Preprint of final paper. DOI: doi: 10.1049/iet-cta.2009.0604 Copyright The Institution of Engineering and Technology 2010

Set Membership approximations of predictive control laws: the tradeoff between accuracy and complexity

M. Canale*, L. Fagiano†, M. Milanese‡ and C. Novara§

Dipartimento di Automatica e Informatica, Politecnico di Torino

Corso Duca degli Abruzzi 24, 10129 Torino, Italy.

*massimo.canale@polito.it, †lorenzo.fagiano@polito.it

‡mario.milanese@polito.it, §carlo.novara@polito.it

Abstract

The paper proposes two new techniques, in the framework of set membership (SM) theory, to derive off-line an approximation of a given nonlinear model predictive control (NMPC) law. The obtained approximated control laws satisfy input constraints and guarantee a bounded worst-case approximation error. Such a bound can be tuned to obtain a tradeoff between closed-loop performance, on-line evaluation complexity, off-line computational burden and memory usage. The presented techniques are suboptimal, since their worst-case approximation error is not minimal, however they are able to obtain good accuracy with efficient on-line computation. Both approaches are based on the prior information given by a finite number ν of nominal control moves, computed off-line and stored. The first technique relies on the piecewise linear interpolation of the off-line computed data, while the second approach is based on the computation of the (sub-optimal) upper and lower bounds of the nominal NMPC law, on the basis of the partial information given by a subset of the whole off-line computed data. A numerical example and an automotive case study are presented in order to show the effectiveness of the proposed approaches and to compare their performance.

I. Introduction

In Nonlinear Model Predictive Control (NMPC) (see e.g. [1]) the control action is computed at each sampling time by means of a receding horizon (RH) strategy, which requires the solution of a constrained optimal control problem, where the systems state x (and, possibly, other measured parameters and reference variables) is a parameter in the optimization. For time invariant systems, the solution of such a parametric optimization problem defines a static nonlinear function $\kappa(x)$, denoted here as the “nominal” control law. A main issue in the practical application of NMPC, for systems with “fast” dynamics, is the inability to solve the on-line optimization, required to compute the value of $\kappa(x)$, within the employed sampling time. In order to solve this issue, a significant research effort has been devoted in recent years to investigate either more efficient on-line optimization techniques ([2], [3]) or the use of approximated control laws $\hat{\kappa} \approx \kappa$ ([4], [5], [6], [7], [8], [9]), computed off-line and evaluated on-line instead of solving the optimization problem. As regards this second line of research, it has to be noted that while in the case of model predictive control for linear systems, with quadratic cost and linear constraints, the nominal controller κ results to be a continuous piecewise affine (PWA) function of x ([10], [11]), which can be explicitly computed off-line and stored, in the general case of nonlinear systems and/or non-quadratic cost and nonlinear constraints no explicit solution can be found. In this case, nonlinear function approximation techniques have to be used. Moreover, the use of an approximated predictive controller can be also useful to simplify the exact PWA solution in the above-mentioned case of linear system model, since the complexity of the exact explicit solution may still be prohibitive for on-line evaluation.

In this context, a first contribution was given in [4], considering a neural approximation of κ. Another approach has been proposed in [5], using an off-line approximate multi-parametric programming algorithm for the construction of a PWA approximation of the nominal predictive control law, defined over an hypercubic partition of the state region $\mathcal{X}$ where the approximation is carried out, and its implementation via a binary search tree. A similar technique, employing a simplicial partition of $\mathcal{X}$ and a PWA approximation, has been introduced in [7]. The approach [5] has been also extended to the robust min-max case in [6], using an off-line approximate multi-parametric programming technique for the construction of a piecewise nonlinear (PWNL) approximation. A further approach for approximate NMPC has been used in [8], by approximating the nonlinear system model with a set of PWA systems over the state space and computing for each one a PWA solution of the quadratic constrained finite-time optimal control ([10], [11]). Then, a set of off-line solutions of such PWA control laws is considered and a polynomial interpolation technique is employed to compute an approximation of the overall control law. Finally, different techniques have been considered in [9], where approximated MPC laws, with guaranteed accuracy (in terms of error bound on $\kappa - \hat{\kappa}$) and consequent performance and stability properties, have been derived using set membership (SM) function approximation theory. Such techniques have been effectively applied to complex problems such as the control of power kites for energy generation ([12]) and the control of semi-active suspension systems ([13]).

Note that a common feature of all of the cited approaches is the need of computing off-line a finite number ν of nominal control moves. In general, with any approximation technique the obtained accuracy improves as ν grows. However, higher ν numbers usually also imply higher memory usage and off-line and on-line computational times. Thus, a compromise has to be chosen in the design of any approximated NMPC law, between accuracy, off-line computational burden, memory usage and on-line evaluation efficiency. Indeed, different approximation techniques are able to reach different compromises: a typical example is given by the SM "optimal" and "nearest point" approximations described in [9]. With the SM "optimal" approximation, the minimal worst-case accuracy is obtained (giving in practice good accuracy with low ν values) at the price of on-line computational complexity which grows linearly with ν. On the contrary, with the SM "nearest point" technique a very efficient on-line evaluation can be obtained, but with lower accuracy.

In the described scenario, this paper proposes two new approximation techniques and investigates their accuracy properties, in the framework of SM function approximation theory. The obtained approximations are both "suboptimal" in the sense that their guaranteed approximation error is not minimal, however they allow to set up different tradeoffs between performance (i.e. accuracy), computational efficiency and memory usage. A numerical example is given to show the effectiveness of the proposed approaches and to compare their performance with that of previously investigated techniques. The paper is organized as follows. Sections II and III describe the problem formulation and the existing SM approximation techniques respectively. The new suboptimal SM approximations are presented and analyzed in Section IV. Section V provides a numerical example of low order and an automotive case study with higher order and practical relevance. Finally, conclusions are given in Section VI.

II. Problem settings

A. Nonlinear Model Predictive Control

Consider the following nonlinear state space model:

$$x_{t+1} = f(x_t, u_t) \tag{1}$$

where $x_t \in \mathbb{R}^n$ and $u_t \in \mathbb{R}^m$ are the system state and control input respectively. In this paper, it is assumed that function f in (1) is continuous over $\mathbb{R}^n \times \mathbb{R}^m$. Assume that the control objective is to regulate the system state to the origin under some input and state constraints represented by a compact set $\mathbb{U} \subseteq \mathbb{R}^m$ and a convex set $\mathbb{X} \subseteq \mathbb{R}^n$ respectively, both containing the origin in their interiors. In this paper the input constraint set is of the form $\mathbb{U} = \{u \in \mathbb{R}^m : \underline{u} \preceq u \preceq \overline{u}\}$, where the symbol $\preceq$ denotes element-wise inequalities and $\underline{u}, \overline{u} \in \mathbb{R}^m$. By denoting with N_p and $N_c \leq N_p$ the prediction horizon and the control horizon respectively, the following cost function J can be defined:

$$J(U, x_{t|t}) \doteq \Phi(x_{t+N_p|t}) + \sum_{l=0}^{N_p-1} L(x_{t+l|t}, u_{t+l|t}) \tag{2}$$

where $x_{t+l|t}$ denotes l step ahead state prediction using the model (1), given the input sequence $u_{t|t}, \ldots, u_{t+l-1|t}$ and the "initial" state $x_{t|t} = x_t$. $U \doteq [u_{t|t}^T, \ldots, u_{t+N_c-1|t}^T]^T$ is the vector of the control moves to be optimized (where T stands for vector transpose operation). The remaining predicted control moves $[u_{t+N_c|t}^T, \ldots, u_{t+N_p-1|t}^T]^T$ can be computed with different strategies, e.g. by setting $u_{t+l|t} = u_{N_c-1|t}$ or $u_{t+l|t} = K\, x_{t+l|t}$, $\forall l \in [N_c, N_p - 1]$, where K is a suitable matrix.

The NMPC control law is then obtained applying the following RH strategy:

1) At time instant t, get x_t.
2) Solve the optimization problem:

$$\min_{U} \quad J(U, x_{t|t}) \tag{3a}$$

$$\text{s. t.}$$

$$x_{t+l|t} \in \mathbb{X},\ l = 1, \ldots, N_p \tag{3b}$$

$$u_{t+l|t} \in \mathbb{U},\ l = 0, \ldots, N_p \tag{3c}$$

$$\text{stabilizing constraints} \tag{3d}$$

3) Apply the first element of the solution sequence U to the optimization problem as the actual control action $u_t = u_{t|t}$.
4) Repeat from step 1) at time $t + 1$.

Additional constraints (e.g. state contraction, terminal set, etc.. . .), indicated as "stabilizing constraints" in (3) are usually employed in order to achieve stability of the closed-loop system. It is assumed that the optimization problem (3) is feasible over a set $\mathcal{F} \subseteq \mathbb{R}^n$ which will be referred to as the "feasibility set". The application of such RH procedure implicitly defines the predictive controller as a nonlinear static function κ of the state, i.e.:

$$u_t = \kappa(x_t)$$

Note that the settings of this paper are those of a standard formulation of NMPC, where a measure of the state is assumed to be available to be used for the control move computation. If the state is not fully measurable, an output-feedback NMPC scheme can be adopted (see e.g. [14]) and the results presented in this paper can be applied without modifications. In this work, it is supposed that the nominal control law κ asymptotically stabilizes the origin of the closed-loop system and that it is continuous over the feasibility set $\mathcal{F}$. These properties respectively depend on the stabilizing constraints (3d) and on the characteristics of the optimization problem (3). For example, the nominal control law κ results to be continuous when the problem (3) is jointly convex in x and U (see e.g. the works [15], [16]). In general, checking a priori the continuity of κ is not an easy task. In practice, continuity of the nominal control law can be usually achieved by using a convex stage cost $L(U, x)$ in (2) and a sufficiently high weight on the control inputs U. On the other hand, it has been recently shown (see [17]) that the knowledge of the discontinuities is needed to achieve an approximated NMPC law with finite and convergent approximation error bound. Due to such a (unavoidable) need of the exact knowledge of discontinuities, in the discontinuous case the approximation problem is more complex than in the continuous one.

B. Stabilizing properties of approximated NMPC laws

In the standard NMPC formulation, the nominal control law κ is evaluated by solving the optimization problem (3) on-line. However, a limitation in the practical use of NMPC is the presence of fast plant dynamics, for which the required sampling time may be too low for real-time optimization. A viable solution to this problem is the use of an approximated control function $\hat{\kappa} \approx \kappa$, derived off-line, whose on-line evaluation is more efficient. It is considered that $\hat{\kappa}$ is defined over a compact set $\mathcal{X}$, containing the origin in its interior:

$$\hat{\kappa} : \mathcal{X} \to \mathbb{R}, \ \mathcal{X} \subseteq \mathcal{F}$$

Moreover, $\hat{\kappa}$ is computed on the basis of the knowledge of a finite number ν of nominal control moves, i.e.:

$$\tilde{u}^k = \kappa(\tilde{x}^k), k = 1, \ldots, \nu \tag{4}$$

where the state values $\tilde{x}^k$ are suitably chosen and define the set:

$$\mathcal{X}_\nu = \{\tilde{x}^k, \, k = 1, \ldots, \nu\} \subseteq \mathcal{F}$$

Note that the points $\tilde{x}^k$, $k = 1, \ldots, \nu$ must be inside the feasibility set $\mathcal{F}$ (in order to be able to compute the corresponding nominal MPC control move $\tilde{u}^k$) but need not to belong to the set $\mathcal{X}$. A major issue arising in the use of approximated NMPC is about the guaranteed closed-loop performance obtained when $\hat{\kappa}$ is employed instead of κ. To this regard, it has been proved (see e.g. [9]) that if $\hat{\kappa}$ has the following key properties:

i) Input constraint satisfaction:

$$\hat{\kappa}(x) \in \mathbb{U} \; \forall x \in \mathcal{X} \tag{5}$$

ii) The pointwise approximation error $\Delta_{\hat{\kappa}}(x) = \kappa(x) - \hat{\kappa}(x)$ is bounded:

$$\|\Delta_{\hat{\kappa}}(x)\| \leq \zeta, \; \forall x \in \mathcal{X} \tag{6}$$

where $\|\cdot\|$ is a suitable norm, e.g. Euclidean.

iii) The bound $\zeta(\nu)$ converges to zero:

$$\lim_{\nu \to \infty} \zeta(\nu) = 0 \tag{7}$$

then it is always possible to explicitly compute a suitable finite value of ν, such that there exist a finite value $\Delta \in \mathbb{R}^+$ with the following characteristics:

i) the distance between the closed loop state trajectories obtained with the nominal and the approximated control laws is bounded by $\Delta(\nu)$, which can be explicitly computed

ii) $\lim_{\nu \to \infty} \Delta(\nu) = 0$

iii) the closed loop state trajectories, obtained when the approximated control law is used, are kept inside the compact set $\mathcal{X}$, and asymptotically converge to an arbitrarily small neighborhood of the origin

As regards the asymptotic behaviour of ζ as $\nu \to \infty$, it is assumed that $\mathcal{X}_\nu$ is chosen such that the following property holds:

$$\lim_{\nu \to \infty} d_H(\mathcal{X}, \mathcal{X}_\nu) = 0 \tag{8}$$

where $d_H(X, X_\nu)$ is defined as:

$$d_H(\mathcal{X}, \mathcal{X}_\nu) \doteq \sup_{x \in \mathcal{X}} \inf_{\tilde{x} \in \mathcal{X}_\nu} \left(\|x - \tilde{x}\|_2\right)$$

The quantity $d_H(X, X_\nu)$ in the described context is equivalent to the Hausdorff distance between the sets $\mathcal{X}$ and $\mathcal{X}_\nu$ (see [18]). Note that uniform gridding over $\mathcal{X}$ satisfies condition

(8).

Properties (5)-(7) imply that there exists a finite and computable bound on the worst-case accuracy obtained with the employed approximated control law, and that such a bound decreases to zero as ν grows. Indeed, several studies in the literature (see e.g. [19] and the references therein) indicate that, if κ is stabilizing and continuous, the possibility to achieve an arbitrarily small approximation error bound ζ is sufficient for closed loop trajectory boundedness and convergence to a neighborhood of the origin. However, larger values of ν, needed to achieve small approximation error bounds, usually correspond to higher memory usage and computational times (both on-line and off-line), leading to the need of choosing a tradeoff between accuracy, computational efficiency and employed memory. In [9], two different SM techniques have been proposed, both satisfying properties (5)-(7), whose performance are complementary. In particular, the optimal (OPT) SM approximation gives the minimal worst-case error, but higher computational complexity with respect to the nearest point (NP) approach, whose guaranteed accuracy is worse but with much more efficient computation. The aim of this paper is to provide new approaches to approximate an NMPC law, able to achieve better tradeoffs between accuracy and complexity. The properties of the OPT and NP techniques are now briefly recalled, as they are instrumental to introduce the main results presented in this work.

III. Set membership approximations of NMPC with guaranteed accuracy

A. *Prior information*

As already pointed out, the approximation of function κ is performed on a compact set $\mathcal{X} \subseteq \mathcal{F}$. Since $\mathcal{X}$ and the image set $\mathbb{U}$ of κ are compact, continuity of κ over $\mathcal{F}$ implies that its components κ_i, $i = 1, \ldots, m$ are Lipschitz continuous functions over $\mathcal{X}$, i.e. there exist finite constants γ_i, $i = 1, \ldots, m$ such that $\forall x^1, x^2 \in \mathcal{X},\ \forall i \in [1, m], |\kappa_i(x^1) - \kappa_i(x^2)| \leq \gamma_i \|x^1 - x^2\|_2$, and $\forall x^1, x^2 \in \mathcal{X}$, $\|\kappa(x^1) - \kappa(x^2)\|_2 \leq \|\gamma\|_2 \, \|x^1 - x^2\|_2$, where $\gamma = [\gamma_1, \ldots, \gamma_m]^T$. This prior information, together with the knowledge of the off-line computed values (4), can be summarized by concluding that

$$\kappa \in FFS, \tag{9}$$

where the set FFS (Feasible Functions Set) is defined as $FFS = \{\kappa : \mathcal{X} \to \mathbb{U}, \kappa = [\kappa_1, \ldots, \kappa_m]^T \ : \ \kappa_i \in FFS_i, \forall i \in [1, m]\}$, with $FFS_i = \{\kappa_i \in \mathcal{A}_{\gamma_i} : \kappa_i(\tilde{x}^k) = \tilde{u}_i^k, \quad k =$

$1, \dots, \nu\}$ and $\mathcal{A}_{\gamma_i}$ being the set of all continuous Lipschitz functions κ_i on $\mathcal{X}$, with constant γ_i. As regards the computation of the Lipschitz constants $\gamma = [\gamma_1, \dots \gamma_m]$, which are needed to apply the presented SM techniques, estimates $\hat{\gamma}_i, i = 1, \dots, m$ can be derived off-line as follows:

$$\hat{\gamma}_i = \inf \left(\tilde{\gamma}_i : \tilde{u}_i^h + \tilde{\gamma}_i \|\tilde{x}^h - \tilde{x}^k\|_2 \geq \tilde{u}_i^k, \ \forall k, h = 1, \dots, \nu \right) \tag{10}$$

Convergence of $\hat{\gamma}_i$ to $\gamma_i, i = 1, \dots, m$ has been showed e.g. in [9], i.e. $\lim\limits_{\nu \to \infty} \hat{\gamma}_i = \gamma_i, \ \forall i = 1, \dots, m$.

B. Optimal SM approximation

With the OPT approach, for a given value of ν the obtained approximating function κ^{OPT} gives the minimal worst-case approximation error.

For each function $\kappa_i, \ i \in [1, m]$ and for a given value $x \in \mathcal{X}$, an approximation $\kappa_i^{\text{OPT}}(x) \approx \kappa_i(x)$ is computed as:

$$\kappa_i^{\text{OPT}}(x) = \tfrac{1}{2}[\overline{\kappa}_i\,(x) + \underline{\kappa}_i\,(x)] \in FFS_i \tag{11}$$

where $\overline{\kappa}_i\,(x) = \sup\limits_{\tilde{\kappa}_i \in FFS_i} \tilde{\kappa}_i\,(x)$ and $\underline{\kappa}_i\,(x) = \inf\limits_{\tilde{\kappa}_i \in FFS_i} \tilde{\kappa}_i\,(x)$, called “optimal bounds”, are the tightest upper and lower bounds of $\kappa_i(x)$, according to the considered prior information. The optimal bounds can be computed as:

$$\begin{aligned} \overline{\kappa}_i(x) &\doteq \min \left[\overline{u}_i, \min_{k=1,\dots,\nu} \left(\tilde{u}_i^k + \gamma_i \|x - \tilde{x}^k\|_2 \right) \right] \in FFS_i \\ \underline{\kappa}_i(x) &\doteq \max \left[\underline{u}_i, \max_{k=1,\dots,\nu} \left(\tilde{u}_i^k - \gamma_i \|x - \tilde{x}^k\|_2 \right) \right] \in FFS_i \end{aligned} \tag{12}$$

Function κ_i^{OPT} (11) is such that the related guaranteed approximation error $E(\kappa_i^{\text{OPT}}) \doteq \sup\limits_{\tilde{\kappa}_i \in FSS_i} \|\tilde{\kappa}_i - \kappa_i^{\text{OPT}}\|_p$ is minimal for any L_p-norm, with $p \in [1, \infty]$, and it is therefore equal to the radius of information $r_{p,i}$ (see e.g. [20]).

Then, the optimal SM approximation $\kappa^{\text{OPT}} \approx \kappa$ is defined as:

$$\kappa^{\text{OPT}} = \left[\kappa_1^{\text{OPT}}, \dots, \kappa_m^{\text{OPT}} \right]^T \tag{13}$$

It has been proved that (see [9]):

$$\kappa^{\text{OPT}} : \mathcal{X} \to \mathbb{U} \tag{14}$$

$$\|\Delta_{\kappa^{\text{OPT}}}(x)\|_2 \doteq \|\kappa(x) - \kappa^{\text{OPT}}(x)\|_2 \leq e^{\text{OPT}}(x) \leq \|r_\infty\|_2 = \zeta^{\text{OPT}}, \ \forall x \in \mathcal{X} \tag{15}$$

$$\lim_{\nu \to \infty} \zeta^{\text{OPT}}(\nu) = 0 \tag{16}$$

where $r_\infty \doteq [r_{\infty,1}, \ldots, r_{\infty,m}]$ and $e^{\text{OPT}}(x) \doteq \|\frac{1}{2}(\overline{\kappa}_i(x) - \underline{\kappa}_i(x))\|_2,\ i \in [1,m]$. Thus, function κ^{OPT} satisfies the key properties (5)-(7). As regards the computation of $r_{\infty,i}, i = 1, \ldots, m$, the results presented in [21] can be employed.

C. Nearest point SM approximation

The NP approach gives an approximating function whose accuracy is not the optimal one, but whose computation is simpler. In particular, for a given value of ν, the NP approximation leads in general to a higher approximation error bound ζ^{NP} than OPT approximation, but to lower on-line computation times, whose growth as a function of ν is much slower than that of OPT approximation. Thus, the NP approximation required to guarantee given stability and performance properties may need much lower on-line computation times with respect to OPT approximation, at the expenses of longer off-line computation time and higher memory usage.

For any $x \in \mathcal{X}$, denote with $\tilde{x}^{\text{NP}} \in \mathcal{X}_\nu$ a state value such that:

$$\|\tilde{x}^{\text{NP}} - x\|_2 = \min_{\tilde{x} \in \mathcal{X}_\nu} \|\tilde{x} - x\|_2. \tag{17}$$

Then, the NP approximation $\kappa^{\text{NP}}(x)$ is computed as:

$$\kappa^{\text{NP}}(x) \doteq \kappa(\tilde{x}^{\text{NP}}) \tag{18}$$

Such approximation trivially satisfies condition (5). Moreover, it has been shown that also the key properties (6), (7) are satisfied:

$$\begin{aligned} &\|\Delta_{\kappa^{\text{NP}}}(x)\|_2 = \|\kappa(x) - \kappa^{\text{NP}}(x)\|_2 \leq \|\gamma\|_2 \|x - \tilde{x}^{\text{NP}}\|_2 = e^{\text{NP}}(x) \leq \\ &\leq \zeta^{\text{NP}} = \|\gamma\|_2 \, d_H(\mathcal{X}, \mathcal{X}_\nu),\ \forall x \in \mathcal{X} \end{aligned} \tag{19}$$

$$\lim_{\nu \to \infty} \zeta^{\text{NP}}(\nu) = 0, \tag{20}$$

and that, for a given value of ν, $\zeta^{\text{OPT}} \leq \zeta^{\text{NP}}$. Thus, the same guaranteed accuracy can be obtained with a lower ν value by OPT approach with respect to NP. However, the evaluation efficiency of NP can be much higher than that of OPT.

The main contribution of this paper is to provide two alternative techniques, giving different compromises between accuracy, memory usage and evaluation complexity with respect to OPT and NP. The accuracy properties of these new approaches will be investigated in the SM framework. Since both the described techniques give a non-minimal worst-case accuracy (as it also happens with the NP approach), they will be referred to as “suboptimal” SM approximations. For simplicity of notation, in the remaining part of the paper it will be assumed that $\kappa : \mathbb{R}^n \to \mathbb{R}$.

IV. SUBOPTIMAL SM APPROXIMATIONS OF NMPC

In this section two suboptimal approximation methods are presented, with improved evaluation efficiency with respect to the OPT technique and improved accuracy with respect to the NP approach, and their accuracy properties are investigated in the framework of SM approximation.

A. Piecewise linear approximation

Let $X_1, X_2, ..., X_q$ be a triangulation defined by the set of points $\mathcal{X}_\nu$. Such a triangulation is a collection of sets $X_1, X_2, ..., X_q$ such that

$$\begin{aligned}
&\bigcup_{j=1}^{q} X_j = \mathrm{chull}(\mathcal{X}_\nu),\\
&\mathrm{int}(X_h) \cap \mathrm{int}(X_j) = 0 \text{ for } h \neq j,\\
&\text{all } X_j\text{'s are simplices (triangles for } n = 2\text{)},\\
&\text{the vertices of the simplices are points of } \mathcal{X}_\nu,\\
&\text{all points of } \mathcal{X}_\nu \text{ are vertices of the simplices.}
\end{aligned}$$

Here $\mathrm{int}(\cdot)$ denotes the interior of a set and $\mathrm{chull}(\cdot)$ denotes the convex hull of a set. A triangulation partitions the convex hull of $\mathcal{X}_\nu$ into a set of simplices, which will be also referred to as “triangles” in the following. For each triangle X_j, consider the set of points

$$\{\tilde{x}^{j,k},\ k = 1, \ldots, n+1 : \tilde{x}^{j,k} \text{ is a vertex of } X_j\}$$

Since a triangle has $n+1$ vertices, such a set contains $n+1$ points in $\mathbb{R}^n$. Let $K_j x + Q_j$ be the linear affine function interpolating these points. The coefficients $K_j \in \mathbb{R}^n,\ Q_j \in \mathbb{R}$ can be trivially obtained as

$$\begin{bmatrix} K_j^T \\ Q_j \end{bmatrix} = \begin{bmatrix} \tilde{x}^{j,1\,T} & 1 \\ \ldots & \ldots \\ \tilde{x}^{j,n+1\,T} & 1 \end{bmatrix}^{-1} \begin{bmatrix} \tilde{u}^{j,1} \\ \ldots \\ \tilde{u}^{j,n+1} \end{bmatrix} \tag{21}$$

where $\tilde{u}^{j,k} = \kappa_i(\tilde{x}^{j,k})$. Assume that $\mathcal{X} \subseteq \mathrm{chull}(\mathcal{X}_\nu)$ and define the piecewise linear approximation (LIN)

$$\kappa^{\mathrm{LIN}}(x) \doteq K_{\hat{j}} x + Q_{\hat{j}} \tag{22}$$

where $\hat{j} \in \arg\min_{j=1,\ldots,q} d(x, X_j)$ and $d(x, X) \doteq \inf_{\xi \in X} (\|x - \xi\|_2)$ is the distance between the point x and the set X. Clearly, for given $x \in \mathcal{X}$, $X_{\hat{j}}$ is a triangle which contains x. If $x \in \mathrm{int}(X_{\hat{j}})$, this triangle is unique. According to the above definition, κ^{LIN} is a continuous piecewise linear function, which can be used to approximate κ. The next result shows that κ^{LIN} satisfies the properties (5)-(7). Define the approximation error:

$$\Delta_{\kappa^{\text{LIN}}}(x) \doteq \kappa(x) - \kappa^{\text{LIN}}(x) \tag{23}$$

Theorem 1: The following properties hold:

i) $\kappa^{\text{LIN}}(x) \in \mathbb{U}, \forall x \in \mathcal{X}$.

ii) The pointwise approximation error $\Delta_{\kappa^{\text{LIN}}}(x)$ of κ^{LIN} is bounded:

$$\begin{aligned} &\forall x \in \mathcal{X},\ |\Delta_{\kappa^{\text{LIN}}}(x)| \leq e^{\text{LIN}}(x) = |\kappa^{\text{OPT}}(x) - \kappa^{\text{LIN}}(x)| + e^{\text{OPT}}(x) \geq e^{\text{OPT}}(x) \\ &\Rightarrow \forall x \in \mathcal{X},\ |\Delta_{\kappa^{\text{LIN}}}(x)| \leq \zeta^{\text{LIN}} = \sup_{x \in \mathcal{X}} e^{\text{LIN}}(x) \geq \zeta^{\text{OPT}} \end{aligned}$$

iii) $\lim_{\nu \to \infty} \zeta^{\text{LIN}}(\nu) = 0$.

Proof: See the appendix. ■

B. SM Neighborhood approximation

Let $X_1, X_2, ..., X_q$ be a collection of sets such that

$$\mathcal{X} \subseteq \bigcup_{j=1}^{q} X_j.$$

For any $x \in \mathcal{X}$, let $\hat{j} \in \arg \min_{j=1,\dots,q} d(x, X_j)$, so that $X_{\hat{j}}$ contains x. Define the sets of indices

$$P_j \doteq \left\{k : \tilde{x}^k \in \left(X_j \bigcap \mathcal{X}_\nu\right) \cup \{\tilde{x}^{\text{NP}}\}\right\},\ j = 1, \dots, q, \tag{24}$$

where $\tilde{x}^{\text{NP}}$ is defined by (17). The SM neighborhood (NB) approximation of κ is given by:

$$\kappa^{\text{NB}}(x) \doteq \frac{1}{2}[\overline{\kappa}^{\text{NB}}(x) + \underline{\kappa}^{\text{NB}}(x)] \tag{25}$$

with

$$\begin{aligned} \overline{\kappa}^{\text{NB}}(x) &\doteq \min\left[\overline{u}, \min_{k \in P_{\hat{j}}} \left(\tilde{u}^k + \gamma \|x - \tilde{x}^k\|_2\right)\right] \\ \underline{\kappa}^{\text{NB}}(x) &\doteq \max\left[\underline{u}, \max_{k \in P_{\hat{j}}} \left(\tilde{u}^k - \gamma \|x - \tilde{x}^k\|_2\right)\right]. \end{aligned} \tag{26}$$

Note that the function κ^{NB} is defined similarly to κ^{OPT}, except that only a subset of points of $\mathcal{X}_\nu$ is used to compute $\overline{\kappa}^{\text{NB}}(x)$ and $\underline{\kappa}^{\text{NB}}(x)$. γ in (26) is selected on the basis of the estimates (10). In the Set Membership context, the estimates (10) guarantee that the prior assumptions on function κ are not invalidated by the available data. If γ is increased, the obtained approximation tends to coincide with the Nearest Point technique.

In order to show that κ^{NB} satisfies the properties (5)-(7) and to evaluate its accuracy, let us define the indices

$$\begin{aligned}
\overline{k} &\doteq \arg \min_{k=1,\ldots,\nu} (\tilde{u}^k + \gamma \|x - \tilde{x}^k\|_2) \\
\underline{k} &\doteq \arg \max_{k=1,\ldots,\nu} (\tilde{u}^k - \gamma \|x - \tilde{x}^k\|_2) \\
\overline{j} &\doteq \arg \min_{k \in P_{\hat{j}}} \left(\tilde{u}^k + \gamma \|x - \tilde{x}^k\|_2\right) \\
\underline{j} &\doteq \arg \max_{k \in P_{\hat{j}}} \left(\tilde{u}^k - \gamma \|x - \tilde{x}^k\|_2\right)
\end{aligned}$$

Moreover, define the following scalar quantities:

$$\delta(x) = \gamma(\|\tilde{x}^{\overline{k}} - \tilde{x}^{\overline{j}}\|_2 + \|\tilde{x}^{\underline{k}} - \tilde{x}^{\underline{j}}\|_2) \tag{27}$$

$$\Delta_{\kappa^{\text{NB}}}(x) \doteq \kappa(x) - \kappa^{\text{NB}}(x)$$

Theorem 2: The following properties hold:

i) $\kappa^{\text{NB}}(x) \in \mathbb{U}$, $\forall x \in \mathcal{X}$.

ii) The pointwise approximation error $\Delta_{\kappa^{\text{NB}}}(x)$ of κ^{NB} is bounded:

$$\begin{aligned}
&\forall x \in \mathcal{X},\ |\Delta_{\kappa^{\text{NB}}}(x)| \le e^{\text{NB}}(x) \doteq \min(e^{\text{NP}}, e^{\text{OPT}}(x) + \delta(x)) \\
&\Rightarrow \forall x \in \mathcal{X},\ |\Delta_{\kappa^{\text{NB}}}(x)| \le \zeta^{\text{NB}} = \sup_{x \in \mathcal{X}} e^{\text{NB}} \le \zeta^{\text{NP}}
\end{aligned}$$

iii) The following convergence property holds:

$$\lim_{\nu \to \infty} \zeta^{\text{NB}}(\nu) = 0.$$

iv) If $\overline{k} = \overline{j}$ and $\underline{k} = \underline{j}$ then

$$\kappa^{\text{NB}}(x) = \kappa^{\text{OPT}}(x).$$

Proof: See the appendix. ■

Remark 1: For a given number of data ν, under suitable choices of the sets $X_1, X_2, ..., X_q$ and using efficient search algorithms, both LIN and NB approximations lead to a significantly better on-line computational efficiency than the OPT approximation, at the expense of higher memory usage and some degradation of the worst case approximation error. For example, a technique that gives dramatically better computational efficiency is to organize off-line the sets X_1, ..., X_q using a uniform gridding of $\mathcal{X}$. Then, a very efficient on-line search for the index $\hat{j}$ corresponding to the active set can be performed. Note that, in practical application, a higher worst-case accuracy does not necessarily imply that the performance of the suboptimal techniques are worse than those of the optimal one. These aspects will be highlighted by the numerical example of Section V-A.

V. EXAMPLES

A. Numerical example

Consider the following two-dimensional continuous-time nonlinear system (see e.g. [22])

$$\begin{cases} \dot{x}_1(t) = x_2(t) + \dfrac{(1 + x_1(t))}{2} u(t) \\ \dot{x}_2(t) = x_1(t) + \dfrac{(1 - 4x_2(t))}{2} u(t) \end{cases} \tag{28}$$

whose origin is an unstable equilibrium point. The input constraint set $\mathbb{U}$ is:

$$\mathbb{U} = \{u \in \mathbb{R} : |u| \leq 4\}$$

The following discrete time model, to be used in the nominal MPC design, has been obtained by forward difference approximation:

$$x_{t+1} = \begin{bmatrix} 1 & T_s \\ T_s & 1 \end{bmatrix} x_t + \frac{T_s}{2} \left(\begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & -4 \end{bmatrix} x_t \right) u_t$$

with sampling time $T_s = 0.1\,\mathrm{s}$. The nominal NMPC controller κ is designed according to (3) with horizons $N_p = 30$, $N_c = 30$ and the following functions L and Φ:

$$L(x, u) = x^T Q x + u^T R u, \quad \Phi = 0$$

where

$$Q = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}, \quad R = 0.5$$

The following linear state inequality constraints define the considered set $\mathbb{X}$:

$$\mathbb{X} = \{x \in \mathbb{R}^2 : \|x\|_\infty \leq 3\}$$

Moreover, the following terminal constraint set (see e.g. [1]) has been included to enforce stability of the origin of the nominal discrete-time model:

$$\mathbb{X}_\mathrm{f} = \{x \in \mathbb{R}^2 : \|x\|_\infty \leq 0.1\}$$

The origin of the closed-loop system with the linear control law $u_t = -K_{\mathrm{LQR}} x_t$, $K_{\mathrm{LQR}} = [2.1,\ 2.1]$ is asymptotically stable for any initial state $x_0 \in \mathbb{X}_\mathrm{f}$.

The optimization problem (3), whose solution defines the control law $\kappa(x)$, has been solved using a sequential constrained Gauss-Newton quadratic programming algorithm (see e.g. [23]), where the underlying quadratic programs have been solved using the MatLab® function `quadprog`. The mean computational time of the on-line optimization was between $1\,\mathrm{s}$ and $8\,10^{-2}\,\mathrm{s}$ (depending on the actual state value x_t) with a mean value of $1.7\,10^{-1}\,\mathrm{s}$, using

MATLAB® 7 with an Intel® Core™2 Duo @2.4 GHz processor and 2 GB RAM.

Fig. 1(a) shows the set $\mathcal{X}$ considered for the approximation, together with the constraint set $\mathbb{X}$. The level curves of the optimal cost function

$$J^*(x) = \min_U J(U, x)$$

are reported too. The following approximating functions have been considered:

i) Optimal SM approximation κ^{OPT}

ii) Nearest point approximation κ^{NP}

iii) Neighborhood SM approximation κ^{NB}, with partitions X_j computed by employing a uniform grid on the set $\mathcal{X}$, in order to achieve fast on-line computational times.

iv) Linear interpolation κ^{LIN}, with partitions X_j computed applying the Delaunay triangulation (see e.g. [24]) to the set $\mathcal{X}_\nu$

Each of the considered approximations has been computed using different values of ν. An example of the simulated closed-loop state trajectories obtained with $\nu = 2.5\,10^3$ and initial condition $x(0) = [2.1,\ -17]^T$ is reported in Figs. 1(b) and 2. It can be noted that the closed-loop trajectories are practically superimposed (see Fig. 1(b)), except for a quite small neighborhood of the origin (see Fig. 2). In particular, it can be noted that control laws κ and κ^{LIN} obtain no steady-state offset, as it can be expected since in the neighborhood of the origin both these controllers are equal to an asymptotically stabilizing linear state feedback law. On the contrary, the SM optimal, nearest point and neighborhood approximations make the system state converge to an equilibrium point close to the origin. Such a behaviour, which is confirmed by the results of extensive simulation tests reported in Table I below, is due to the fact that the origin is an unstable equilibrium point and that both κ^{OPT} and κ^{NB} are equal to zero in its proximity (provided that the equilibrium point $\tilde{x} = [0,\ 0]^T,\ \tilde{u} = 0$ is included in the off-line computed data set $\mathcal{X}_\nu$). The regulation precision obtained with the OPT and NB laws can be improved by using a higher number of off-line computed points near the origin, making the state converge to an arbitrary small neighborhood of $[0,\ 0]^T$. Alternatively, a dual-mode controller could be used, switching to a linear stabilizing state feedback control law when the system state enters the related reachable set (or a subset of it).

To evaluate the performance and computational times of the considered control laws, 500 simulations have been performed starting from different initial conditions, chosen with uniform random gridding over $\mathcal{X}$. Each simulation lasted 300 time steps (i.e. 30 simulation seconds). The mean computational times $\bar{t}$, over all time steps of all simulations, obtained with each controller, are reported in Table II. As a measure of control system performance, the relative Euclidean distance $d^r(t)$, $r = 1, \ldots, 500$ has been considered:

$$d^r(t) = \frac{\|\phi^r(t) - \hat{\phi}^r(t)\|_2}{\|\phi^r(t)\|_2}$$

where $\phi^r(t)$ and $\hat{\phi}^r(t)$ are the closed-loop state trajectories obtained in the r-th simulation with the nominal controller and the approximated one respectively, given the same initial state $x^r(0)$. Then, the following definition of transient interval has been considered:

$$t^{\text{TI},r} = \arg\min_t t : \|\hat{\phi}^r(t)\|_2 \leq 0.1\,\|x^r(0)\|_2$$

and the mean relative distance $\overline{d}$ over the time intervals [0,$t^{\text{TI},r}$] of all the simulations has been computed:

$$\overline{d} = \frac{1}{500}\sum_{r=1}^{500}\left(\frac{1}{t^{\text{TI},r}}\int_0^{t^{\text{TI},r}} d^r(t)\,dt\right)$$

Moreover, as a measure of regulation precision, the mean value d^{OR} of the norm of the state trajectory $\|\hat{\phi}^r(t)\|_2$, $r = 1, \ldots, 500$ over the last 2 seconds of all the simulations has been also evaluated:

$$d^{\text{OR}} = \frac{1}{500}\sum_{r=1}^{500}\left(\frac{1}{2}\int_{28}^{30} \|\hat{\phi}^r(t)\|_2\,dt\right)$$

The values of $\overline{d}$ and d^{OR} obtained with each approximated controller are given in Tables I and III respectively. Finally, Table IV shows the memory required by each of the approximated control laws for each value of ν. Indeed, the reported computational times and memory requirements are intended to be used to compare the different control laws in relative terms only. No memory optimization effort has been done on the employed data structures and all the variables have been stored using 4-Byte floating point representation.

TABLE I
NUMERICAL EXAMPLE: MEAN REGULATION PRECISION d^{OR}.

ν	κ^{OPT}	κ^{NP}	κ^{NB}	κ^{LIN}
$2.5\,10^3$	$6.0\,10^{-3}$	$6.0\,10^{-3}$	$6.0\,10^{-3}$	$\mathbf{2\,10^{-13}}$
$4.9\,10^3$	$4.4\,10^{-9}$	$4.4\,10^{-9}$	$4.4\,10^{-9}$	$\mathbf{2\,10^{-13}}$
$9.7\,10^3$	$4.4\,10^{-9}$	$4.4\,10^{-9}$	$4.4\,10^{-9}$	$\mathbf{2\,10^{-13}}$
$2.5\,10^4$	$4.4\,10^{-9}$	$4.4\,10^{-9}$	$4.4\,10^{-9}$	$\mathbf{2\,10^{-13}}$

TABLE II

NUMERICAL EXAMPLE: MEAN COMPUTATIONAL TIMES.

ν	κ^{OPT}	κ^{NP}	κ^{NB}	κ^{LIN}
$2.5\,10^3$	$3.3\,10^{-4}$	**$9.0\,10^{-5}$**	$1.3\,10^{-4}$	$3.8\,10^{-4}$
$4.9\,10^3$	$1.0\,10^{-3}$	**$1.0\,10^{-4}$**	$1.5\,10^{-4}$	$5.9\,10^{-4}$
$9.7\,10^3$	$2.0\,10^{-3}$	**$1.1\,10^{-4}$**	$1.7\,10^{-4}$	$8.1\,10^{-4}$
$2.5\,10^4$	$5.0\,10^{-3}$	**$7.2\,10^{-5}$**	$1.9\,10^{-4}$	$7.0\,10^{-4}$

TABLE III

NUMERICAL EXAMPLE: MEAN TRAJECTORY DISTANCE $\overline{d}$.

ν	κ^{OPT}	κ^{NP}	κ^{NB}	κ^{LIN}
$2.5\,10^3$	7.8%	8.6%	5.9%	**1.6%**
$4.9\,10^3$	2.5%	3.0%	2.7%	**0.7%**
$9.7\,10^3$	1.5%	1.9%	1.5%	**0.2%**
$2.5\,10^4$	1.1%	1.7%	1.3%	**0.1%**

TABLE IV

NUMERICAL EXAMPLE: MEMORY USAGE (KB)

ν	κ^{OPT}	κ^{NP}	κ^{NB}	κ^{LIN}
$2.5\,10^3$	**$0.6\,10^2$**	$0.9\,10^2$	$0.7\,10^2$	$3.0\,10^2$
$4.9\,10^3$	**$1.2\,10^2$**	$1.6\,10^2$	$1.3\,10^2$	$7.0\,10^2$
$9.7\,10^3$	**$2.3\,10^2$**	$3.6\,10^2$	$2.8\,10^2$	$1.5\,10^3$
$2.5\,10^4$	**$6.0\,10^2$**	$1.3\,10^3$	$7.5\,10^2$	$4.3\,10^3$

From Table II it can be noted that the NP approximation κ^{NP} achieves the lowest computational values, however its performance (Table III) is also the worst (though quite close to those of OPT and NB approximations) and the memory occupation is high (only LIN technique has higher memory requirements). Function κ^{OPT} has better precision and the lowest memory usage, but also the highest computational times. The best performance is obtained for any ν value by the linear interpolation κ^{LIN}, at the cost of higher computational time (but still about 250-500 times lower than on-line optimization) and memory usage. In particular, with $\nu = 2.5\,10^3$ points the linear interpolation achieves better performance than the other techniques in most cases, together with asymptotic stability of the origin. Note

that the optimal SM approximation has worse performance than LIN technique: this doesn't contradict the theoretical results since the OPT approximation guarantees the lowest worst case error, which doesn't imply that the average precision in practice is the best. This is also the reason why in some cases (see Table III for the case $\nu = 2.5\,10^3$) the NB technique (which employs only a subset of the data considered by the OPT approximation) has better average performance than OPT. In fact, the SM neighborhood approximation has performance close to those of OPT and quite fast computational times (slower than the NP technique only, see Table II). This is also put into evidence by the fact that in most cases (93% with $\nu = 2.5\,10^3$ up to 96% with $\nu = 2.5\,10^4$) the input $\kappa^{\text{NB}}(x) = \kappa^{\text{OPT}}(x)$ given the same x value.
Thus, the presented example shows how both LIN and NB techniques can be tuned to achieve a suitable tradeoff between precision, on-line evaluation time, memory usage and off-line computation, providing more degrees of freedom in the control design than the previously introduced OPT and NP approaches.

B. Case study: vehicle stability control

In order to give an example of higher order and practical relevance of the application of the presented SM approximation techniques, an automotive case study is presented here. For more details on this application, the interested reader is referred to [25], [26]. A car equipped with a Rear Active Differential (RAD) is considered. Such a device can be used to split the traction torques acting on the left and right rear wheels with different ratios than the 50%-50% imposed by standard (passive) differentials. This way, an equivalent yaw moment applied to the car center of gravity can be generated, able to influence the vehicle lateral dynamics in order to improve vehicle maneuverability and to assist the driver in critical situations and harsh maneuvers. The lateral vehicle dynamics can be described using the single track model reported in Fig. 3, whose equations are the following (see e.g. [27]):

$$\begin{array}{l} 1715v(\tau)\dot{\beta}(\tau) + 1715v(\tau)\dot{\psi}(\tau) = F_{yf}(\tau) + F_{yr}(\tau) \\ 2700\ddot{\psi}(\tau) = 1.07F_{yf}(\tau) - 1.47F_{yr}(\tau) + M_z(\tau) \end{array} , \tag{29}$$

where τ is the continuous time variable. In model (29), M_z is the yaw moment issued by the RAD and δ is the front steering angle commanded by the driver through the handwheel. Moreover, β is the sideslip angle, ψ is the yaw angle and v is the vehicle speed. F_{yf} and F_{yr} are the front and rear tyre lateral forces, which can be expressed as nonlinear functions of the other variables (see e.g. [28] for more details):

$$\begin{aligned} F_{yf} &= F_{yf}(\beta, \dot{\psi}, v, \delta) \\ F_{yr} &= F_{yr}(\beta, \dot{\psi}, v, \delta) \end{aligned} \tag{30}$$

The yaw moment given by the RAD depends on the current $i(\tau)$ originated by the control algorithm (see [26] for a detailed description of the RAD device). As a first approximation, the actuator behavior can be described as:

$$M_z(\tau) = 2500 i(\tau - 0.02) \tag{31}$$

Equations (29), (30) and (31) can be rearranged in the state equation form:

$$\begin{bmatrix} \ddot{\psi}(\tau) \\ \dot{\beta}(\tau) \end{bmatrix} = f(\dot{\psi}(\tau), \beta(\tau), v(\tau)\delta(\tau), i(\tau - 0.02)) \tag{32}$$

The input variable $i(\tau)$ is employed for control purposes, while $\delta(\tau)$ is not manipulable and describes the driver's maneuvering intention and $v(\tau)$ is a measured model parameter. The control requirements in terms of understeer characteristics improvements can be taken into account by a suitable choice of a reference signal $\dot{\psi}_{\text{ref}}(\tau)$ generated on the basis of the current values of the steering angle $\delta(\tau)$ and of the vehicle speed $v(\tau)$. Details on the computation of the reference yaw rate can be found in [26]. In order to take into account such reference following requirements, the NMPC cost function can be designed in order to minimize the amount of the error variable $e(\tau)$:

$$e(\tau) = \dot{\psi}_{\text{ref}}(\tau) - \dot{\psi}(\tau)$$

The NMPC move computation is performed at discrete time instants $t\,T_s$, $t \in \mathbb{N}$, defined by the sampling period $T_s = 0.01\,\text{s}$ and on the basis of the following state equations obtained by discretization of (32), e.g. by means of forward difference approximation (for simplicity, the notation $t + l \triangleq (t + l)T_s$ will be used):

$$\begin{bmatrix} \dot{\psi}_{t+1} \\ \beta_{t+1} \end{bmatrix} = \tilde{f}(\dot{\psi}_t, \beta_t, v_t, \delta_t, i_{t-2}) \tag{33}$$

At each sampling time t, the measured values of the state $\dot{\psi}_t, \beta_t$ and of the speed v_t, together with the requested value of the yaw rate reference $\dot{\psi}_{\text{ref},t}$, and of the input variables $\delta_t, i_{t-1}, \ldots, i_{t-2}$ are used to compute the control move through the optimization of the following performance index:

$$J_v = \sum_{l=0}^{N_p-1} e^2_{t+l+1|t} + 10^{-6}\, i^2_{t+l|t}, \tag{34}$$

where $N_p = 100$, $e_{t+l|t}$ is the l^{th} step ahead prediction of the error variable obtained as

$$e_{t+l|t} \triangleq \dot{\psi}_{\text{ref},t} - \dot{\psi}_{t+l|t}.$$

As anticipated, the value of $\dot{\psi}_{\text{ref},t}$ is computed using the current values of δ_t and v_t. The predicted yaw rate $\dot{\psi}_{t+l|t}$ is obtained via the state equation (33), starting from the "initial condition":

$$\begin{bmatrix} \dot{\psi}_t \\ \beta_t \end{bmatrix}$$

and using the following values of the inputs i and δ:

$$\begin{bmatrix} \delta_{t|t} = \delta_{t+1|t} = \ldots = \delta_{t+99|t} \\ i_{t-2}, \ldots, i_{t-1}, i_{t|t}, \ldots, i_{t+4|t}, \ldots, i_{t+99|t} \end{bmatrix}$$

where the control moves $i_{t|t}, \ldots i_{t+4|t}$ are optimization variables (i.e. the control horizon is equal to 5 steps) and $i_{t+l|t} = i_{t+4|t}$, $l = 5, \ldots, 99$. Thus, since during the prediction horizon the value of the steering angle δ is kept constant at the value $\delta_{t|t}$ measured at time instant t, the optimization of the index (34) is performed with respect to the variables $I = [i_{t|t}, \ldots, i_{t+4|t}]^T$. Therefore the performance index J_v depends on the vector $w_t \in \mathbb{R}^6$ of the measured variables:

$$w_t \triangleq \left[\dot{\psi}_t, \beta_t, \delta_t, v_t, i_{t-2}, \ldots, i_{t-1}\right]^T \tag{35}$$

Moreover, the amount of the yaw moment generated by the employed active device is subject to its physical limits. In particular, the considered device has an input current limitation of ± 1 A which correspond to the range of allowed yaw moment of ±2500 Nm that can be mechanically generated. Therefore, saturation aspects of the control input (i.e. the actuator current $i(\tau)$) have to be carefully taken into account in the control design. Thus, the optimization problem (3) considered in the nominal NMPC law is the following:

$$\min_{I} \quad J_v \tag{36a}$$

subject to

$$I \in \mathbb{I} = \left\{ i_{t|t} \,:\, |i_{t+4|t}| \leq 1 \right\} \tag{36b}$$

The obtained predictive controller results to be a nonlinear static function of the variable w_t defined in (35):

$$i_t = \kappa_v(w_t) \tag{37}$$

For a given w_t, the value of the function $\kappa_v(w_t)$ is computed by solving at each sampling time t the constrained optimization problem (36). However, such online optimization problem can not be solved at the sampling period required for this application (i.e. 0.01 s). This issue has been tackled in [25] by using the NP approximation method recalled in Section III-C. Here, the NB approach introduced in Section IV-B is applied to the same problem and the obtained results are compared to those of the NP technique. A number $\nu = 2.28\,10^4$ of nominal control values $\tilde{i} = \kappa_v(\tilde{w})$ and of the respective values of $\tilde{w}$ has been collected off-line inside an hypercubic set $\mathcal{W}$, defined by the following inequalities:

$$\mathcal{W} = \left\{ w : \begin{bmatrix} -0.5 \\ -0.1 \\ -0.1 \\ 22 \\ -1 \\ -1 \end{bmatrix} \preceq w \preceq \begin{bmatrix} 0.5 \\ 0.1 \\ 0.1 \\ 33 \\ 1 \\ 1 \end{bmatrix} \right\}.$$

Such a set has been chosen through extensive closed loop simulations of harsh maneuvers using the nominal control law. Then, the set $\mathcal{W}_\nu$ of data collected off-line has been used to derive the NP and NB approximations. The latter has been designed by partitioning the set $\mathcal{W}_\nu$ into sets W_j, $j = 1, \ldots, 4$ defined on the basis of the value of v_t (i.e. the fourth component of the variable w_t, see (35)) as follows:

$$\begin{array}{l} W_1 = \{w : 22 \leq v \leq 24.75\} \\ W_2 = \{w : 24.75 < v \leq 27.5\} \\ W_3 = \{w : 27.5 < v \leq 30.25\} \\ W_4 = \{w : 30.25 < v \leq 33\} \end{array}. \tag{38}$$

This way, at each sampling time the active set $W_{\hat{j}}$, needed to compute on-line the NB approximation (25), can be found on the basis of a simple search based on the value of v_t. The memory requirement of the NP approximation is equal to 160 KB (i.e. $2.28\,10^4$ vectors $\tilde{w}$ of 6 Byte each, plus $2.28\,10^4$ values of the corresponding control inputs $\tilde{u}$ of 1 Byte each), while 165 KB are required for the NB approach (i.e. the same as the NP, plus the vehicle speed values that delimitate the intervals in (38)). Fig. 4(a) shows the performance obtained by the two considered approximated NMPC laws and by the uncontrolled vehicle during a step steer maneuver performed at 100 km/h with a handwheel amplitude of 50°. It can be noted that both the NB and NP approaches are able to improve the vehicle performance with respect to the uncontrolled case; the NB approach achieves a better steady-state accuracy than NP, as well as a more damped transient behavior. Such a better accuracy is obtained at the expense of a higher on-line computational time: $0.004\,\mathrm{s}$ are needed by the NB approach with respect to the $5\,10^{-5}\,\mathrm{s}$ achieved the NP technique. These computational times, obtained using MatLab® 7 under MS Windows XP and an Intel® Core(tm)2 Duo T7700@2.4 GHz processor with 2 GB RAM, are both lower than the employed sampling time of $0.01\,\mathrm{s}$. Thus, given its better performance, in the presented case the NB approach should be preferred with respect to the NP approximation. Indeed, since a present automotive Electronic Control Unit (ECU) has lower computational capacity than the computer employed in this example, a different tradeoff (or the use of the NP approach, which is less computationally expensive) may be needed with the NB technique to allow the implementation on a real ECU. Fig. 4(b) shows the courses of the control input obtained during the step steer maneuver: it can be noted that both the NB and NP approaches satisfy input constraints, as expected from the theoretical results. As a final remark, note that in this example the nominal control law is a static function not only of the state $x_t = [\dot{\psi}_t, \beta_t]^T$, but also of past control input values and of other inputs and parameters like the front steering angle δ_t and the vehicle speed v_t: thus, this example also shows how the presented theoretical results, which have been formulated for the case of a state feedback control law, can be straightforwardly applied also to the case of NMPC laws that depend on other variables and parameters, as long as the control law is a static function of all its arguments.

VI. Conclusions

The linear interpolation (LIN) and Set Membership (SM) neighborhood (NB) techniques have been presented, with the aim of approximating a given predictive control law in the framework of SM theory. The newly introduced techniques have been also compared with the optimal (OPT) and with the Nearest Point (NP) SM approaches described in previous works. All these techniques rely on the off-line computation of a finite number ν of nominal NMPC control moves. A numerical example has been employed to show that the various approaches can be tuned to achieve a suitable tradeoff between on-line efficiency, accuracy, memory usage and off-line computation. It has been shown that, for a given value of ν, the LIN approach gives the highest accuracy in practice, at the expense of higher memory usage. Thus, the LIN technique should be used when high memory is available. The NB approach is practically less accurate than LIN but it achieves a guaranteed error and computational time between those of the OPT technique and of the NP approach, with lower memory requirements than LIN, so that it can be tuned to adapt to the available memory and computational power.

Finally, an automotive case study where the nominal NMPC law depends on 6 variables has been briefly described, in order to show the applicability of the presented approaches also to problems of higher order and practical relevance. It has to be noted that the proposed approach can be effectively used in applications where the nominal NMPC law depends on up to 8-10 variables.

REFERENCES

[1] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, pp. 789–814, 2000.

[2] M. Diehl, H. Bock, and J. Schlöder, "A real-time iteration scheme for nonlinear optimization in optimal feedback control," *SIAM J. Control Optim.*, vol. 43, no. 5, pp. 1714–1736, 2005.

[3] M. Alamir, *Stabilization of Nonlinear Systems by Receding-Horizon Control Schemes: Parametrized Approach for Fast Systems*. Springer, 2006.

[4] T. Parisini and R. Zoppoli, "A receding-horizon regulator for nonlinear systems and a neural approximation," *Automatica*, vol. 31, no. 10, pp. 1443–1451, 1995.

[5] T. Johansen, "Approximate explicit receding horizon control of constrained nonlinear systems," *Automatica*, vol. 40, pp. 293–300, 2004.

[6] A. Grancharova and T. A. Johansen, "Computation, approximation and stability of explicit min–max nonlinear model predictive control," *Automatica*, vol. 45, no. 5, pp. 1134–1143, 2009.

[7] D. Muñoz de la Peña, A. Bemporad, and C. Filippi, "Robust explicit MPC based on approximate multiparametric convex programming," *IEEE Transactions on Automatic Control*, vol. 51, no. 8, pp. 1399–1403, August 2006.

[8] A. Ulbig, S. Olaru, D. Dumur, and P. Boucher, "Explicit solutions for nonlinear model predictive control: A linear mapping approach," in *9^{th} European Control Conference*, 2007, pp. 3295–3302.

[9] M. Canale, L. Fagiano, and M. Milanese, "Set Membership approximation theory for fast implementation of model predictive control laws," *Automatica*, vol. 45, no. 1, pp. 45–54, 2009, doi: 10.1016/j.automatica.2008.06.015.

[10] A. Bemporad, M. Morari, V. Dua, and E. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, pp. 3–20, 2002.

[11] M. Seron, G. Goodwin, and J. D. Doná, "Characterization of receding horizon control for constrained linear systems," *Asian Journal of Control*, vol. 5, no. 2, pp. 271–286, 2003.

[12] M. Canale, L. Fagiano, and M. Milanese, "High altitude wind energy generation using controlled power kites," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 2, pp. 279–293, 2010.

[13] M. Canale, M. Milanese, and C. Novara, "Semi-active suspension control using "fast" model-predictive techniques," *IEEE Transactions on Control System Technology*, vol. 14, no. 6, pp. 1034–1046, November 2006.

[14] L. Magni, G. D. Nicolao, and R. Scattolini, "On the stabilization of nonlinear discrete-time systems with output feedback," *International Journal of Robust and Nonlinear Control*, vol. 14, pp. 1379–1391, 2004.

[15] J. Spjøtvold, P. Tøndel, and T. A. Johansen, "Continuous selection and unique polyhedral representation of solutions to convex parametric quadratic programs," *Journal of Optimization Theory and Applications*, vol. 134, pp. 177–189, 2007.

[16] M. Baes, M. Diehl, and I. Necoara, "Every continuous nonlinear control system can be obtained by parametric convex programming," *IEEE Transactions on Automatic Control*, vol. 53, no. 8, pp. 1963–1967, 2008.

[17] L. Fagiano, M. Canale, and M. Milanese, "Set membership approximation of discontinuous nmpc laws," in *$48^t h$ IEEE Conference on Decision and Control*, Shanghai, China, December 2009, pp. 8636–8641.

[18] C. S. Blagovest, *Hausdorff Approximations*. Springer, 1990.

[19] G. Grimm, M. J. Messina, S. E. Tuna, and A. R. Teel, "Examples when nonlinear model predictive control is nonrobust," *Automatica*, vol. 40, no. 5, pp. 1729–1738, 2004.

[20] J. Traub and H. Woźniakowski, *A General Theory of Optimal Algorithms*. New York: Academic Press, 1980.

[21] M. Milanese and C. Novara, "Computation of local radius of information in SM–IBC identification of nonlinear systems," *Journal of Complexity*, vol. 4, no. 6, pp. 937–951, August 2007.

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

[23] J. Nocedal and S. Wright, *Numerical Optimization*. Springer, 2006.

[24] C. B. Barber, D. Dobkin, and H. Huhdanpaa, "The quickhull algorithm for convex hulls," *ACM Transactions on Mathematical Software*, vol. 22, no. 4, pp. 469–483, 1996.

[25] M. Canale and L. Fagiano, "Vehicle yaw control using a fast NMPC approach," in *47th IEEE Conference on Decision and Control*, Cancun, Mexico, 2008, pp. 5360–5365.

[26] M. Canale, L. Fagiano, M. Milanese, and P. Borodani, "Robust vehicle yaw control using an active differential and IMC techniques," *Control Engineering Practice*, vol. 15, pp. 923–941, 2007.

[27] R. Rajamani, *Vehicle Dynamics and Control*. Springer Verlag, 2005.

[28] E. Bakker, L. Lidner, and H. Pacejka, "A new tyre model with an application in vehicle dynamics studies," in *SAE Paper 890087*, 1989.

APPENDIX

Proof of Theorem 1

i) For any $x \in \mathcal{X}$, consider the the vertices $\tilde{x}^l$, $l = 1, \ldots, n+1$ of the partition $X_{\hat{j}}$: $\hat{j} \in \arg \min\limits_{j=1,\ldots,q} d(x, X_j)$, and the corresponding nominal control moves $\tilde{u}^l = \kappa(\tilde{x}^l)$. Note that $\tilde{u}^l = \kappa^{\text{LIN}}(\tilde{x}^l)$ by definition (21). The point x can be expressed as:

$$x = \sum_{l=1}^{n+1} w_l \, \tilde{x}^l, \; w_l > 0 \, \forall l \in [1, n+1], \; \sum_{l=1}^{n+1} w_l = 1$$

and the approximated control move $\kappa^{\text{LIN}}(x)$ can be therefore computed as:

$$\begin{aligned} \kappa^{\text{LIN}}(x) &= \sum_{l=1}^{n+1} w_l \left(K_j \tilde{x}^l + Q_j \right) = \\ &= \sum_{l=1}^{n+1} w_l \, \kappa^{\text{LIN}}(\tilde{x}^l) = \sum_{l=1}^{n+1} w_l \, \tilde{u}^l \end{aligned}$$

thus it can be noted that:

$$\begin{aligned} \kappa^{\text{LIN}}(x) &\leq \max_{l=1,\ldots,n+1} (\tilde{u}^l) \sum_{l=1}^{n+1} w_l = \max_{l=1,\ldots,n+1} (\tilde{u}^l) \leq \overline{u} \\ \kappa^{\text{LIN}}(x) &\geq \min_{l=1,\ldots,n+1} (\tilde{u}^l) \sum_{l=1}^{n+1} w_l = \min_{l=1,\ldots,n+1} (\tilde{u}^l) \geq \underline{u} \\ &\Rightarrow \kappa^{\text{LIN}}(x) \in \mathbb{U} \end{aligned}$$

ii) $|\Delta_{\kappa^{\text{LIN}}}(x)| = |\kappa(x) - \kappa^{\text{LIN}}(x)| = |\kappa^{\text{OPT}}(x) + \Delta_{\kappa^{\text{OPT}}}(x) - \kappa^{\text{LIN}}(x)| \leq |\kappa^{\text{OPT}}(x) - \kappa^{\text{LIN}}(x)| + |\Delta_{\kappa^{\text{OPT}}}(x)| \leq |\kappa^{\text{OPT}}(x) - \kappa^{\text{LIN}}(x)| + e^{\text{OPT}}(x) = e^{\text{LIN}}(x) \geq e^{\text{OPT}}(x)$

$e^{\text{LIN}}(x) \leq \sup_{x\in\mathcal{X}} e^{\text{LIN}}(x) \doteq \zeta^{\text{LIN}} \geq \sup_{x\in\mathcal{X}} e^{\text{OPT}}(x) = \zeta^{\text{OPT}}$

iii) Considering that $\lim_{\nu\to\infty} e^{\text{OPT}}(x) = 0$, $\forall x \in \mathcal{X}$ (i.e. $\lim_{\nu\to\infty} \kappa^{\text{OPT}}(x) = \kappa(x)$, see [9]) and that, since κ is Lipschitz continuous, $\lim_{\nu\to\infty} \kappa^{\text{LIN}}(x) = \kappa(x)$, $\forall x \in \mathcal{X}$, it can be noted that:

$$\begin{aligned}
&\forall x \in \mathcal{X},\ \lim_{\nu\to\infty} e^{\text{LIN}}(x) = \\
&= \lim_{\nu\to\infty} |\kappa^{\text{OPT}}(x) - \kappa^{\text{LIN}}(x)| + \lim_{\nu\to\infty} e^{\text{OPT}}(x) = \\
&|\kappa(x) - \kappa(x)| + 0 = 0
\end{aligned}$$

thus

$$\lim_{\nu\to\infty} \zeta^{\text{LIN}}(\nu) = \lim_{\nu\to\infty} \sup_{x\in\mathcal{X}} e^{\text{LIN}}(x,\nu) = 0$$

■

Proof of Theorem 2

i) From (25)-(26) it can be noted that, for any $x \in \mathcal{X}$:

$$\begin{aligned}
\kappa^{\text{NB}}(x) &= \frac{1}{2}(\overline{\kappa}^{\text{NB}} + \underline{\kappa}^{\text{NB}}) \leq \overline{\kappa}^{\text{NB}} \leq \overline{u} \\
\kappa^{\text{NB}}(x) &= \frac{1}{2}(\overline{\kappa}^{\text{NB}} + \underline{\kappa}^{\text{NB}}) \geq \underline{\kappa}^{\text{NB}} \geq \underline{u} \\
&\Rightarrow \kappa^{\text{NB}}(x) \in \mathbb{U}
\end{aligned}$$

ii) For any $x \in \mathcal{X}$, note that (from (12) and (26)):

$$\begin{aligned}
&\underline{\kappa}^{\text{NB}}(x) \leq \underline{\kappa}^{\text{OPT}}(x) \leq \kappa(x) \leq \overline{\kappa}^{\text{OPT}}(x) \leq \overline{\kappa}^{\text{NB}}(x) \\
&\underline{\kappa}^{\text{NB}}(x) - \kappa^{\text{NB}}(x) \leq \kappa(x) - \kappa^{\text{NB}}(x) \leq \overline{\kappa}^{\text{NB}}(x) - \kappa^{\text{NB}}(x) \\
&-\frac{1}{2}(\overline{\kappa}^{\text{NB}}(x) - \underline{\kappa}^{\text{NB}}(x)) \leq \kappa(x) - \kappa^{\text{NB}}(x) \leq \\
&\leq \frac{1}{2}(\overline{\kappa}^{\text{NB}}(x) - \underline{\kappa}^{\text{NB}}(x))
\end{aligned}$$

$$\Rightarrow |\kappa(x) - \kappa^{\text{NB}}(x)| \leq \frac{1}{2}(\overline{\kappa}^{\text{NB}}(x) - \underline{\kappa}^{\text{NB}}(x)) \tag{39}$$

Consider now the distance between the optimal upper bound $\overline{\kappa}^{\text{OPT}}(x)$ (12) and the suboptimal upper bound $\overline{\kappa}^{\text{NB}}(x)$ (26). Since by definition $\overline{\kappa}^{\text{OPT}}(x) \leq \overline{\kappa}^{\text{NB}}(x) \leq \overline{u}$, if $\overline{\kappa}^{\text{OPT}}(x) = \overline{u}$ then $\overline{\kappa}^{\text{NB}}(x) - \overline{\kappa}^{\text{OPT}}(x) = \overline{u} - \overline{u} = 0$. Otherwise note that:

$$\begin{aligned}
&0 < \overline{\kappa}^{\text{NB}}(x) - \overline{\kappa}^{\text{OPT}}(x) \le \\
&\tilde{u}^{\overline{j}} + \gamma\|x - \tilde{x}^{\overline{j}}\|_2 - \tilde{u}^{\overline{k}} - \gamma\|x - \tilde{x}^{\overline{k}}\|_2 \le \\
&\le \gamma\|\tilde{x}^{\overline{j}} - \tilde{x}^{\overline{k}}\|_2 + \gamma\|x - \tilde{x}^{\overline{j}} - x + \tilde{x}^{\overline{k}}\|_2 = \\
&= 2\gamma\|\tilde{x}^{\overline{j}} - \tilde{x}^{\overline{k}}\|_2
\end{aligned}$$

Similarly, it can be obtained that:

$$\begin{aligned}
&0 < \underline{\kappa}^{\text{OPT}}(x) - \underline{\kappa}^{\text{NB}}(x) \le \\
&\tilde{u}^{\underline{k}} - \gamma\|x - \tilde{x}^{\underline{k}}\|_2 - \tilde{u}^{\underline{j}} + \gamma\|x - \tilde{x}^{\underline{j}}\|_2 \le \\
&\le \gamma\|\tilde{x}^{\underline{j}} - \tilde{x}^{\underline{k}}\|_2 + \gamma\|x - \tilde{x}^{\underline{j}} - x + \tilde{x}^{\underline{k}}\|_2 = \\
&= 2\gamma\|\tilde{x}^{\underline{j}} - \tilde{x}^{\underline{k}}\|_2
\end{aligned}$$

thus, the distance between the OPT and NB approximations is bounded:

$$\begin{aligned}
&|\kappa^{\text{OPT}}(x) - \kappa^{\text{NB}}(x)| = \\
&= \frac{1}{2}|\overline{\kappa}^{\text{OPT}}(x) + \underline{\kappa}^{\text{OPT}}(x) - \overline{\kappa}^{\text{NB}}(x) - \underline{\kappa}^{\text{NB}}(x)| \le \\
&\le \frac{1}{2}(|\overline{\kappa}^{\text{OPT}}(x) - \overline{\kappa}^{\text{NB}}(x)| + |\underline{\kappa}^{\text{OPT}}(x) - \underline{\kappa}^{\text{NB}}(x)|) \le \\
&\le \gamma(\|\tilde{x}^{\overline{j}} - \tilde{x}^{\overline{k}}\|_2 + \|\tilde{x}^{\underline{j}} - \tilde{x}^{\underline{k}}\|_2) = \delta(x)
\end{aligned} \tag{40}$$

Consequently, note that:

$$\begin{aligned}
&|\Delta_{\kappa^{\text{NB}}}(x)| = |\kappa(x) - \kappa^{\text{NB}}(x)| \le \\
&\le |\kappa(x) - \kappa^{\text{OPT}}(x) + (\kappa^{\text{OPT}}(x) - \kappa^{\text{NB}}(x))| \le \\
&\le e^{\text{OPT}}(x) + \delta(x),\ \forall x \in \mathcal{X}
\end{aligned} \tag{41}$$

At the same time, since by construction (24) for any $x \in \mathcal{X}$ the set of points $\{\tilde{x}^j : j \in P_{\hat{j}}\}$ contains the nearest neighbor $\tilde{x}^{\text{NP}}$ of x, it can be noted that (from (26)):

$$\begin{aligned}
\overline{\kappa}^{\text{NB}} &\le \tilde{u}^{\text{NP}} + \gamma\|x - \tilde{x}^{\text{NP}}\|_2 \\
\underline{\kappa}^{\text{NB}} &\ge \tilde{u}^{\text{NP}} - \gamma\|x - \tilde{x}^{\text{NP}}\|_2
\end{aligned}$$

Thus, from (39):

$$\begin{aligned}
&|\Delta_{\kappa^{\text{NB}}}(x)| = |\kappa(x) - \kappa^{\text{NB}}(x)| \le \\
&\frac{1}{2}(\overline{\kappa}^{\text{NB}}(x) - \underline{\kappa}^{\text{NB}}(x)) \le \\
&\le \frac{1}{2}(\tilde{u}^{\text{NP}} + \gamma\|x - \tilde{x}^{\text{NP}}\|_2 - \tilde{u}^{\text{NP}} + \gamma\|x - \tilde{x}^{\text{NP}}\|_2) = \\
&= \gamma\|x - \tilde{x}^{\text{NP}}\|_2 = e^{\text{NP}}
\end{aligned} \tag{42}$$

By considering the tightest bound between (41) and (42), it can be obtained that:

$$\begin{aligned}&|\Delta_{\kappa^{\mathrm{NB}}}(x)| \leq \min(e^{\mathrm{NP}}(x), e^{\mathrm{OPT}}(x) + \delta(x)) = e^{\mathrm{NB}}(x) \leq \\ &\leq e^{\mathrm{NP}}(x),\ \forall x \in \mathcal{X} \\ &\zeta^{\mathrm{NB}} = \sup_{x \in \mathcal{X}} e^{\mathrm{NB}} \leq \gamma d_H(\mathcal{X}, \mathcal{X}_\nu) = \zeta^{\mathrm{NP}}\end{aligned}$$

iii) Trivially follows from (42) and the properties of the NP approximation (20)

iv) Trivially follows from (40) by using $\overline{k} = \overline{j}$ and $\underline{k} = \underline{j}$

■

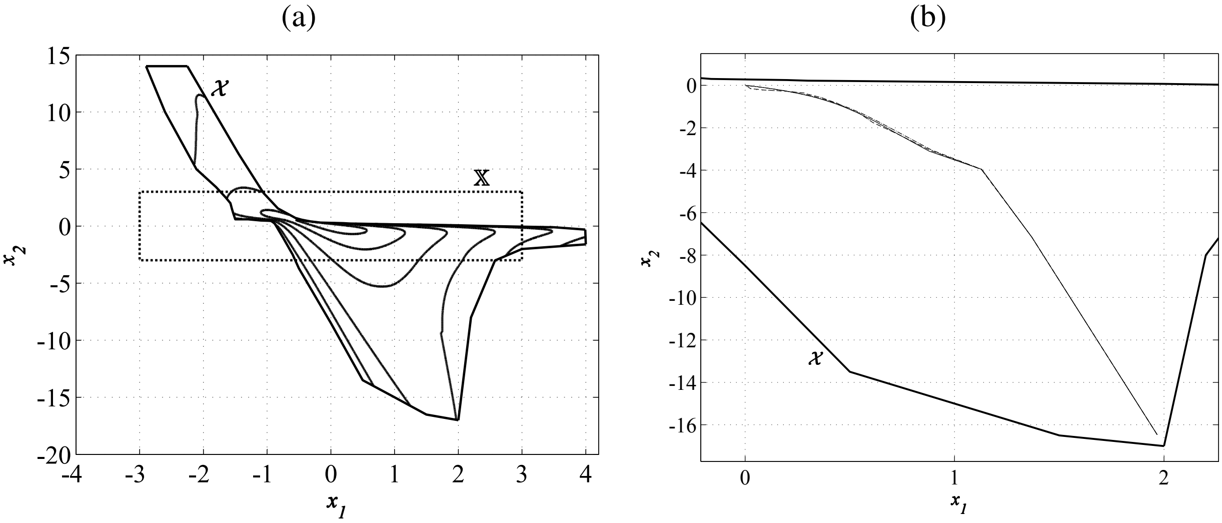


Fig. 1. Numerical example. (a) Set $\mathcal{X}$, constraint set $\mathbb{X}$ (thick dotted line), level curves of the optimal cost function $J^*(x)$ (thick solid lines). (b) Closed loop state trajectories obtained with controllers κ (solid), κ^{OPT} (dotted), κ^{LIN} (dash-dot) and κ^{NB} (dashed). Initial state $x(0) = [2.1,\ -17]^T$, approximations computed using $\nu = 2.5\,10^3$ points.

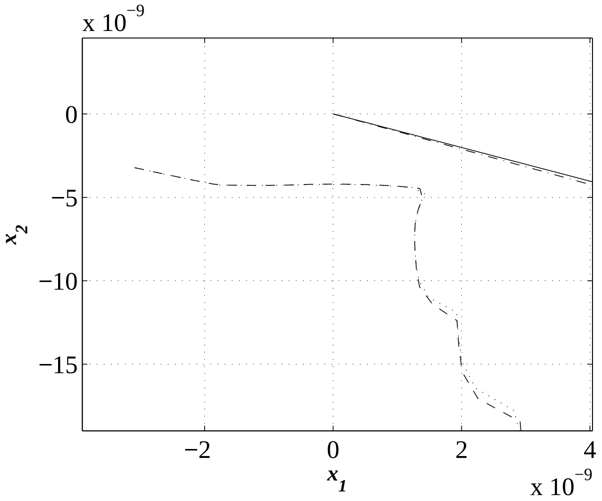


Fig. 2. Numerical example: closed loop state trajectories near the origin, obtained with controllers κ (solid), κ^{OPT} (dotted), κ^{LIN} (dash-dot) and κ^{NB} (dashed). Initial state $x(0) = [2.1,\, -17]^T$, approximations computed using $\nu = 2.5\, 10^3$ points.

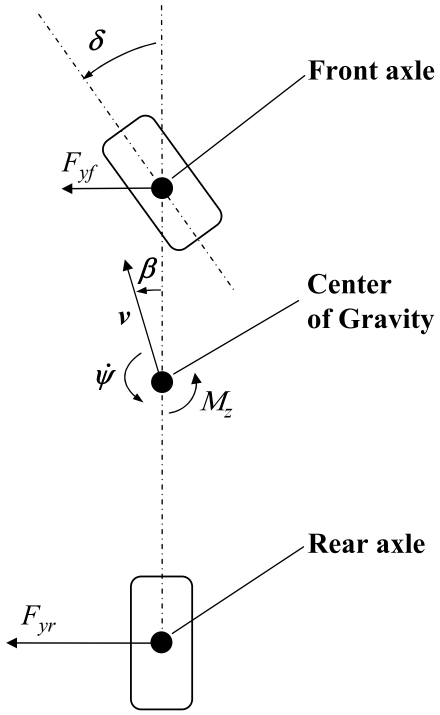


Fig. 3. Sketch of single track vehicle model.

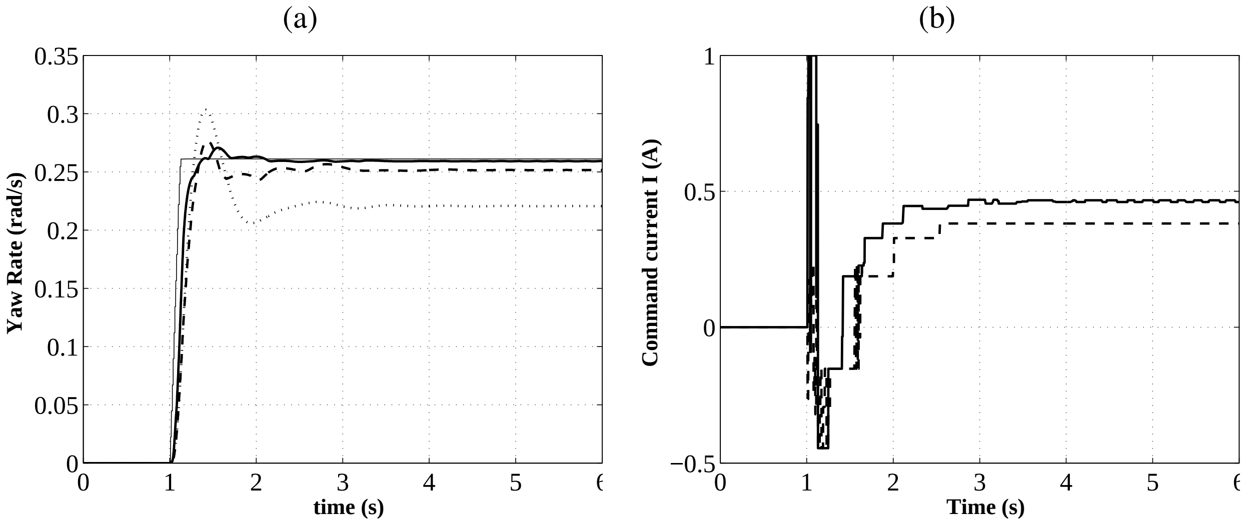
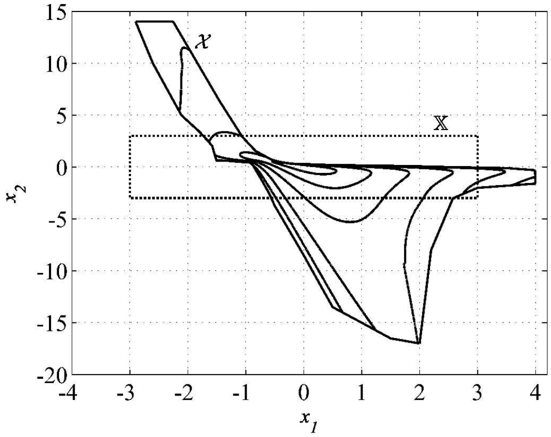
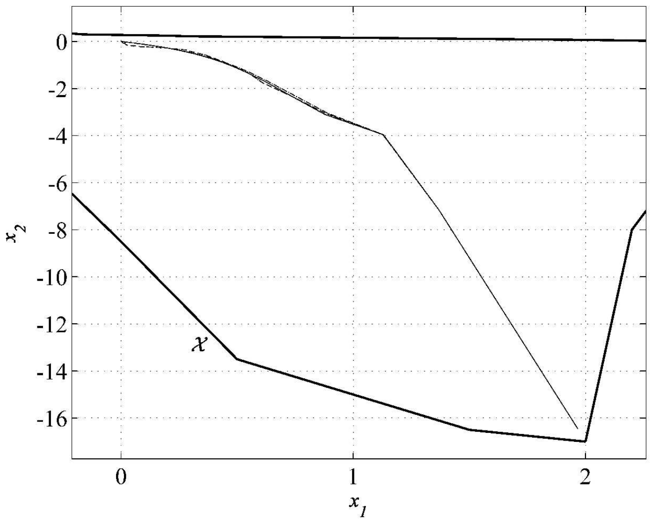


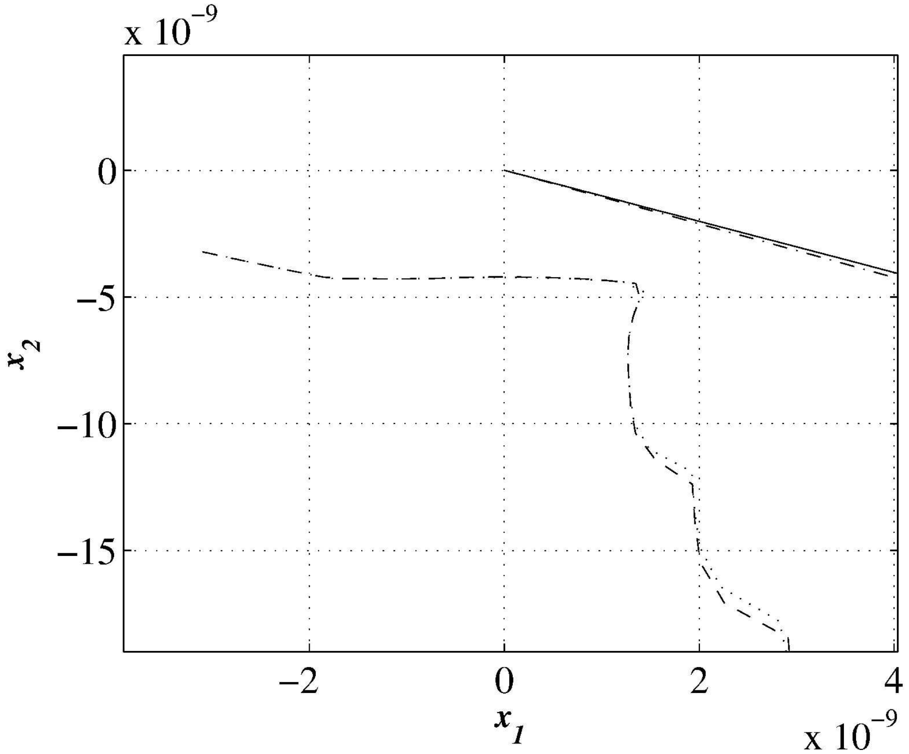
Fig. 4. Case study: vehicle stability control, 50° step steer maneuver at 100 km/h. (a) Courses of the reference yaw rate (thin solid line) and of the yaw rate obtained with the uncontrolled vehicle (dotted) and the controlled one, using the NB approach (solid) or the NP approach (dashed). (b) Courses of the control input i using the NB approach (solid) or the NP approach (dashed).



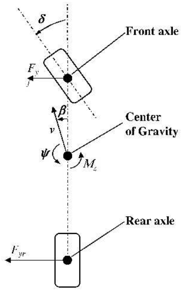
201x288mm (600 x 600 DPI)



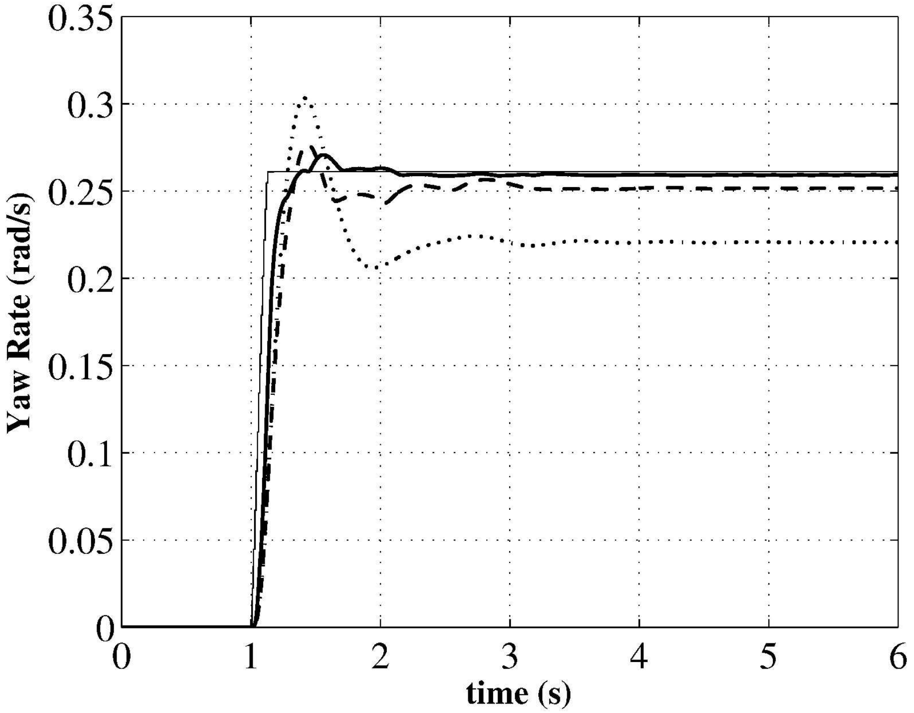
201x288mm (600 x 600 DPI)



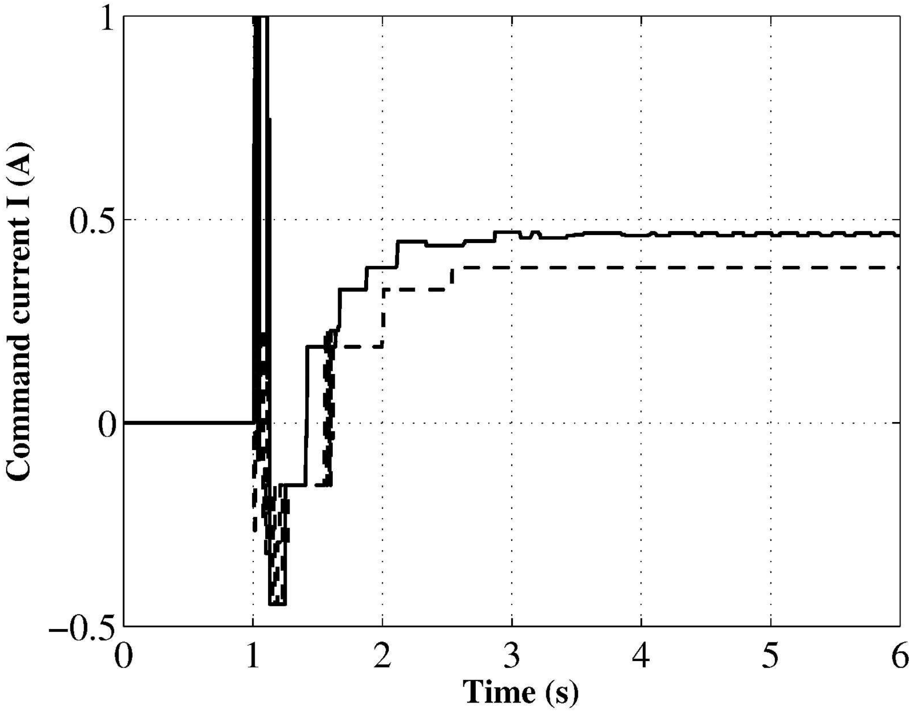
147x110mm (600 x 600 DPI)



288x200mm (600 x 600 DPI)



148x111mm (600 x 600 DPI)



148x111mm (600 x 600 DPI)