$. Indeed, such a property is sufficient to guarantee the stability of Δx around the origin. □

4. NUMERICAL EXAMPLE

Let us consider the same discrete-time system (16) as in example 1 given by

$$\mathbf{A} = \begin{bmatrix} 0.9 & 0.25 \\ -0.25 & 0.9 \end{bmatrix}, \ \mathbf{B} = \begin{bmatrix} 0.5 \\ 2 \end{bmatrix}. \tag{32}$$

The aim is to calculate a control law in order to track the reference given by

$$\bar{x}(k) = \begin{cases} [0.5; 0]^T & \text{for } k < 100, \\ [0.1; 0.4]^T & \text{for } k \geqslant 100, \end{cases} \tag{33}$$

subject to the polytopic time-varying tracking constraints $X_\Delta(k)$ and $U_\Delta(k)$ given by

$$X_\Delta(k) = \{\Delta x \in \mathbb{R}^2 : \mathbf{H}_{X_\Delta}(k)\Delta x \leqslant \bar{1}\}, \tag{34}$$

with

$$\mathbf{H}_{X_\Delta}(k) = \begin{cases} \mathbf{H}_{X_\Delta}^{(1)} & \text{for } k < 30, \\ \mathbf{H}_{X_\Delta}^{(2)} & \text{for } 30 \leqslant k < 90, \\ \mathbf{H}_{X_\Delta}^{(3)} & \text{for } 90 \leqslant k < 140, \\ \mathbf{H}_{X_\Delta}^{(4)} & \text{for } k \geqslant 140, \end{cases} \tag{35}$$

where

$$\mathbf{H}_{X_\Delta}^{(1)} = \begin{bmatrix} 2.5 & 0 \\ 0 & 2.5 \\ -2.5 & 0 \\ 0 & -2.5 \end{bmatrix}, \ \mathbf{H}_{X_\Delta}^{(2)} = \mathbf{H}_{X_\Delta}^{(4)} = \begin{bmatrix} 10 & 0 \\ 0 & 10 \\ -10 & 0 \\ 0 & -10 \end{bmatrix}, \ \mathbf{H}_{X_\Delta}^{(3)} = \begin{bmatrix} 3.333 & 0 \\ 0 & 2.5 \\ -3.333 & 0 \\ 0 & -2.5 \end{bmatrix}. \tag{36}$$

Besides,

$$U_\Delta(k) = \{\Delta u \in \mathbb{R} : \mathbf{H}_{U_\Delta}(k)\Delta u \leqslant \bar{1}\}, \tag{37}$$

with

$$\mathbf{H}_{U_\Delta}(k) = \begin{cases} \mathbf{H}_{U_\Delta}^{(1)} & \text{for } k < 30, \\ \mathbf{H}_{U_\Delta}^{(2)} & \text{for } 30 \leqslant k < 90, \\ \mathbf{H}_{U_\Delta}^{(3)} & \text{for } 90 \leqslant k < 140, \\ \mathbf{H}_{U_\Delta}^{(4)} & \text{for } k \geqslant 140, \end{cases} \tag{38}$$

where

$$\mathbf{H}_{U_\Delta}^{(1)} = \begin{bmatrix} 25 \\ -25 \end{bmatrix}, \ \mathbf{H}_{U_\Delta}^{(2)} = \mathbf{H}_{U_\Delta}^{(4)} = \begin{bmatrix} 100 \\ -100 \end{bmatrix}, \ \mathbf{H}_{U_\Delta}^{(3)} = \begin{bmatrix} 20 \\ -25 \end{bmatrix}. \tag{39}$$

DOI: 10.1002/oca

Let us choose the feedback gain K as $K = [-0.0343, -0.1478]$, noticing that it does not depend on the constraints. Then, the time-varying closed-loop constraints $\mathcal{X}_\Delta(k)$ are given by the polytopic representation

$$\mathcal{X}_\Delta(k) = \{\Delta x \in \mathbb{R}^2 : \mathbf{H}_{\mathcal{X}_\Delta}(k)\Delta x \leqslant \bar{1}\}, \tag{40}$$

with

$$\mathbf{H}_{\mathcal{X}_\Delta}(k) = \begin{cases} \mathbf{H}_{\mathcal{X}_\Delta}^{(1)} & \text{for } k < 30, \\ \mathbf{H}_{\mathcal{X}_\Delta}^{(2)} & \text{for } 30 \leqslant k < 90, \\ \mathbf{H}_{\mathcal{X}_\Delta}^{(3)} & \text{for } 90 \leqslant k < 140, \\ \mathbf{H}_{\mathcal{X}_\Delta}^{(4)} & \text{for } k \geqslant 140, \end{cases} \tag{41}$$

where

$$\mathbf{H}_{\mathcal{X}_\Delta}^{(1)} = \begin{bmatrix} 2.5 & 0 \\ 0 & 2.5 \\ -2.5 & 0 \\ 0 & -2.5 \\ -0.86 & -3.69 \\ 0.86 & 3.69 \end{bmatrix}, \mathbf{H}_{\mathcal{X}_\Delta}^{(2)} = \mathbf{H}_{\mathcal{X}_\Delta}^{(4)} = \begin{bmatrix} 10 & 0 \\ 0 & 10 \\ -10 & 0 \\ 0 & -10 \\ -3.43 & -14.78 \\ 3.43 & 14.78 \end{bmatrix}, \mathbf{H}_{\mathcal{X}_\Delta}^{(3)} = \begin{bmatrix} 3.33 & 0 \\ 0 & 2.5 \\ -3.33 & 0 \\ 0 & -2.5 \\ -0.69 & -2.96 \\ 0.69 & 2.96 \end{bmatrix}. \tag{42}$$

4.1. *Off-line step: invariant set*

According to Remark 2, the choice of the invariant set $\hat{\Omega}$ does not impact the stability and must be made according to some heuristics related to the specificity of the application. Here, as an arbitrary choice, it is built from the feedback gain K (obtained by solving the LQR problem with weighting matrices $\mathbf{Q} = [1\,0; 0\,1]$, $\mathbf{R} = 30$ and $\mathbf{P} = [4.6534, 0.5613; 0.5613, 3.0237]$ of the MPC problem (21)) and the same constraints (17) as in the time-invariant case. Hence, the resulting invariant $\hat{\Omega}$ is defined as in (19).

4.2. *On-line steps: MPC with homothetic transformation*

By applying the strategy (31), the following dilating/contracting homothetic factors have been obtained

$$\alpha(k) = \begin{cases} 2.67 & \text{for } k < 30, \\ 0.67 & \text{for } 30 \leqslant k < 90, \\ 2 & \text{for } 90 \leqslant k < 140, \\ 0.67 & \text{for } k \geqslant 140. \end{cases} \tag{43}$$

The time evolution of $\alpha(k)$ is plotted in Figure 4. The time-varying constraints (36) and (42) are depicted in Figures 5–7 for each time interval. The nominal invariant set $\hat{\Omega}$ and the invariant sets obtained after the homothetic transformation with factors (43) are depicted in Figures 8–10, respectively. The figures clearly illustrate that for $\alpha(k) > 1$ (respectively $\alpha(k) < 1$), the nominal invariant set $\hat{\Omega}$ dilates (respectively contracts) homogeneously. From the figures, it is clear that the largest homothethic invariant set is still a subset of $\mathcal{X}_\Delta(k)$ as expected. The homothetic dilation and contraction of the nominal invariant set according to (43) is depicted in Figure 11.

The tracking of the reference (33) with initial conditions $x(0) = [0; 0.3]^T$ is depicted for $x_1(k)$ and $x_2(k)$ in Figures 12 and 13, respectively. The tracking errors $\Delta x_1(k)$ and $\Delta x_2(k)$ are depicted in Figures 14 and 15. The control $\Delta u(k)$ is the solution of the MPC problem (21), being the terminal constraint $X_f(k)$ equal to the time-parameter dependent invariant set $\alpha(k)\hat{\Omega}$ as given by (43). It is depicted in Figure 16. The plots highlight that the tracking is achieved while the time-varying constraints are fulfilled.

DOI: 10.1002/oca

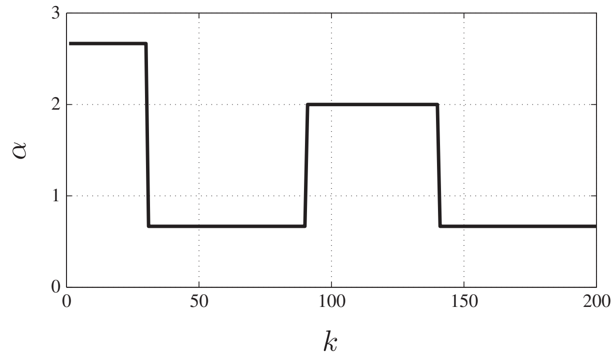

Figure 4. Behaviour of $\alpha(k)$.

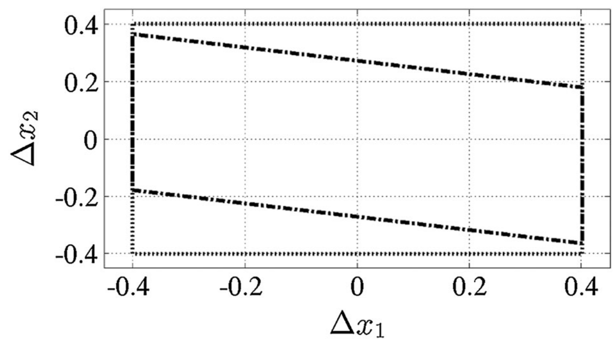

Figure 5. Constraints for $k < 30$. Dotted line, constraint $X_\Delta(k)$; dash-dotted line, constraint $\mathcal{X}_\Delta(k)$.

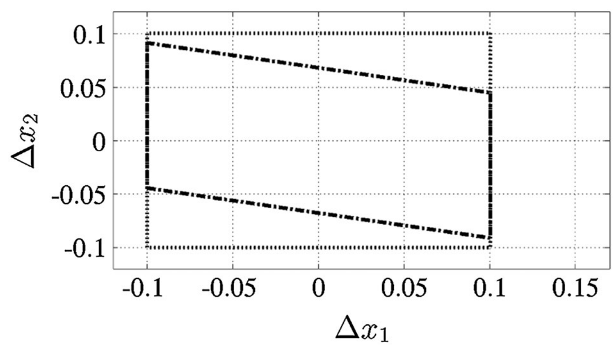

Figure 6. Constraints for $30 \leqslant k < 90$ or $k \geqslant 140$. Dotted line, constraint $X_\Delta(k)$; dash-dotted line, constraint $\mathcal{X}_\Delta(k)$.

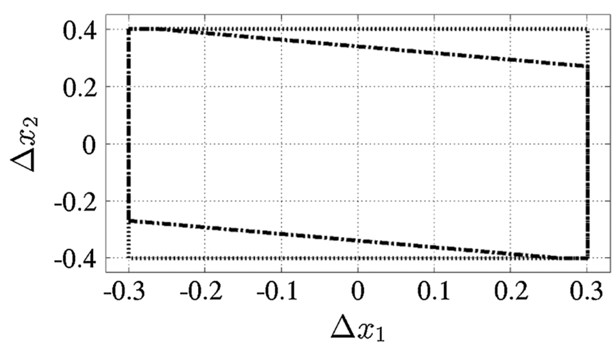

Figure 7. Constraints for $90 \leqslant k < 140$. Dotted line, constraint $X_\Delta(k)$; dash-dotted line, constraint $\mathcal{X}_\Delta(k)$.

DOI: 10.1002/oca

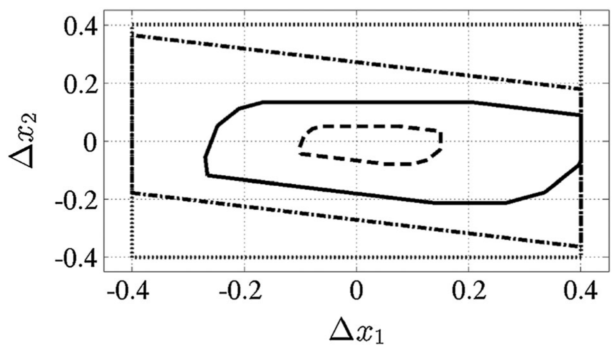

Figure 8. Dotted line, constraint $X_{\Delta}(k)$; dash-dotted line, constraint $\mathcal{X}_{\Delta}(k)$; dashed line, nominal invariant set $\hat{\Omega}$; solid line, $\alpha(k)\hat{\Omega}$ with $\alpha(k) = 2.67$.

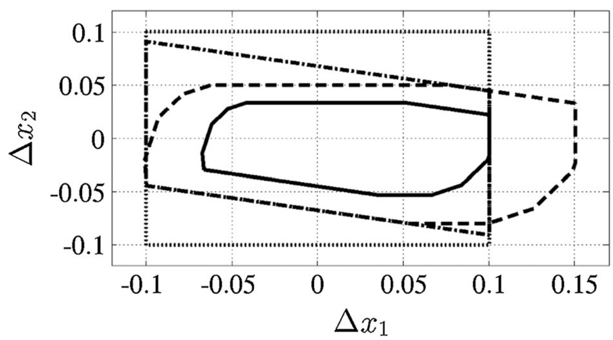

Figure 9. Dotted line, constraint $X_{\Delta}(k)$; dash-dotted line, constraint $\mathcal{X}_{\Delta}(k)$; dashed line, nominal invariant set $\hat{\Omega}$; solid line, $\alpha(k)\hat{\Omega}$ with $\alpha(k) = 0.67$.

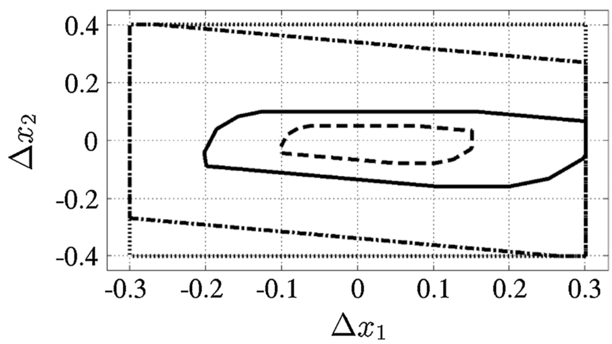

Figure 10. Dotted line, constraint $X_{\Delta}(k)$; dash-dotted line, constraint $\mathcal{X}_{\Delta}(k)$; dashed line, nominal invariant set $\hat{\Omega}$; solid line, $\alpha(k)\hat{\Omega}$ with $\alpha(k) = 2$.

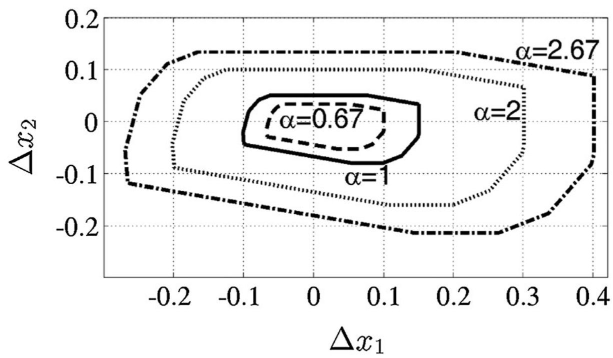

Figure 11. Solid line, nominal invariant set $\hat{\Omega}$; dashed line, $0.67\hat{\Omega}$; dotted line, $2\hat{\Omega}$; dash-dotted line, $2.67\hat{\Omega}$.

DOI: 10.1002/oca

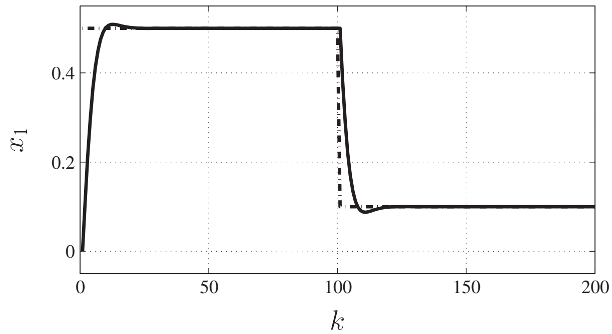

Figure 12. Solid line, tracking response of $x_1(k)$; dash-dotted line, reference.

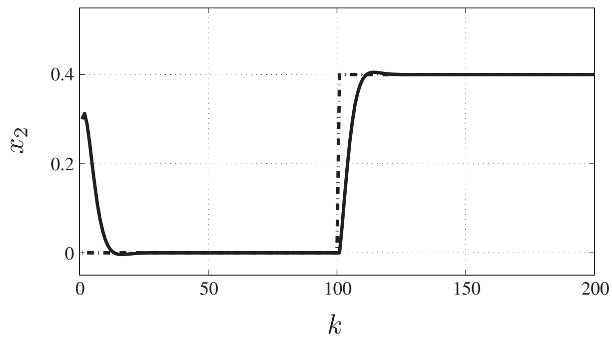

Figure 13. Solid line, tracking response of $x_2(k)$; dash-dotted line, reference.

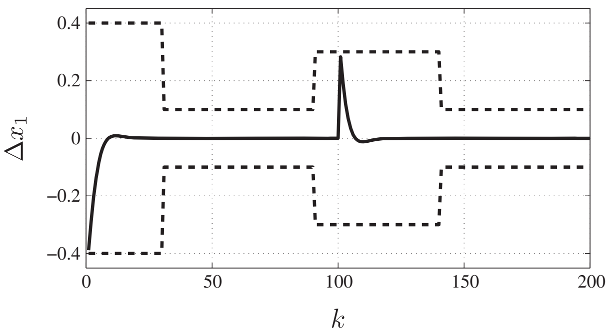

Figure 14. Solid line, tracking error $\Delta x_1(k)$; dashed line, time-varying constraints.

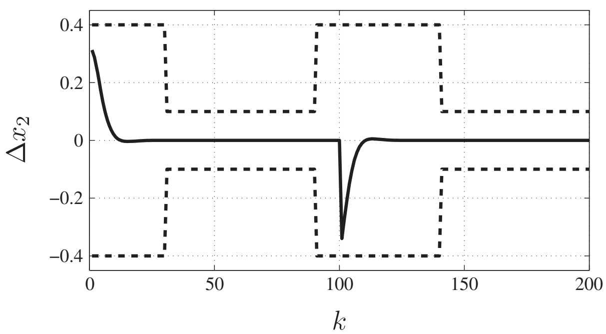

Figure 15. Solid line, tracking error and $\Delta x_2(k)$; dashed line, time-varying constraints.

DOI: 10.1002/oca

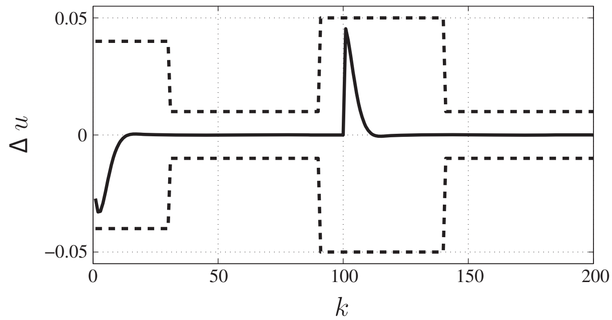

Figure 16. Solid line, tracking error $\Delta u(k)$; dashed line, time-varying constraints.

Table I. Parameters involved in the electric vehicle dynamics.

Coefficients	Description	Value
m	vehicle mass	90 kg
η	efficiency of the inverter	0.97
k_t	motor constant	0.0604 Nm/A
g_r	transmission gear ratio	8.5
r_w	radius of the wheels	0.24 m
ρ	air density coefficient	1.225 kg/m^3
$C_d A_f$	aerodynamic drag coefficient × vehicle frontal area	0.1031 m^2
g	gravity acceleration coefficient	9.81 m/s^2
C_r	wheels rolling resistance coefficient	8.1549×10^{-4}

5. CASE STUDY : VELOCITY TRACKING FOR AN ELECTRIC VEHICLE

This section is devoted to a case study. An electric vehicle is required to follow a minimum consumption velocity profile according to its actual position. The velocity profile, also known as *optimal driving strategy* [15, 16], is computed off-line for a prescribed path (succession of straight lines and curves) and guarantees the minimum energetic consumption to travel the path in a finite time t_f. The real-time MPC controller must ensure that the vehicle velocity tracks the optimal reference. The actual position and velocity of the vehicle are measured on-line. The tracking task is subject to time-varying velocity and input constraints as it will be detailed subsequently. The model under consideration corresponds to the prototype developed by the Research Center for Automatic Control of Nancy, in France, which is annually involved in the European Shell Eco-Marathon race in the Plug-in (battery) category.

5.1. *Electric vehicle dynamics*

In the electric vehicle, the battery provides all the traction power. Therefore, the dynamics of the electric vehicle can be expressed by means of state-space equations where the battery current $\tilde{I}_{batt}(t)$ is the control signal, denoted hereafter with $\tilde{u}(t)$. The current delivered by the battery saturates at $\tilde{u}_{max}$.

The nonlinear discretized dynamics (for a flat road) reads

$$\check{x}_2(k+1) = \check{x}_2(k) + T_s \frac{\eta k_t g_r \check{u}(k)}{m r_w} - \frac{1}{2m} T_s \rho C_d A_f \check{x}_2(k)^2 - T_s g C_r, \tag{44}$$

being $\check{x}_2(k)$ the discretized velocity of the vehicle and T_s the sampling time. The position of the vehicle, here noted as $\check{x}_1(k)$, follows $\check{x}_1(k+1) = \check{x}_1(k) + T_s \check{x}_2(k)$. The numerical values of the parameters involved in the dynamics (44) have been obtained from both physical considerations and experimental data-based parameter identification. They are reported in Table I.

DOI: 10.1002/oca

Having in mind the design of an optimal control minimizing the consumption of the vehicle, it is useful for the sequel to derive a linearized model around an average velocity $\check{x}_{2e}$. The corresponding operating point is the pair $(\check{x}_{2e}, \check{u}_e)$ where $\check{u}_e \in [0, \check{u}_{max} = \tilde{u}_{max}]$ is the solution of the steady-state condition $\check{x}_2(k+1) = \check{x}_2(k)$ in (44), that is,

$$\check{u}_e = \frac{\rho C_d A_f r_w (\check{x}_{2e})^2 + 2mgr_w C_r}{2\eta k_t g_r}. \tag{45}$$

The following linearized discrete-time state-space dynamics is obtained from (44) around the operating point $(\check{x}_{2e}, \check{u}_e)$:

$$x(k+1) = \mathbf{A}x(k) + \mathbf{B}u(k), \tag{46}$$

where $u(k) = \check{u}(k) - \check{u}_e$ and $x(k) = [x_1(k),\ x_2(k)]^T = [\check{x}_1(k) + \check{x}_{2e}T_s,\ \check{x}_2(k) - \check{x}_{2e}]^T \in \mathbb{R}^2$. The matrices $\mathbf{A}$ and $\mathbf{B}$ are given by

$$\mathbf{A} = \begin{bmatrix} 1 & T_s \\ 0 & 1 - \frac{T_s \rho C_d A_f}{m}\check{x}_{2e} \end{bmatrix}, \qquad \mathbf{B} = \begin{bmatrix} 0 \\ \frac{T_s \eta k_t g_r}{m r_w} \end{bmatrix}. \tag{47}$$

The output equation is the linearized velocity $x_2(k) = \check{x}_2(k) - \check{x}_{2e}$ and reads

$$y(k) = \mathbf{C}x(k) \text{ where } \mathbf{C} = \begin{bmatrix} 0 & 1 \end{bmatrix}. \tag{48}$$

The full state is assumed to be accessible, which means that both the position and the velocity are measurable.

5.2. *Optimal driving strategy*

Finding a low-consumption strategy amounts to an electrical resource management problem. Indeed, the energy level of the battery is the bottleneck in the optimization problem and requires an optimal driving strategy solution [15, 17]. Only the discharging of the battery is taken into account. Disregarding the losses, the energy $\tilde{E}_s$ stored in the battery at time t can be merely expressed in terms of the current flowing through the battery $\tilde{u}(t)$ and the open circuit voltage V_{oc} [17, 18]:

$$\tilde{E}_s(t) = \tilde{E}_s(0) - \int_0^t V_{oc}\tilde{u}(\tau)\, d\tau, \tag{49}$$

where $\tilde{E}_s(0)$ is the initial energy stored in the battery. From the right-hand side of (49), it is clear that the minimum consumption is achieved if the minimum battery current $\tilde{u}^*(t)$ is used to perform the driving task, the voltage V_{oc} being assumed to be constant during the race.

The open-loop control is obtained off-line by solving the nonlinear optimal control problem for a flat path of 3.266 km long, a maximum allowed time of $t_f = 468$ s and a maximum allowed of input $\tilde{u}_{max} = 7$ Å. The resulting optimal trajectory and open-loop control are depicted in Figures 17

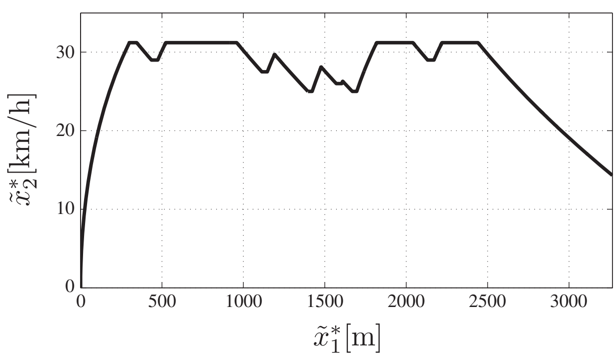

Figure 17. Optimal velocity profile.

DOI: 10.1002/oca

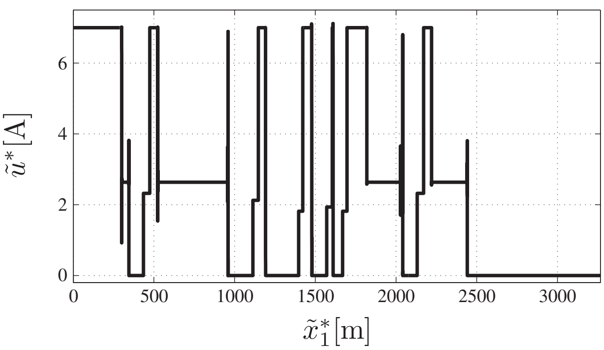

Figure 18. Optimal battery current profile.

and 18, respectively, with respect to the vehicle position in the path. The driving strategy gives an expected performance of 474.0826 km/kwh. In the following, it is detailed how an MPC-based strategy can be used to steer the state towards the optimal open-loop trajectory.

5.3. Time-varying tracking constraints.

Because the vehicle starts at a speed equal to zero, the initial natural strategy is to accelerate as much as possible to reach the optimal nominal velocity. That precisely corresponds to the optimal solution as illustrated in Figures 17 and 18. Indeed, the optimal battery current profile is $\tilde{u}^* = \tilde{u}_{max} = 7$ Å for a distance less than 300 m. For the distance $\tilde{x}_1^* = 300$ m, the optimal velocity $\tilde{x}_2^*$ is 31.2 km/h. Analogously, when the vehicle approaches the end of the path, the natural energetically optimal behaviour consists in switching off the motor propulsion. The switching off is carried out at the distance of $\tilde{x}_1^* = 2441$ m. In the range 300 2441 m, as aforementioned, a control law must be designed to track the optimal driving strategy. In this range, the optimal velocity is not considered to have large fluctuations. Thus, it is reasonable to consider the linearized model (46)–(48) with $\check{x}_{2e}$ as the average optimal velocity within such a range, that is, 27 km/h.

Our objective is to derive an MPC law which will be applied to the nonlinear dynamics of the vehicle to guarantee that the nonlinear state remains as close as possible to the optimal one. Because the MPC is a discrete linear law, the following quantities must be introduced. The states $\check{x}^*(k) = [\check{x}_1^*(k),\ \check{x}_2^*(k)]^T \in \mathbb{R}^2$ and the input $\check{u}^*(k)$ are obtained from the sampling of the optimal trajectory $\tilde{x}^*(t) = [\tilde{x}_1^*(t),\ \tilde{x}_2^*(t)]^T \in \mathbb{R}^2$ and the sampling of the optimal input $\tilde{u}^*(t)$, respectively, with sampling period T_s. Besides, we define the error $\Delta u(k)$ for the system (46) as

$$\Delta u(k) = u(k) - (\check{u}^*(k) - \check{u}_e), \tag{50}$$

and the error $\Delta x(k)$ as

$$\Delta x(k) = x(k) - (\check{x}^*(k) - \check{x}_e). \tag{51}$$

with $\check{x}_e = [-\check{x}_{2e}T_s, \check{x}_{2e}]^T$. The dynamics of the errors $\Delta u(k)$ and $\Delta x(k)$ are nonlinear, but close enough to the operating point $(\check{x}_{2e}, \check{u}_e)$, the dynamics can be approximated by the following linear one

$$\Delta x(k+1) = \mathbf{A}\Delta x(k) + \mathbf{B}\Delta u(k). \tag{52}$$

Now, a bounded set of constraints $X_\Delta(k)$ for every k is imposed. As it turns out, no special constraints are required for the position accuracy Δx_1, because the tracking is only concerned with the velocity. To this end, in practice, we impose very large constraints as -100 m $\leqslant \Delta x_1(k) \leqslant 100$ m for every k. Regarding the tracking constraints on the velocity, it is worth considering robustness issues. Indeed, there are unavoidable mismatches between the parameters of the model and the actual ones. Discrepancies regarding the mass of the vehicle or the frictions will be more critical during the initial acceleration and the final deceleration. For example, an increasing of the mass will lead to a slower acceleration. Therefore, the constraints for Δx_2 must be relaxed at the beginning and at the

DOI: 10.1002/oca

end of the path. Consequently, the constraints for Δx_2 are time varying. The resulting constraints $X_\Delta(k)$ are represented by the following polytopic description $\mathbf{H}_{X_\Delta}(k)$ as defined in (34):

$$\mathbf{H}_{X_\Delta}(k) = \begin{cases} \mathbf{H}_{X_\Delta}^{(1)} & \text{for } \check{x}_1^*(k) < 944 \text{ m}, \\ \mathbf{H}_{X_\Delta}^{(2)} & \text{for } 944\text{m} \leqslant \check{x}_1^*(k) < 2588 \text{ m}, \\ \mathbf{H}_{X_\Delta}^{(3)} & \text{for } \check{x}_1^*(k) \geqslant 2588 \text{ m}, \end{cases} \tag{53}$$

where

$$\mathbf{H}_{X_\Delta}^{(1)} = \begin{bmatrix} 0.01 & 0 \\ 0 & 3.6 \\ -0.01 & 0 \\ 0 & -0.6 \end{bmatrix}, \mathbf{H}_{X_\Delta}^{(2)} = \begin{bmatrix} 0.01 & 0 \\ 0 & 3.6 \\ -0.01 & 0 \\ 0 & -1.2 \end{bmatrix}, \mathbf{H}_{X_\Delta}^{(3)} = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.6 \\ -0.1 & 0 \\ 0 & -1.8 \end{bmatrix}. \tag{54}$$

As far as the input tracking constraints U_Δ are concerned, the battery current $\check{u}(k)$ in (44) must be fulfilled, for all k

$$0 \leqslant \check{u}(k) \leqslant \check{u}_{max}. \tag{55}$$

Then, to avoid the inadmissible values of the input, $\Delta u(k)$ in (52) has to satisfy the following constraint

$$\begin{aligned} \Delta u_{min}(k) \leqslant \Delta u(k) \leqslant \Delta u_{max}(k), \\ -\check{u}^*(k) \leqslant \Delta u(k) \leqslant \check{u}_{max} - \check{u}^*(k). \end{aligned} \tag{56}$$

The constraints are time varying, because $\Delta u(k)$ depends on the optimal control input $\check{u}^*(k)$ (Figure 18) that may change in time. Additionally, $0 \in int(U_\Delta(k))$ must be satisfied for every k, and therefore, a small tolerance $\epsilon = 1 \times 10^{-6}$ is introduced for the cases where $\check{u}^*(k) = \check{u}_{max}$ or $\check{u}^*(k) = 0$. In this way, the constraints $U_\Delta(k)$ on the input, as defined in (37), obey

$$\mathbf{H}_{U_\Delta}(k) = \begin{cases} \mathbf{H}_{U_\Delta}^{(1)} & \text{if } \check{u}^*(k) = \check{u}_{max}, \\ \mathbf{H}_{U_\Delta}^{(2)} & \text{if } \check{u}^*(k) = 0, \\ \mathbf{H}_{U_\Delta}^{(3)} & \text{otherwise}, \end{cases} \tag{57}$$

where

$$\mathbf{H}_{U_\Delta}^{(1)} = \begin{bmatrix} (\epsilon)^{-1} \\ (-\check{u}_{max})^{-1} \end{bmatrix}, \mathbf{H}_{U_\Delta}^{(2)} = \begin{bmatrix} (\check{u}_{max})^{-1} \\ (-\epsilon)^{-1} \end{bmatrix}, \mathbf{H}_{U_\Delta}^{(3)} = \begin{bmatrix} (\check{u}_{max} - \check{u}^*(k))^{-1} \\ (-\check{u}^*(k))^{-1} \end{bmatrix}. \tag{58}$$

The time-varying tracking constraints for $\Delta u(k)$ (56) are depicted in Figure 19 with respect to the vehicle position.

5.4. Model predictive control

In order to make the tracking task appropriate for real-time implementation despite the time-varying constraints, the strategy of homothetic transformation of the invariant set is applied. To this end, a nominal invariant set $\hat{\Omega}$ must be precomputed off-line for (52) and then scaled on-line to fit the time-varying constraints represented by (53) and (57), as seen in Section 3. This invariant set will act as a terminal set constraint, and then stability and convergence of the state to steady-state values will be guaranteed by the MPC algorithm.

DOI: 10.1002/oca

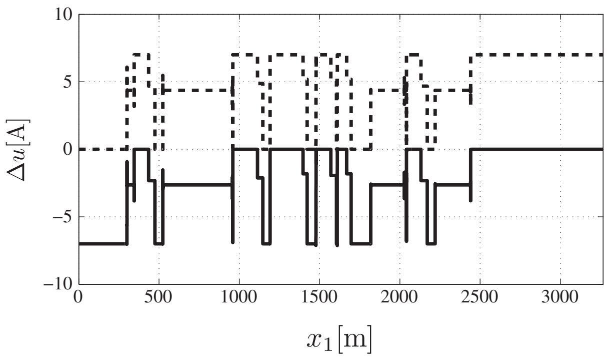

Figure 19. Solid line, $\Delta u_{min}(k)$; dashed line, $\Delta u_{max}(k)$.

5.4.1. Off-line step: invariant set. Setting $\mathbf{Q} = [0\ 0; 0\ 1]$, $\mathbf{R} = 1$, $\mathbf{P} = [0\ 0; 0\ 139.75]$ and by choosing the nominal tracking constraints $\hat{X}_\Delta$ and $\hat{U}_\Delta$ as

$$\mathbf{H}_{\hat{X}_\Delta} = \begin{bmatrix} 0.02 & 0 \\ 0 & 7.20 \\ -0.02 & 0 \\ 0 & -7.20 \end{bmatrix}, \ \mathbf{H}_{\hat{U}_\Delta} = \begin{bmatrix} 0.1 \\ -0.1 \end{bmatrix}, \tag{59}$$

the following closed-loop nominal constraints $\hat{\mathcal{X}}_\Delta$ are obtained with $K = [0\ \ -0.6411]$

$$\mathbf{H}_{\hat{\mathcal{X}}_\Delta} = \begin{bmatrix} 0.02 & 0 \\ 0 & 7.20 \\ -0.02 & 0 \\ 0 & -7.20 \\ 0 & -0.064 \\ 0 & 0.064 \end{bmatrix}. \tag{60}$$

The maximal invariant set $\hat{\Omega}$ that fits (60) is given by the following vertex representation

$$\mathbf{V}_{\hat{\Omega}} = \begin{bmatrix} 50 & -0.1389 \\ 50 & -0 \\ 49.7020 & 0.1389 \\ -49.7020 & -0.1389 \\ -50 & 0.1389 \\ -50 & 0 \end{bmatrix}^T. \tag{61}$$

5.4.2. On-line steps: MPC with homothetic transformation. From the strategy (31), the scaling factor is calculated on-line and verifies $\alpha(k) \in \{0.067,\ 0.561,\ 1.752,\ 2\}$. Then, the nominal invariant set $\hat{\Omega}$, given by (61), is scaled with the homothetic transformation to fit the closed-loop constraints $\mathcal{X}_\Delta(k)$ given by

$$\mathbf{H}_{\mathcal{X}_\Delta}(k) = \begin{bmatrix} \mathbf{H}_{X_\Delta}(k) \\ \mathbf{H}_{U_\Delta}(k)K \end{bmatrix}, \tag{62}$$

with $\mathbf{H}_{X_\Delta}(k)$ and $\mathbf{H}_{U_\Delta}(k)$ as given by (53) and (57), respectively. The invariant set $\hat{\Omega}$ and the different homothetic transformations $\alpha(k)\hat{\Omega}$ for the distinct values of α are depicted in Figure 20.

The tracking task is performed by closing the loop with the nonlinear discrete dynamics, as is depicted in Figure 21. The optimal solution $\Delta u(k), \Delta u(k+1), \ldots, \Delta u(k+N_p)$ of the MPC

DOI: 10.1002/oca

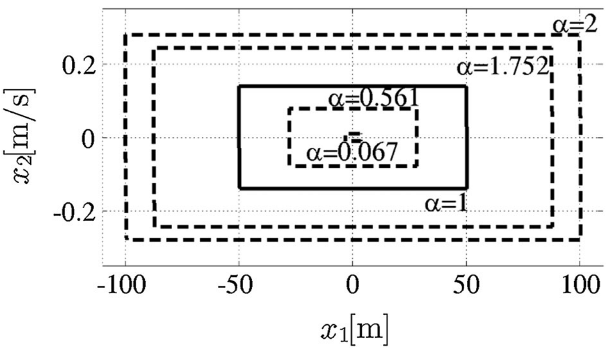

Figure 20. Solid line, nominal invariant set $\hat{\Omega}$; dashed lines, dilation or contraction of the nominal invariant set.

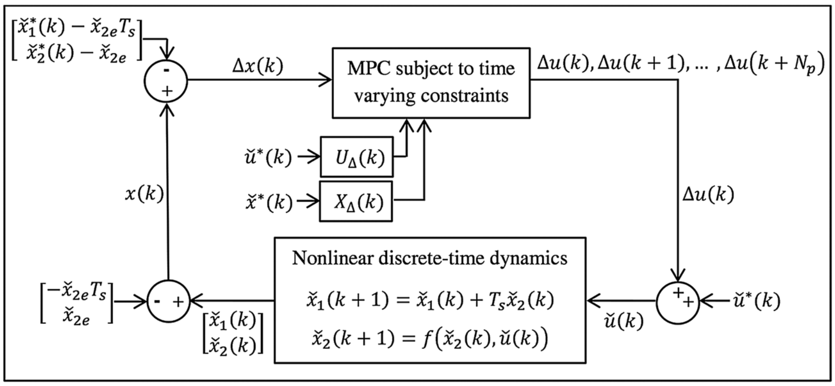

Figure 21. Closed-loop implementation of the time-varying MPC. The function that describes the velocity $\check{x}_2(k+1) = f(\check{x}_2(k), \check{u}(k))$ is given by (44).

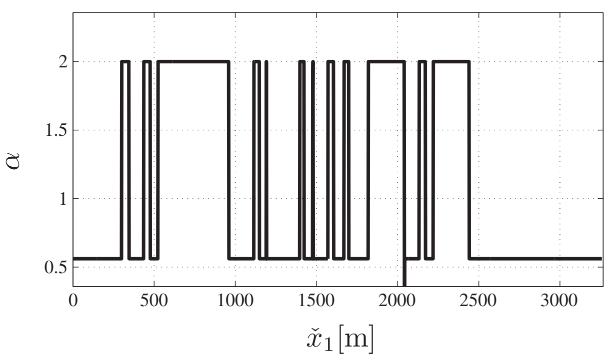

Figure 22. Behaviour of α.

problem (21) is computed from the actual state $\check{x}(k) = [\check{x}_1(k), \check{x}_2(k)]^T$ of the nonlinear dynamics (44) using the linearized tracking error $\Delta x(k) = x(k) - (\check{x}^*(k) - \check{x}_e)$ (Figure 21). The homothetic dilation/contraction of the invariant set is applied within the MPC algorithm to fit the closed-loop constraints $\mathcal{X}_\Delta(k)$. The first component $\Delta u(k)$ of the optimal solution of the MPC problem is used to shape the control input $\check{u}(k)$ for the nonlinear dynamics by making $\check{u}(k) = \Delta u(k) + \check{u}^*(k)$ (Figure 21). Let us note that to comply with Assumption 1, the references $\check{u}^*(k)$ and $\check{x}^*(k)$ are kept constant during the prediction horizon N_p. The sampling period is $T_s = 0.2$ s and the prediction horizon is $N_p = 10$.

The evolution of α is plotted in Figure 22 with respect to the vehicle position. The tracking task is performed for a mass variation of 10% and 50% in the nonlinear model (44). The tracking response

DOI: 10.1002/oca

$\check{x}_2(k)$ of the nonlinear system is depicted in Figure 23 with respect to the vehicle position. The difference between the actual nonlinear velocity $\check{x}_2(k)$ and the target state $\check{x}^*(k)$ at time k, that is, $\Delta\check{x}_2(k) = \check{x}_2(k) - \check{x}_2^*(k)$, is depicted in Figure 24 with respect to the vehicle position. In Figure 25, the input for the nonlinear tracking $\Delta\check{u}(k) = \check{u}(k) - \check{u}^*(k)$ is depicted with respect to the vehicle position.

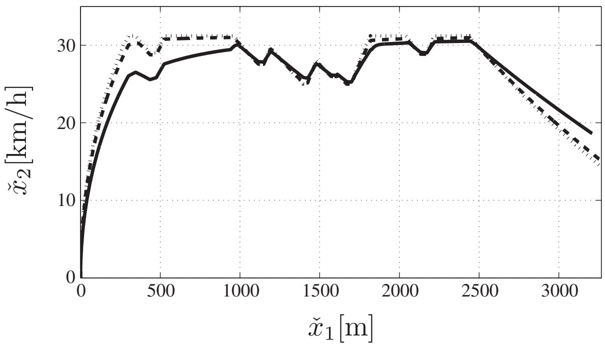

Figure 23. Dotted line, reference $\check{x}_2^*$; dashed line, tracking response of the nonlinear system with a mass variation of 10%; solid line, tracking response of the nonlinear system with a mass variation of 50%.

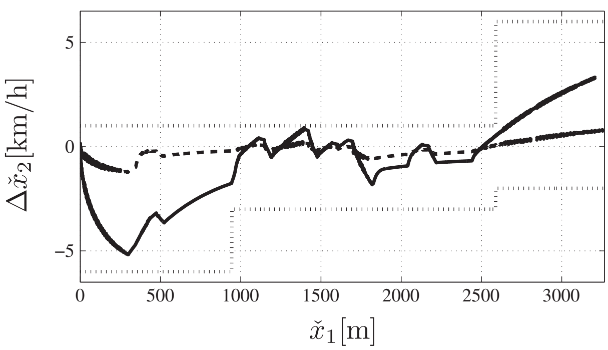

Figure 24. Dotted lines, $\Delta x_{2min}(k)$ and $\Delta x_{2max}(k)$; dashed line, tracking error $\Delta\check{x}_2(k)$ of the nonlinear system with a mass variation of 10%; solid line, tracking error $\Delta\check{x}_2(k)$ of the nonlinear system with a mass variation of 50%.

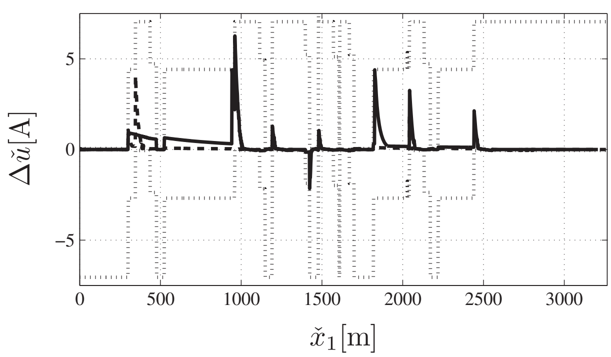

Figure 25. Dotted lines, $\Delta u_{min}(k)$ and $\Delta u_{max}(k)$; dashed line, tracking input $\Delta\check{u}(k)$ of the nonlinear system with a mass variation of 10%; solid line, tracking input $\Delta\check{u}(k)$ of the nonlinear system with a mass variation of 50%.

DOI: 10.1002/oca

6. CONCLUSION AND FUTURE WORKS

This paper has presented a real-time MPC-based tracking strategy for linear systems subject to time-varying constraints in the state and/or the input. A polytopic invariant set computed off-line has been homogeneously dilated or contracted on-line to fit the polytopic time-varying constraints. The contraction/dilation of the invariant set has been performed using the invariant set theory and the properties of invariant sets for discrete-time linear system. The resulting time-varying invariant set has been used as an admissible terminal constraint set so that it has guaranteed stability and convergence in the tracking task. It has been shown that time-varying constraints can allow to take into account practical concerns such as the consideration of saturations, preserving feasibility despite uncertainties and so on.

Beyond the interest of a solution which is appropriate to cope with time-varying constraints, the approach is well suited for real-time applications. Indeed, the additional cost of the approach, compared with the standard MPC for the time-invariant case, is quite negligible.

As future work, we will study the necessary conditions to ensure recursive feasibility under different assumptions regarding the available information on the time-varying constraints. Additionally, it would be interesting to particularize the approach to specific nonlinear dynamics like LPV systems.

REFERENCES

1. Mayne DQ, Rawlings JB, Rao CV, Scokaert POM. Constrained model predictive control: stability and optimality. *Automatica* 2000; **36**(6):789–814.
2. Bemporad A, Morari M. Robust model predictive control: a survey. *Lecture Notes in Control and Information Sciences* 1999; **245**:207–226.
3. Camacho EF, Bordons C. *Model Predictive Control* (2nd edn). Springer: London, UK, 2004.
4. Limon D, Alvarado I, Alamo T, Camacho EF. MPC for tracking piecewise constant references for constrained linear systems. *Automatica* 2008; **44**(9):2382–2387.
5. Gilbert EG, Tan KT. Linear systems with state and control constrains: the theory and application of maximal output admissible sets. *IEEE Transactions on Automatic Control* 1991; **36**(9):1008–1020.
6. Kolmanovsky I, Gilbert EG. Theory and computation of disturbance invariant sets for discrete time linear systems. *Mathematical Problems in Engineering: Theory, Methods and Applications* 1998; **4**(4):317–367.
7. Blanchini F, Miani S. *Set-Theoretic Methods in Control*. Birkhäuser: Boston, MA, 2007.
8. Bemporad A, Morari M, Dua V, Pistikopoulos E. The explicit linear quadratic regulator for constrained systems. *Automatica* 2002; **38**(1):3–20.
9. Alessio A, Bemporad A. A survey on explicit model predictive control. *Lecture Notes in Control and Information Sciences* 2009; **384**:345–369.
10. Mayne D, Raković S, Findeisen R, Allgöwer F. Robust output feedback model predictive control of constrained linear systems: time varying case. *Automatica* 2009; **45**(9):2082–2087.
11. Wada N, Tomosugi H, Saeki M. Model predictive tracking control for a linear system under time-varying input constraints. *International Journal of Robust and Nonlinear Control* 2013; **23**(9):945–964.
12. Scokaert POM, Rawlings JB. Constrained linear quadratic regulation. *IEEE Transactions on Automatic Control* 1998; **43**(8):1163–1169.
13. Manrique T, Fiacchini M, Chambrion T, Millerioux G. MPC for a low consumption electric vehicle with time-varying constraints. *IFAC System, Structure and Control* 2013; **5**(1):833–838.
14. Blanchini F. Set invariance in control. *Automatica* 1999; **35**(11):1747–1767.
15. Sciarretta A, Guzzella L. Control of hybrid electric vehicles. *IEEE Control Systems Magazine* 2007; **27**(2):60–70.
16. Santin JJ. *The Worlds Most Fuel Efficient Vehicle: Design and Development of PAC-Car II*. Vdf Hochschulverlag AG: Zürich, 2007.
17. Koot M, Kessels JTBA, Jager BD, Heemels WPMH, Bosch PPJVD, Steinbuch M. Energy management strategies for vehicular electric power systems. *IEEE Transactions on Vehicular Technology* 2005; **54**(3):771–782.
18. Ehsani M, Gao Y, Gay SE, Emadi A. *Modern Electric, Hybrid Electric, and Fuel Cell Vehicles: Fundamentals, Theory and design*. CRC Press: Boca Raton, FL, 2004.

DOI: 10.1002/oca