2017 IEEE 56th Annual Conference on Decision and Control (CDC)
December 12-15, 2017, Melbourne, Australia

# Stabilizing Non-linear MPC Using Linear Parameter-Varying Representations

Jurre Hanema, Roland Tóth, Mircea Lazar

*Abstract*— **We propose a model predictive control approach for non-linear systems based on linear parameter-varying representations. The non-linear dynamics are assumed to be embedded inside an LPV representation. Hence, the non-linear MPC problem is replaced by an LPV MPC problem, which can be solved through convex optimization. Doing so, the non-linear system can be controlled efficiently and with strong guarantees on feasibility and stability at the possible sacrifice of achievable performance. In this paper, the LPV MPC problem is solved using a tube-based approach, requiring the on-line solution of a single linear- or quadratic program. The computational properties of the approach are demonstrated on two examples.**

## I. INTRODUCTION

Many systems in engineering exhibit significant non-linearities when controlled over a wide operating regime. Because working with complex non-linear models is challenging, it is often useful to employ "simplifying" approaches to model the non-linear system. One such way of modeling the behavior of a non-linear system is by embedding it in a *linear parameter-varying* (LPV) representation [1], [2], [3]. In an LPV system, the dynamical mapping between inputs and outputs is linear while the mapping itself depends on a time-varying and on-line measurable *scheduling variable*. Thus, non-linearities in the original system can be represented by variations in the scheduling signal.

In an LPV system, it is assumed that the scheduling variable is an exogeneous signal that is independent from the states or control inputs. This assumed freedom enables the use of computationally efficient control design procedures. If an LPV model is used to embed an underlying non-linear system and the scheduling signal represents the non-linear behavior, this introduces some conservatism. This is because, generally, the solution set of the LPV embedding will be larger than that of the original non-linear system. The true dependence of the scheduling variable on the states or inputs can be described by a so-called *scheduling map*, and such a map can be used to bound all the possible behaviors of the scheduling variable given knowledge of, e.g., the allowable state trajectories.

In general, non-linear MPC requires the on-line solution of a non-convex optimization problem. If the non-linear model is embedded in an LPV representation, however, an LPV MPC strategy can be employed which only requires the solution of convex programs. Although the optimization problems in LPV MPC often have more decision variables and constraints, they can still be solved efficiently due to their convex structure.

In the MPC scheme of [4], which is based on LMI optimization, it is assumed that the underlying non-linear system can be represented by an LPV model obtained as a family of linearized models around various operating points. It is shown that if the optimization remains feasible, the closed loop is asymptotically stable. However, because the family of linearized models is not guaranteed to embed the true non-linear dynamics, recursive feasibility can not be established a-priori. In the method of [5], the non-linear dynamics are embedded in a LPV representation with a discrete scheduling set, effectively representing a collection of uncertain linear models. *Rate-of-variation* (ROV) bounds on the state – and hence, on the scheduling variable – are imposed in a way reminiscent of classical gain scheduling [1]. The LMI-based approach of [6] also imposes restrictions on the scheduling ROV but does not require the scheduling set to be discrete. The computational complexity of the method therein grows exponentially in the prediction horizon.

In [7], an "iterative" MPC scheme is presented to control non-linear systems embedded in an LPV representation. Unlike the previously discussed approaches, knowledge of the scheduling map is exploited in the prediction stage. Based on an initial guess of the future scheduling trajectory, a simple *linear time-varying* (LTV) MPC problem is solved. The resulting optimal state- and input trajectories are used to generated a new future predicted scheduling trajectory through the scheduling map, and the procedure is iterated until the predicted trajectories converge. This is computationally highly efficient because at each time instant, only the solution of a sequence of LTV MPC problems is required. The authors demonstrated that it works well on some examples, but there are no guarantees concerning convergence of the iterations.

In this paper, we develop a non-linear MPC based on LPV embeddings, which integrates the explicit use of a scheduling map from [7] into a tube-based LPV MPC formulation. In this way, we trade achievable performance for computational efficiency, without sacrificing guarantees on recursive feasibility and stability. The future evolution of the scheduling variable is assumed to belong to a sequence of sets, which is a more general setup compared to assuming just bounds on the rate of variation. The proposed method requires the on-line solution of a single linear- or quadratic program with a number of variables and constraints which grows linearly in the prediction horizon.

Previously, tube-based methods have been used to derive efficient non-linear model predictive controllers based on

This research was supported by the Impulse 1 program of the Eindhoven University of Technology and ASML.

The authors are with the Control Systems Group, Department of Electrical Engineering, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands. E-mails: {j.hanema,r.toth,m.lazar}@tue.nl.

978-1-5090-2872-6/17/$31.00 ©2017 IEEE

linearization. In [8], [9] the non-linear dynamics are linearized around a given feasible "seed" trajectory, and the arising linearization errors are represented as additive perturbations. Although both the proposed method and [8], [9] belong to the class of tube-based policies, the linearized system representation adopted in [8], [9] is fundamentally different from the LPV embedding considered in the current paper.

The paper is organized as follows. Section II describes the notation and problem setup. Section III discusses the proposed MPC methodology, and in Section IV some corresponding initialization strategies are given. Examples demonstrating the properties of the algorithm are provided in Section V.

## II. Preliminaries

### A. Notation

The set of nonnegative real numbers is denoted by $\mathbb{R}_+$ and $\mathbb{N}$ denotes the set of nonnegative integers including zero. Define closed- and open index sets as $\mathbb{N}_{[a,b]} = \mathbb{N} \cap [a, b]$ and $\mathbb{N}_{[a,b)} = \mathbb{N} \cap [a, b)$. The predicted value of a variable $z$ at time instant $k + i$ given the information available at time $k$ is denoted by $z_{i|k}$. A set $X \subset \mathbb{R}^n$ which is convex, compact, and contains the origin in its non-empty interior is called a PC-set. A subset of $\mathbb{R}^n$ is a polyhedron if it is an intersection of finitely many half-spaces. A polytope is a compact polyhedron and can equivalently be represented as the convex hull of finitely many points in $\mathbb{R}^n$. For a set $V \subseteq \mathbb{R}^n$ and a scalar $\alpha \in \mathbb{R}$ let $\alpha V = \{\alpha v \mid v \in V\}$, and for a vector $z \in \mathbb{R}^n$ let $z \oplus V = \{z + v \mid v \in V\}$. The Hausdorff distance between a nonempty set $X \subset \mathbb{R}^n$ and the origin is $d_H^0(X) = \sup_{x\in X} \|x\|$. For a vector $x \in \mathbb{R}^n$, observe that $d_H^0(\{x\}) = \|x\|$. A function $f : \mathbb{R}_+ \to \mathbb{R}_+$ is of class $\mathcal{K}_\infty$ when it is continuous, strictly increasing, $f(0) = 0$, and $\lim_{\xi\to\infty} f(\xi) = \infty$. The gauge function $\psi_S : \mathbb{R}^n \to \mathbb{R}_+$ of a given PC-set $S \subset \mathbb{R}^n$ is $\psi_S(x) = \inf\{\gamma \mid x \in \gamma S\}$. We use the following generalized "set"-gauge function:

*Definition 1:* The set-gauge function $\Psi_S : 2^{\mathbb{R}^n} \to \mathbb{R}_+$ corresponding to a PC-set $S \subset \mathbb{R}^n$ is

$$\Psi_S(X) := \sup_{x\in X} \psi_S(x) = \inf \{\gamma \mid X \subseteq \gamma S\}.$$

### B. Problem setup

This paper considers non-linear systems

$$x(k+1) = f\left(x(k), u(k)\right), \ k \in \mathbb{N}, \ x(0) = x_0, \quad (1)$$

that can be embedded[1] in an LPV representation like

$$x(k+1) = A\left(\theta(k)\right) x(k) + Bu(k), \quad (2a)$$
$$\theta(k) = T\left(x(k)\right), \quad (2b)$$

where $u : \mathbb{N} \to \mathbb{U} \subseteq \mathbb{R}^{n_u}$ is the control input and $x : \mathbb{N} \to \mathbb{X} \subseteq \mathbb{R}^{n_x}$ is the state vector. The sets $\mathbb{U}$ and $\mathbb{X}$ are the input- and state constraint sets, respectively. The scheduling signal $\theta : \mathbb{N} \to \Theta \subseteq \mathbb{R}^{n_\theta}$ is state-dependent through the *scheduling map* $T : \mathbb{X} \to \Theta$. The set $\Theta$ is called the scheduling set. The matrix $A(\theta)$ in (2) is assumed to be affine in $\theta$, i.e.,

$$A(\theta) = A_0 + \sum_{i=1}^{n_\theta} \theta_i A_i \quad (3)$$

where $A_i$, $i \in \mathbb{N}_{[0,n_\theta]}$ are matrices of conformable dimensions. We assume that the system satisfies the following assumptions, which are standard in state-feedback MPC design:

*Assumption 1:* (i) The state vector $x(k)$ can be measured for all $k \in \mathbb{N}$. (ii) $\mathbb{X}$ and $\mathbb{U}$ are PC-sets. (iii) The origin is an equilibrium of (1), i.e., $f(0, 0) = 0$.

The representation (2) is an LPV embedding of (1), if it satisfies the following definition.

*Definition 2:* Let $\Theta \subseteq \mathbb{R}^{n_\theta}$ be compact. The representation (2) is an LPV embedding of (1) on the open set $\mathcal{X} \subset \mathbb{R}^{n_x}$, if for all signals $x : \mathbb{N} \to \mathcal{X}$ and $u : \mathbb{N} \to \mathbb{U}$ satisfying (1), there exists a $\theta : \mathbb{N} \to \Theta$ such that $(x, u, \theta)$ satisfies (2a).

From the above definition it follows that if $\mathbb{X} \subset \mathcal{X}$, the embedding is valid on the state constraint set $\mathbb{X}$ which is compact by Assumption 1. The signal $\theta$ in Definition 2 is not necessarily unique. The embedding is in general a "over-approximation" of (1), in the sense that the set of trajectories satisfying (2a) can be larger than the set of trajectories satisfying (1). In this definition, the scheduling map $T(\cdot)$ does not play a role, as the scheduling signal in an LPV representation is considered free. Knowledge of $T(\cdot)$ can be exploited to obtain refined sets of all possible scheduling trajectories; this is what will be done in the MPC developed in this paper. There is unfortunately no general approach to construct an embedding for arbitrary non-linear systems, and the choice of embedding is not unique. Methods of constructing embeddings for relevant classes of systems can be found in, e.g., [2], [3], [10]. An important property of embeddings is the following.

*Lemma 1:* Suppose that there exists a control $u : \mathbb{N} \to \mathbb{U}$ such that all possible trajectories of (2a) are driven to the origin. Then, the same control sequence drives the state trajectory of (1) to the origin as well.

In this paper, a stabilizing MPC will be designed for the representation (2), which by the above Lemma is then guaranteed to stabilize (1). In the sequel, the map $T(\cdot)$ will be applied to set-valued arguments $X \subseteq \mathbb{X}$ according to

$$T\left(X\right) := \operatorname{convh}\left\{T(x) \mid x \in X\right\}. \quad (4)$$

Furthermore, it must satisfy the following properties.

*Assumption 2:* (i) $T : \mathbb{X} \to \Theta$ is continuous, bounded, and such that (4) maps compact sets into compact sets. (ii) For any two compact convex sets $X_1, X_2 \subseteq \mathbb{X}$ such that $X_1 \subset X_2$, it holds $T\left(X_1\right) \subset T\left(X_2\right)$.

## III. The MPC approach

We adopt a modified version of the setup proposed in [11]:

*Definition 3:* A tube is defined as

$$\mathbf{T}_k := \left(\left\{X_{0|k}, \ldots, X_{N|k}\right\}, \left\{\Pi_{0|k}, \ldots, \Pi_{N-1|k}\right\}\right)$$

where $X_{i|k} \subseteq \mathbb{R}^{n_x}, i \in \mathbb{N}_{[0,N]}$ are sets ("cross sections") and $\Pi_{i|k} : X_{i|k} \times \Theta_{i|k} \to \mathbb{U}, i \in \mathbb{N}_{[0,N-1]}$ are control laws

[1] A class of systems that can always be embedded in the required form is $x(k+1) = \tilde{f}(x)x + Bu$ where $\tilde{f} : \mathbb{X} \to \mathbb{R}^{n_x}$ is bounded on a compact set $\mathbb{X} \subset \mathbb{R}^{n_x}$. Conditions on when an embedding of the form (2) can be constructed for an arbitrary continuous-time non-linear system can be found in [10]. The discrete-time case is still a topic of ongoing research.

satisfying the condition $\forall(x,\theta) \in X_{i|k} \times \Theta_{i|k} : A(\theta)x + B\Pi_{i|k}(x,\theta) \in X_{i+1|k}$ and $X_{i|k} \subseteq \mathbb{X}_{i|k}$.

In the considered setting, the role of the sequence

$$\vec{\Theta}_k := \{\Theta_{0|k}, \ldots, \Theta_{N-1|k}\} \tag{5}$$

is to bound all possible behaviors of $\theta_{i|k}$ in the embedding (2). Definition 3 allows for time-varying state constraints

$$\vec{\mathbb{X}}_k := \{\mathbb{X}_{0|k}, \ldots, \mathbb{X}_{N-1|k}\} \tag{6}$$

and the basic approach in this work is to construct at each time instant a state constraint sequence $\vec{\mathbb{X}}_k$, and generate an associated scheduling sequence $\vec{\Theta}_k$ by applying the scheduling map $T(\cdot)$. So, the signal $\theta$ is still free, but all its possible future trajectories are known to belong to $\vec{\Theta}_k$.

In [11], the state constraints were represented by a single time-invariant set $\mathbb{X}$. Furthermore, in Definition 3 it is required that the cross sections $X_{i|k}$ are fully included in the state constraints, whereas [11] used the relaxed requirement that only the realized trajectories need to satisfy the constraints. Due to the relationship between $\vec{\mathbb{X}}_k$ and $\vec{\Theta}_k$ that exists in the current work through $T(\cdot)$, the more restrictive condition of Definition 3 is necessary for recursive feasibility. The MPC optimization problem to be solved on-line is

$$\begin{aligned} V(x_{0|k}, \vec{\Theta}_k, \vec{\mathbb{X}}_k) &= \min_{\mathbf{d}} J_N(\mathbf{d}) \\ \text{subject to} \quad & \mathbf{d} \in \mathcal{D}_N(x_{0|k}, \vec{\Theta}_k, \vec{\mathbb{X}}_k) \end{aligned} \tag{7}$$

where $\mathbf{d} \in \mathbb{R}^{n_\mathrm{d}}$ is a decision variable and where

$$J_N(\mathbf{d}) = \sum_{i=0}^{N-1} \ell_p\left(X_{i|k}, \Pi_{i|k}\right) + F\left(X_{N|k}\right). \tag{8}$$

The stage cost $\ell_p(\cdot,\cdot)$ is designed to meet the user specified performance objective and the terminal cost $F(\cdot)$ is designed to achieve stability. The set $\mathcal{D}_N(\cdot,\cdot,\cdot)$ is characterized by

$$\begin{aligned} \mathcal{D}_N(x, \vec{\Theta}, \vec{\mathbb{X}}) = \{ &\mathbf{d} \in \mathbb{R}^{n_\mathrm{d}} \mid x \in X_{0|k},\, X_{N|k} \subseteq X_f, \\ &\forall i \in \mathbb{N}_{[0,N]} : \left(X_{i|k}, \Pi_{i|k}\right) \text{ satisfies Def. 3} \} \end{aligned} \tag{9}$$

where $X_f$ is a terminal set designed to obtain recursive feasibility as described later. In (7)-(9), the sets $X_{i|k}$ and controllers $\Pi_{i|k}$ are functions of the decision variable $\mathbf{d}$, but this dependency is omitted from the notation for brevity. The set of feasible initial states corresponding to (7) is

$$\mathcal{X}_N\left(\vec{\Theta}, \vec{\mathbb{X}}\right) = \left\{x \;\middle|\; \mathcal{D}_N(x, \vec{\Theta}, \vec{\mathbb{X}}) \neq \emptyset\right\}. \tag{10}$$

In (7), it is desired to parameterize the cross-sections and control laws, such that $\mathbf{d}$ is of finite dimension and $\mathcal{D}_N(\cdot,\cdot,\cdot)$ can be described by finitely many constraints. This paper considers "homothetic tubes", as originally used in TMPC for linear systems subject to additive disturbances [12], [13], which are summarized in the following definition.

*Definition 4:* Let $S = \operatorname{convh}\{\bar{s}^1, \ldots, \bar{s}^{q_s}\}$ be a PC-set, and let each $\Theta_{i|k}$ in $\vec{\Theta}_k$ be a polytope $\Theta_{i|k} = \operatorname{convh}\left\{\bar{\theta}^1_{i|k}, \ldots, \bar{\theta}^q_{i|k}\right\}$. A tube satisfying Definition 3 is called a homothetic tube if for all $i \in \mathbb{N}_{[0,N]}$, $X_{i|k} = z_{i|k} \oplus \alpha_{i|k} S$, and $\Pi_{i|k}$ are vertex controllers $\Pi_{i|k}(x,\theta) = \sum_{j=1}^{q_s} \zeta_j \sum_{l=1}^{q} \eta_l u^{(j,l)}_{i|k}$ where $(\zeta_j, \eta_l)$ are convex multipliers in the state- and scheduling spaces, respectively. Thus, each pair $\left(X_{i|k}, \Pi_{i|k}\right)$ is fully and uniquely determined by the parameters $p_{i|k} = \left(p^{\mathrm{x}}_{i|k}, p^{\pi}_{i|k}\right)$ where $p^{\mathrm{x}}_{i|k} = \left(z_{i|k}, \alpha_{i|k}\right) \in \mathbb{R}^{n_\mathrm{x}} \times \mathbb{R}_+$ and $p^{\pi}_{i|k} = \left(u^{(1,1)}_{i|k}, \ldots, u^{(q_s,q)}_{i|k}\right)$. In other words, there exists a function $\bar{P}(\cdot)$ such that $\bar{P}\left(p_{i|k}\right) = \left(X_{i|k}, \Pi_{i|k}\right)$.

From Definition 4, it follows that the vector of decision variables $\mathbf{d} \in \mathbb{R}^{n_\mathrm{d}}$ contains all parameters $p_{i|k}$, $i \in \mathbb{N}_{[0,N]}$ defining a tube of length $N$. Define $r(2) := 2$ and $r(\infty) := 1$. A stage cost function can be designed as

$$\begin{aligned} \ell_p\left(X_{i|k}, \Pi_{i|k}\right) = \|Q z_{i|k}\|_p^{r(p)} + P|\alpha_{i|k}|^{r(p)} \\ + \frac{1}{q q_s} \sum_{i=1}^{q_s} \sum_{j=1}^{q} \|R \bar{u}^{(i,j)}_{i|k}\|_p^{r(p)} \end{aligned} \tag{11}$$

where we allow $p \in \{2, \infty\}$, and where $Q \in \mathbb{R}^{n_\mathrm{q} \times n_\mathrm{x}}$, $R \in \mathbb{R}^{n_\mathrm{r} \times n_\mathrm{u}}$ and the scalar $P > 0$ are tuning parameters. The matrices $(Q, R)$ must be of full column rank. This cost is different from that in [11], where a worst-case objective was minimized and where only the infinity-norm case was treated.

To guarantee recursive feasibility of (7), a terminal set constraint $X_{N|k} \subseteq X_f$ is included in (9). The set $X_f$ must be designed to be controlled $\lambda$-contractive for (2), which in the current setting must be understood as follows.

*Definition 5:* Let $X_f$ and $\breve{\mathbb{X}}$ both be PC-sets such that $X_f \subseteq \breve{\mathbb{X}} \subseteq \mathbb{X}$ and let $\lambda \in [0, 1)$. Then, $X_f$ is controlled $\lambda$-contractive for the system (2) if $\forall(x,\theta) \in X_f \times T(\breve{\mathbb{X}})$ : $\exists u \in \mathbb{U} :\ A(\theta)x + Bu \in \lambda X_f$.

Suppose that $X_f$ is $\lambda$-contractive according to the above definition. A suitable terminal cost $F(\cdot)$ to be used in (8) is

$$\begin{aligned} F\left(X_{N|k}\right) &= \frac{\Psi_{X_f}\left(X_{N|k}\right)}{1-\lambda} \max_{X \subseteq X_f, \theta \in T(\breve{\mathbb{X}})} \ell_p\left(X, \kappa\right) \\ &= \frac{c}{1-\lambda} \Psi_{X_f}\left(X_{N|k}\right), \end{aligned} \tag{12}$$

where the maximization is done with respect to sets $X = z \oplus \alpha S$. The function $\Psi_{X_f}(\cdot)$ is the set-gauge of $X_f$, as specified in Definition 1, and $\kappa : X_f \times T\big(\breve{\mathbb{X}}\big) \to \mathbb{U}$ is a local set-induced controller that renders $X_f$ $\lambda$-contractive. The maximization over $\theta \in T(\breve{\mathbb{X}})$ is necessary, because $\kappa(\cdot,\cdot)$ depends on $\theta$. As proven later in Theorem 1, this terminal cost leads to asymptotic closed-loop stability. Because $\ell_p(\cdot,\cdot)$ is convex, (12) reduces to a finite-dimensional optimization problem provided that $T\big(\breve{\mathbb{X}}\big)$ is a polytope, as then $\ell_p(\cdot,\cdot)$ takes its maximum on one of the vertices of $X_f \times T\big(\breve{\mathbb{X}}\big)$.

The proposed MPC approach is summarized in Algorithm 1, and Theorem 1 gives the main result concerning its properties:

*Theorem 1:* Let $X_f$ satisfy Definition 5, let $F(\cdot)$ be as in (12), and in Definition 4 set $S = X_f$. Then, Algorithm 1 has the following properties: (i) It is recursively feasible, i.e., if $\mathcal{D}_N\left(x_{0|0}, \vec{\Theta}_0, \vec{\mathbb{X}}_0\right) \neq \emptyset$ then $\forall k \in \mathbb{N}_{[1,\infty)}$ : $\mathcal{D}_N\left(x_{0|k}, \vec{\Theta}_k, \vec{\mathbb{X}}_k\right) \neq \emptyset$. (ii) The state of the controlled system reaches $\breve{\mathbb{X}}$ in $N$ steps or less, i.e., $\exists k_\star \in \mathbb{N}_{[0,N]}$ such that $\forall k \geq k_\star :\ x(k) \in \breve{\mathbb{X}}$. (iii) The origin is asymptotically closed-loop attractive, i.e., $x(k) \to 0$ as $k \to \infty$.

*Proof:* See the Appendix. ■

**Algorithm 1** The MPC algorithm.

**Require:** a given sequence $\vec{\mathbb{X}}_0$

1: $k \leftarrow 0$

2: **loop**

3: $\quad \forall i \in \mathbb{N}_{[0,N-1]} : \Theta_{i|k} \leftarrow \begin{cases} \{T\left(x_{0|k}\right)\}, & i = 0, \\ T\left(\mathbb{X}_{i|k}\right), & i > 0. \end{cases}$

4: $\quad$ Solve optimization (7)

5: $\quad$ **if** feasible **then**

6: $\quad\quad$ Apply $u(k) = \Pi^{\star}_{0|k}\left(x_{0|k}, \theta_{0|k}\right) = u^{\star}_{0|k}$

7: $\quad\quad \forall i \in \mathbb{N}_{[0,N-1]} : \mathbb{X}_{i|k+1} \leftarrow \begin{cases} \mathbb{X}_{i+1|k}, & i < N-1, \\ \breve{\mathbb{X}}, & i = N-1. \end{cases}$

8: $\quad\quad k \leftarrow k+1$

9: $\quad$ **else**

10: $\quad\quad$ **abort**

11: $\quad$ **end if**

12: **end loop**

*Remark 1:* If the sets in $\vec{\Theta}_k$ are polytopes, then (7) can be formulated as a linear- or quadratic program along the lines discussed in [11]. This problem has a number of variables and constraints that grows linearly in $N$. If $X$ is a polytope, then $T(X)$ is compact by Assumption 2, but it is not necessarily a polytope. It is therefore assumed from now on that if this is the case, each non-polytopic set $\Theta_{i|k} = T(\mathbb{X}_{i|k})$ can be replaced by a polytopic outer-approximation. Such approximations can be found, e.g., using interval arithmetic [14].

## IV. Initialization procedures

In the preceding section it was shown that the constructed MPC is recursively feasible and stabilizes (2) – and hence the non-linear system (1). For this result, it was necessary to assume that a feasible solution exists at $k = 0$. Hence, an initialization procedure is necessary to construct a sequence $\vec{\mathbb{X}}_0$ which achieves this initial feasibility. This section discusses two different possible initialization schemes.

### A. Bounded rate of variation

Let $\delta \in \mathbb{R}^{n_x}_{+}$ denote a bound on the *rate of variation* (ROV) of the state variable and define

$$\Delta(\delta) = \left\{x \in \mathbb{R}^{n_x} \;\middle|\; \forall i \in \mathbb{N}_{[1,n_x]} : \; |x_i| \leq \delta_i\right\}. \tag{13}$$

Then, the initial constraint sequence $\vec{\mathbb{X}}$ can be computed as

$$\forall i \in \mathbb{N}_{[0,N]} : \mathbb{X}_{i|0} = \begin{cases} \{x_{0|0}\}, & i = 0, \\ \left(\mathbb{X}_{i-1|0} \oplus \Delta(\delta)\right) \cap \mathbb{X}, & i > 0. \end{cases} \tag{14}$$

Using a ROV bound is simple, but might be conservative. A different initialization is proposed in the next subsection.

### B. Initial feasible trajectory

Suppose that a feasible initial state- and input trajectory

$$\tilde{\mathbf{T}}_0 = \left(\left\{\tilde{x}_{0|0}, \tilde{x}_{1|0}, \ldots, \tilde{x}_{N|0}\right\}, \left\{\tilde{u}_{0|0}, \ldots, \tilde{u}_{N-1|0}\right\}\right)$$

is known where $\tilde{x}_{N|k} \in X_f$ and where for all $i \in \mathbb{N}_{[0,N-1]}$: $\tilde{x}_{i|0} \in \mathbb{X}$, $\tilde{u}_{i|0} \in \mathbb{U}$, and $\tilde{x}_{i+1|0} = A\big(T(\tilde{x}_{i|0})\big)\tilde{x}_{i|0} + B\tilde{u}_{i|0}$. Then, by defining some tolerance $\delta \in \mathbb{R}^{n_x}_{+}$, the initial state constraint sets can be computed according to the relationship

$$\forall i \in \mathbb{N}_{[0,N]} : \; \mathbb{X}_{i|0} = \begin{cases} \{x_{0|0}\}, & i = 0, \\ \left(\tilde{x}_{i|0} \oplus \Delta(\delta)\right) \cap \mathbb{X}, & i > 0, \end{cases} \tag{15}$$

with $\Delta(\cdot)$ as in (13). In comparison to the bounded ROV initialization discussed earlier, this approach is generally more likely to yield a feasible solution to (7), but the assumed availability of the initially feasible trajectory $\tilde{\mathbf{T}}_0$ can be considered a disadvantage (note that linearization-based approaches [8], [9] also presume the knowledge of such a trajectory). If $\vec{\mathbb{X}}_0$ is constructed according to (15) with $\delta = 0$, then (7) is always feasible at $k = 0$. Due to the constraint update Step 7 of Algorithm 1, a value of $\delta = 0$ however means that the controller can never deviate from the initial trajectory, leading to poor robustness against unmodeled perturbations.

The trajectory $\tilde{\mathbf{T}}_0$ can be obtained by solving a non-linear MPC problem. This could be computationally expensive, but this problem only needs to be solved once at the first sample. An efficient approach that can be used to generate an initial trajectory is [7]. The method [7] is not guaranteed to converge to a solution, but if it converges, the result can be used to initialize the MPC proposed in this paper which subsequently guarantees recursive feasibility and stability.

## V. Numerical examples

In this section, the properties of the approach are demonstrated on two numerical examples. All non-linear MPC problems were solved to optimality using the SQP method implemented in `fmincon` in MATLAB 2016b with its default settings. The linear- and quadratic programs of the proposed LPV-based MPC solution were solved by Gurobi 6.5. Everything was executed on a PC with a 3.6 GHz Intel Core i7-4790 processor and 8 GB RAM, running Arch Linux.

### A. Example 1

This example considers a controlled Van der Pol oscillator described by $\ddot{q}(t) = \mu\left(1 - q^2\right)\dot{q}(t) - q(t) + u(t)$. An equivalent LPV form of this non-linear model is

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ -1 & \mu\left(1-\theta\right) \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$
$$\theta(t) = T\left(x(t)\right) = \left(\begin{bmatrix} 1 & 0 \end{bmatrix} x(t)\right)^2$$

with the state vector $x = \begin{bmatrix} q & \dot{q} \end{bmatrix}^\top$. The model that will be used for control and simulation is an Euler-discretized version of the above continuous-time LPV representation and is

$$x(k+1) = \begin{bmatrix} 1 & \tau \\ -\tau & 1 + \tau\mu\left(1-\theta\right) \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ \tau \end{bmatrix} u(k)$$
$$\theta(k) = T\left(x(k)\right) = \left(\begin{bmatrix} 1 & 0 \end{bmatrix} x(k)\right)^2.$$

The used damping parameter is $\mu = 2$ and the sampling time is $\tau = 0.1$. The scheduling map is not affine, however it is convex and exact upper- and lower bounds on $T(X)$ can be obtained easily for polytopes $X \subseteq \mathbb{X}$. In this example, the stage cost was of the quadratic type ($p = 2$) with $Q = I$, $R = 0.1$ and $P = 15$, and the prediction horizon was $N = 10$.

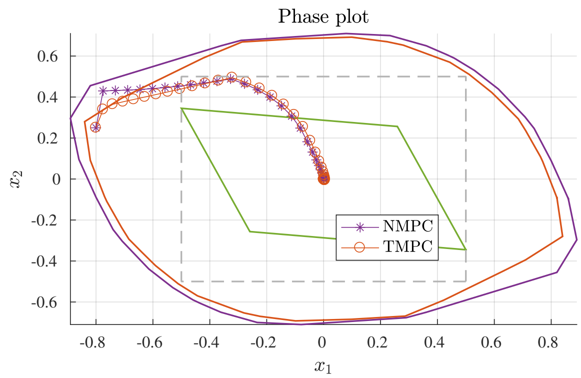


Fig. 1. Phase plot of closed-loop trajectory in Example 1 with terminal set (green), set $\breve{\mathbb{X}}$ (dashed grey), and estimated domains of attraction.

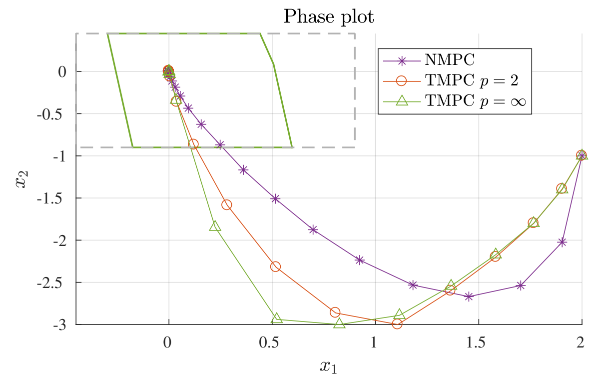


Fig. 2. Phase plot of closed-loop trajectory in Example 2 with terminal set (green) and set $\breve{\mathbb{X}}$ (dashed grey).

| Case | Mean | Min | Max | Std.dev. |
|---|---|---|---|---|
| NMPC (fmincon) | 48 | 34 | 74 | 13 |
| TMPC (proposed) | 10 | 9.1 | 12 | 1.0 |

TABLE I

COMPUTATION TIMES PER SAMPLE IN EXAMPLE 1 (VALUES IN MS).

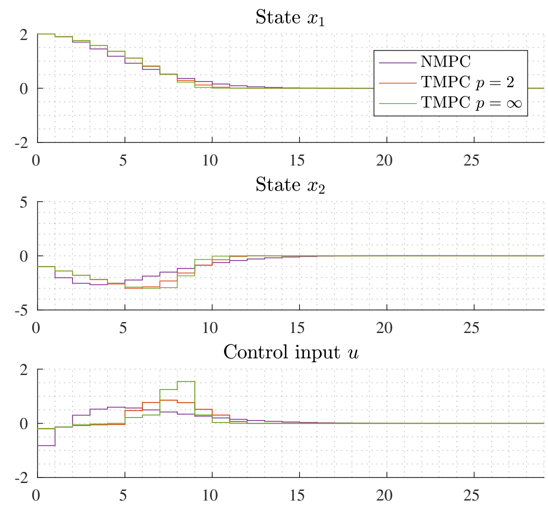


Fig. 3. Closed-loop state- and input trajectories in Example 2.

The state constraints are $\mathbb{X} = \{x \,|\, \|x\|_\infty \leq 1\}$, the input constraint is $\mathbb{U} = \{u \,|\, |u| \leq 1\}$, and the set $\breve{\mathbb{X}}$ was chosen to be $\breve{\mathbb{X}} = 0.5\mathbb{X}$. Associated with this $\breve{\mathbb{X}}$, a 0.96-contractive set $X_f$ satisfying Definition 5 was computed using the standard reachable set-iteration approach.

The simulation results are shown in Figure 1. For this example, no special initialization of $\vec{\mathbb{X}}_0$ was necessary: the method already works well with $\mathbb{X}_{i|0} = \mathbb{X}$ for all $i$. The same terminal set and terminal cost (12) was used in both the proposed tube-based and in the non-linear MPC. In Figure 1, it can be seen that the obtained closed-loop trajectories are comparable. The computation time per sample of the proposed method was lower than that of `fmincon`, as summarized in Table I. The greater efficiency comes at the cost of a slightly reduced domain of attraction $\mathcal{X}_N(\cdot, \cdot)$, as visible in Figure 1.

### B. Example 2

Consider a non-linear system similar to [1, Example 4]

$$\dot{x}_1 = \sin(\omega x_1) + x_2, \quad \dot{x}_2 = x_1 x_2 + u$$

which, after Euler discretization, admits a representation

$$x(k+1) = \left( \begin{bmatrix} 1 & \tau \\ 0 & 1 \end{bmatrix} + \theta_1(k) \begin{bmatrix} \tau & 0 \\ 0 & 0 \end{bmatrix} + \theta_2(k) \begin{bmatrix} 0 & 0 \\ 0 & \tau \end{bmatrix} \right) x(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u, \quad \theta(k) = \begin{bmatrix} \operatorname{sinc}(\omega x_1(k)) \\ x_1(k) \end{bmatrix},$$

where

$$\operatorname{sinc}(x) := \begin{cases} 1, & x = 0, \\ \frac{\sin(\pi x)}{\pi x}, & x \neq 0. \end{cases}$$

In this example, the polytopic outer-approximations of $T(\cdot)$ (see Remark 1) were computed by densely gridding the state constraint sets, evaluating the sinc-function on the grid to approximate its minimum- and maximum values, and adding a small tolerance to account for possible approximation errors. Here, such an approach is tractable because the sinc-term is only dependent on the first state variable $x_1$. The tuning parameters are $Q = \operatorname{diag}\{1, 0.1\}$, $R = 1$, $P = 5$, and $N = 10$. The state constraints are $\mathbb{X} = \{x \,|\, -1.5 \leq x_1 \leq 3, -3 \leq x_2 \leq 1.5\}$, the input constraint is $\mathbb{U} = \{u \,|\, -1 \leq u \leq 2\}$, and the set $\breve{\mathbb{X}}$ was chosen to be $\breve{\mathbb{X}} = 0.30\mathbb{X}$. A 0.95-contractive set $X_f$ satisfying Definition 5 was computed.

A simulation result for the initial state $x_0 = \begin{bmatrix} 2 & -1 \end{bmatrix}^\top$ is shown in Figures 2-3 and Table II. In this example, the performance of using a quadratic ($p = 2$) cost is compared to that achieved with a infinity-norm based cost ($p = \infty$). The ROV-bound initialization was used with $\delta = \begin{bmatrix} 0.40 & 0.40 \end{bmatrix}^\top$.

Note that for small systems, explicit LPV MPC could also be a computationally viable alternative to non-linear MPC [15]. However, in explicit MPC the controller is computed off-line, so it can not handle the time-varying constraints and scheduling sets of (5)-(6). This makes a direct comparison to the proposed approach difficult.

| Case | Mean | Min | Max | Std.dev. |
|---|---|---|---|---|
| NMPC (fmincon, $p = 2$) | 89 | 68 | 135 | 21 |
| TMPC (proposed, $p = 2$) | 12 | 11 | 14 | 1.1 |
| TMPC (proposed, $p = \infty$) | 12 | 7.2 | 16 | 1.6 |

TABLE II

COMPUTATION TIMES PER SAMPLE IN EXAMPLE 2 (VALUES IN MS).

## APPENDIX

The proof of Theorem 1 given in this Appendix builds on the results presented in [16]. Adaptations are made to accommodate the state-dependent scheduling map (2b), the specialized notion of $\lambda$-contractive set from Definition 5, and the different stage cost (11). Because $X_f$ is $\lambda$-contractive in the sense of Definition 5, there exists a controller $\kappa : X_f \times T(\breve{\mathbb{X}}) \to \mathbb{U}$ with corresponding local set-valued dynamics

$$\begin{aligned} X(k+1) &= G\left(X(k)|\kappa\right) \\ &= \left\{A(\theta)x + B\kappa(x,\theta) \mid x \in X(k),\ \theta \in T(\breve{\mathbb{X}})\right\} \end{aligned} \tag{16}$$

for which it holds

$$\forall X \subseteq X_f :\ \Psi_{X_f}\left(G\left(X|\kappa\right)\right) \leq \lambda \Psi_{X_f}\left(X\right). \tag{17}$$

The next non-restrictive assumption on $\kappa(\cdot,\cdot)$ can be made.

*Assumption 3:* The local controller $\kappa : X_f \times T\left(\breve{\mathbb{X}}\right)$ is (i) continuous and (ii) positively homogeneous in the sense that $\forall \alpha \in [0,1] :\ \kappa(\alpha x, \theta) = \alpha\kappa(x,\theta)$.

The following three auxiliary lemmas are necessary. Their proofs are omitted for lack of space.

*Lemma 2:* The terminal cost $F(\cdot)$ defined in (12) satisfies: (i) $\exists \underline{s}_F, \overline{s}_F \in \mathcal{K}_\infty$ such that $\forall X \subseteq X_f : \underline{s}_F\left(d_H^0\left(X\right)\right) \leq F\left(X\right) \leq \overline{s}_F\left(d_H^0\left(X\right)\right)$, and (ii) $\forall X \subseteq X_f :\ F\left(G\left(X|\kappa\right)\right) - F\left(X\right) \leq -\ell_p\left(X,\kappa\right)$ for $p \in \{2,\infty\}$.

*Lemma 3:* Assume that $Q, R$ are full-rank matrices. Let $\Pi : \mathbb{X} \times \Theta \to \mathbb{U}$ be any controller. Define $\Omega := \left\{z \oplus \alpha S \mid (\alpha, z) \in \mathbb{R}_+ \times \mathbb{R}^{n_x}\right\}$. The stage cost (11) satisfies $\exists \underline{s}_\ell \in \mathcal{K}_\infty : \forall X \in \Omega :\ \underline{s}_\ell\left(d_H^0\left(X\right)\right) \leq \ell_p\left(X, \Pi\right)$ for $p \in \{2,\infty\}$.

*Lemma 4:* Let $\left(\vec{\mathbb{X}}, \vec{\Theta}\right)$ be given sequences according to (6)-(5). There exist $\mathcal{K}_\infty$-functions $\underline{s}_V, \overline{s}_V$ such that for all $x \in \mathcal{X}_N\left(\vec{\mathbb{X}}, \vec{\Theta}\right)$ it holds $\underline{s}_V\left(\|x\|\right) \leq V\left(x, \vec{\mathbb{X}}, \vec{\Theta}\right) \leq \overline{s}_V\left(\|x\|\right)$.

The main result can now be proven.

*Proof of Theorem 1:* (i) Suppose that (7) is feasible at time $k$ and let $\mathbf{T}_k^\star = \left(\left\{X_{0|k}, \ldots, X_{N|k}\right\}, \left\{\Pi_{0|k}, \ldots, \Pi_{N-1|k}\right\}\right)$ be the tube resulting from the optimal solution of (7) at time $k$, which satisfies the state constraints $\vec{\mathbb{X}}_k$ by construction, i.e., $\forall i \in \mathbb{N}_{[0,N-1]} : X_{i|k} \subseteq \mathbb{X}_{i|k}$. Furthermore, $X_{N|k} \subseteq \breve{\mathbb{X}}$ which follows from $X_{N|k} \subseteq X_f$ and $X_f \subseteq \breve{\mathbb{X}}$ (see Definition 5). After applying $\Pi_{0|k}$ to the system, at time $k+1$ a tube

$$\begin{aligned} \mathbf{T}_{k+1}^\circ &= \left(\left\{X_{0|k+1}^\circ, \ldots, X_{N|k+1}^\circ\right\}, \left\{\Pi_{0|k+1}^\circ, \ldots, \Pi_{N-1|k+1}^\circ\right\}\right) \\ &= \left(\left\{X_{1|k}, X_{2|k}, \ldots, X_{N-1|k}, X_f, G\left(X_f|\kappa\right)\right\},\right. \\ &\quad \left.\left\{\Pi_{1|k}, \ldots, \Pi_{N-1|k}, \kappa\right\}\right) \end{aligned}$$

can be constructed. Because of Step 7 of Algorithm 1, to be feasible this must satisfy $\forall i \in \mathbb{N}_{[0,N-2]} : X_{i|k+1}^\circ \subseteq \mathbb{X}_{i+1|k}$ and $X_{N-1|k}^\circ \subseteq \breve{\mathbb{X}}$: this holds by construction. Through the definition of $\vec{\Theta}_{k+1}$ in Step 3 of Algorithm 1 it is guaranteed that for all $i \in \mathbb{N}_{[0,N-2]}$, the controllers $\Pi_{i|k+1}^\circ = \Pi_{i+1|k}$ map $X_{i|k+1}^\circ$ into $X_{i+1|k+1}^\circ$. Since $X_{N-1|k+1}^\circ = X_f \subseteq \breve{\mathbb{X}}$ we have $\Theta_{N-1|k+1} \subseteq T\left(\breve{\mathbb{X}}\right)$ and therefore $\Pi_{N-1|k+1}^\circ(\cdot,\cdot) = \kappa(\cdot,\cdot)$ maps $X_{N|k+1}^\circ = X_f$ into $G\left(X_f|\kappa\right) \subseteq \lambda X_f$ according to (16)-(17). Thus $\mathbf{T}_{k+1}^\circ$ is feasible at $k+1$. Since (7) only optimizes over parameterized tubes satisfying Definition 4, there must additionally exist parameters $\left(p_{f|k}, p_{f|k+1}\right)$ such that $\bar{P}\left(p_{f|k}\right) = \left(X_f, \kappa\right)$ and $\bar{P}\left(p_{f|k+1}\right) = \left(G\left(X_f|\kappa\right), *\right)$ where $*$ is an irrelevant quantity. This is guaranteed by the choice that $S = X_f$, and so $\mathcal{D}_N\left(x_{0|k+1}, \vec{\Theta}_{k+1}, \vec{\mathbb{X}}_{k+1}\right) \neq \emptyset$.

(ii) Given recursive feasibility, this property is satisfied by construction of Step 7 in Algorithm 1.

(iii) Based on the feasible but non-optimal $\mathbf{T}_{k+1}^\circ$ derived above and on Lemmas 2-3, the value function $V(\cdot,\cdot,\cdot)$ can be shown to be decreasing along closed-loop trajectories:

$$\begin{aligned} &V\left(x_{0|k+1}, \vec{\Theta}_{k+1}, \vec{\mathbb{X}}_{k+1}\right) - V\left(x_{0|k}, \vec{\Theta}_k, \vec{\mathbb{X}}_k\right) \\ &\leq \ell_p\left(X_f, \Pi_f\right) + F_{k+1}\left(G_{f|k}\left(X_{f|k}\right)\right) - F_k\left(X_{N|k}\right) \\ &\quad + \sum_{i=1}^{N-1} \ell_p\left(X_{i|k}, \Pi_{i|k}\right) - \sum_{i=0}^{N-1} \ell_p\left(X_{i|k}, \Pi_{i|k}\right) \\ &\leq -\ell_p\left(X_{0|k}, \Pi_{0|k}\right) \leq -\underline{s}_\ell\left(d_H^0\left(X_{0|k}\right)\right). \end{aligned}$$

With Lemma 4 this implies that $V(\cdot,\cdot,\cdot)$ is a Lyapunov function for the closed-loop system [17, Theorem 2]. ■

## REFERENCES


[1] W. J. Rugh and J. S. Shamma, "Research on gain scheduling," *Automatica*, vol. 36, pp. 1401–1425, 2000.

[2] R. Tóth, *Modeling and Identification of Linear Parameter-Varying Systems*. Springer, 2010.

[3] H. S. Abbas, R. Tóth *et al.*, "Embedding of Nonlinear Systems in a Linear Parameter-Varying Representation," in *Proc. of the 19th IFAC World Congress*, 2014, pp. 6907–6913.

[4] Y. Lu and Y. Arkun, "A scheduling quasi–min-max model predictive control algorithm for nonlinear systems," *Journal of Process Control*, vol. 12, pp. 589–604, 2002.

[5] L. Chisci, P. Falugi, and G. Zappa, "Gain-scheduling MPC of nonlinear systems," *Int. J. of Robust and Nonlinear Control*, vol. 13, pp. 295–308, 2003.

[6] A. Casavola, D. Famularo, and G. Franzè, "Predictive control of constrained nonlinear systems via LPV linear embeddings," *Int. J. of Robust and Nonlinear Control*, vol. 13, pp. 281–294, 2003.

[7] P. S. G. Cisneros, S. Voss, and H. Werner, "Efficient Nonlinear Model Predictive Control via quasi-LPV Representation," in *Proc. of the 55th IEEE Conference on Decision and Control*, 2016, pp. 3216–3221.

[8] Y. I. Lee, B. Kouvaritakis, and M. Cannon, "Constrained receding horizon predictive control for nonlinear systems," *Automatica*, vol. 38, pp. 2093–2102, 2002.

[9] M. Cannon, J. Buerger *et al.*, "Robust Tubes in Nonlinear Model Predictive Control," *IEEE Transactions on Automatic Control*, vol. 56, pp. 1942–1947, 2011.

[10] H. S. Abbas, R. Tóth *et al.*, "LPV Modeling of Nonlinear Systems: A Multi-Path Feedback Linearization Approach," *Submitted to IEEE Transactions on Automatic Control*, 2017.

[11] J. Hanema, R. Tóth, and M. Lazar, "Tube-based anticipative model predictive control for linear parameter-varying systems," in *Proc. of the 55th IEEE Conference on Decision and Control*, 2016, pp. 1458–1463.

[12] W. Langson, I. Chryssochoos *et al.*, "Robust model predictive control using tubes," *Automatica*, vol. 40, pp. 125–133, 2004.

[13] S. V. Raković, B. Kouvaritakis *et al.*, "Homothetic tube model predictive control," *Automatica*, vol. 48, pp. 1631–1638, 2012.

[14] D. Limon, T. Alamo *et al.*, "Interval Arithmetic in Robust Nonlinear MPC," in *Assessment and Future Directions of Nonlinear Model Predictive Control*. Springer, 2007, pp. 317–326.

[15] T. Besselmann, J. Löfberg, and M. Morari, "Explicit MPC for LPV Systems: Stability and Optimality," *IEEE Transactions on Automatic Control*, vol. 57, pp. 2322–2332, 2012.

[16] J. Hanema, M. Lazar, and R. Tóth, "Stabilizing Tube-Based Model Predictive Control: Terminal Set and Cost Construction for LPV Systems," *Automatica*, vol. 85, pp. 137–144, 2017.

[17] D. Aeyels and J. Peuteman, "A New Asymptotic Stability Criterion for Nonlinear Time-Variant Differential Equations," *IEEE Transactions on Automatic Control*, vol. 43, pp. 968–971, 1998.