Nonlinear Model Predictive Control based on K-step Control Invariant Sets

Zhixin Zhao, Antoine Girard, Sorin Olaru

Nonlinear Model Predictive Control based on K-step Control Invariant Sets*

Zhixin Zhao[a,*], Antoine Girard[a], Sorin Olaru[a]

[a] *Université Paris-Saclay, CNRS, CentraleSupélec, Laboratoire des Signaux et Systèmes, 3 Rue Joliot Curie, Gif-sur-Yvette, 91190, France*

Abstract

One of the fundamental issues in Nonlinear Model Predictive Control (NMPC) is to be able to guarantee the recursive feasibility of the underlying receding horizon optimization. In other terms, the primary condition for a safe NMPC design is to ensure that the closed-loop solution remains indefinitely within the feasible set of the optimization problem. This issue can be addressed by introducing a terminal constraint described in terms of a control invariant set. However, the control invariant sets of nonlinear systems are often impractical to use or even to construct due to their complexity. The K-step control invariant sets are representing generalizations of the classical one-step control invariant sets and prove to retain the useful properties for MPC design, but often with simpler representations, and thus greater applicability. In this paper, a novel NMPC scheme based on K-step control invariant sets is proposed. We employ symbolic control techniques to compute a K-step control invariant set and build the NMPC framework by integrating this set as a terminal constraint, thereby ensuring recursive feasibility.

Keywords: Nonlinear Model Predictive Control, Symbolic control, Safety

1. Introduction

Model predictive control (MPC) is a widely used control strategy [1] in which the current control action is obtained by solving, at each sampling instant, a finite horizon open-loop control problem, using the current state of the plant as the initial state. The iterative procedure yields an optimal control sequence [2], and the first component in this sequence is applied to the plant before reiterating the procedure thus obtaining a closed-loop control formulation [3].

One of the fundamental problems in MPC is that of providing guarantees for recursive feasibility of the receding horizon optimization [4]. This desideratum can be translated in terms of a certificate that the closed-loop solutions remain indefinitely within a safe set which in turn represents a classical notion of positive set-invariance [5]. Recursive feasibility is classically obtained by introducing a control invariant set as terminal constraint at the end of the prediction horizon [6]. However, for nonlinear systems, the construction of such sets is difficult. Some approaches involve solving partial differential equations [7], or recursing to interval arithmetics [8, 9] and result in complex set representations that are unsuitable for inclusion in an optimization problem. For polynomial systems, control invariant sets can be approximated using semi-algebraic sets [10]. Other approaches constrain the control invariant set to be convex [11, 12, 13], which may be a conservative assumption.

To address this issue, several research works point to relaxations, extensions, or alternatives to the classical control invariant set. [14, 15] impose only periodic validation of the set-membership conditions and thus provides a relaxation with respect to the classical invariance condition. A similar approach is followed in [16, 17], where the time-invariant terminal constraint is replaced by a sequence of constraints which are periodically enforced. In a broader sense, cyclic invariance has been studied for time-delay systems in [18], while [19] uses an invariant family of sets for decentralized control and generalize the periodic sequences of sets by dropping the order relationship. In a different perspective, the integration of a less rigid set-membership constraint within MPC has been explored in [20]. This study incorporate inner-outer pairs of sets [21] instead of classical control invariant sets, thus ensuring stability across various control and prediction horizons. In this context, the K-step control invariant sets were recently proposed for the constrained control design [22]. The K-step control invariant sets ensure the existence of a control sequence such that the system trajectory cannot leave the given set for more than K time steps and can be linked to the $K-$step constructiveness studied in early research works [23].

The first contribution of the present paper is to propose Nonlinear Model Predictive Control (NMPC) scheme based on K-step control invariant sets. A theoretical comparison of NMPC with control invariant sets and K-step control invariant sets is provided to determine the equivalence of the two approaches in terms of closed-loop behavior. From the point of view of practical construction, a symbolic control design procedure is employed. It leverages on the recent advancements on the computation of the maximum control invariant set in this framework proposed in [24, 25]. As one of the main contributions of the present paper, it is shown that K-step control invariant sets can be computed using a similar principles. It is worth to be mentioned that symbolic control is a computational ap-

*This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

*Corresponding author.

Email address: `Zhao.zhixin@centralesupelec.fr` (Zhixin Zhao)

proach for synthesizing controllers for general nonlinear systems with state and input constraints and thus is not suffering from the structural constraints of the classical invariant set constructions. More precisely, the continuous dynamics of the system are approximated using a finite-state dynamical system called symbolic abstraction. The key advantage of symbolic abstractions is the ability to use algorithmic techniques to synthesize controllers that enforce various specifications such as safety, reachability, attractivity, etc. [26]. The newly proposed NMPC framework, which combines symbolic control through the use of K-step invariant sets, capitalizes on the control performance benefits offered by NMPC while ensuring safety.

The paper is structured as follows. Section II provides an introduction to the relevant preliminary theory. Section III provides the novel NMPC scheme. Section IV outlines the way in which the NMPC framework demonstrated in Section III can be accomplished through offline and online computation. Finally, Section V offers a numerical example that demonstrates the computation of K-step control invariant sets and their use in a reference tracking problem.

2. Preliminaries

In this section, we recall some classical results on NMPC, which will be useful for subsequent discussions.

2.1. Nonlinear model predictive control

We consider a nonlinear discrete-time system of the form:

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

where $x_t \in \mathbb{R}^n$ is the state of the system and $u_t \in \mathbb{R}^m$ is the control input.

The constrained control problem accounts for state and input limitations given by compact sets $\mathbb{X} \subseteq \mathbb{R}^n$ and $\mathbb{U} \subseteq \mathbb{R}^m$. The goal is to design an NMPC scheme enforcing the constraints while optimizing some performance criteria. For that purpose, we consider at each discrete-time instant $t \in \mathbb{N}$ the following optimization problem parameterized by the current state vector assumed to be measurable $x \in \mathbb{X}$:

$$\begin{array}{ll} & \min_{\mathbf{u}_t^{[0,N-1]}} J(t, x_t^0, \underbrace{u_t^0, \ldots, u_t^{N-1}}_{\mathbf{u}_t^{[0,N-1]}}, \underbrace{x_t^1, \ldots, x_t^N}_{\mathbf{x}_t^{[1,N]}}) \\ & \begin{array}{ll} x_t^0 = x, & \\ x_t^{i+1} = f(x_t^i, u_t^i), & i = 0, \ldots, N-1 \end{array} \\ \text{s.t.} & \begin{array}{ll} u_t^i \in \mathbb{U}, & i = 0, \ldots, N-1 \\ x_t^i \in \mathbb{X}, & i = 0, \ldots, N \\ x_t^N \in \mathbb{X}_0. & \end{array} \end{array} \tag{2}$$

In this classical NMPC formulation (2), N is a positive integer defining the horizon for the evaluation of a (possibly time-varying) performance cost function

$$J : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^{N\times m} \times \mathbb{R}^{N\times n} \to \mathbb{R}.$$

$J(t, x_t^0, \mathbf{u}_t^{[0,N-1]}, \mathbf{x}_t^{[1,N]})$ is parameterized by the current state x_t^0 while $\mathbf{u}_t^{[0,N-1]}$ and $\mathbf{x}_t^{[1,N]}$ represent the optimization arguments linked through the constraints. Finally, $\mathbb{X}_0 \subseteq \mathbb{X}$ is a compact set defining terminal constraints.

The *feasible set* of (2), is the set of states $x \in \mathbb{X}$ such that (2) has a feasible solution. We assume that constraints in (2) are time-invariant, i.e. that $\mathbb{X}$, $\mathbb{U}$ and $\mathbb{X}_0$ do not depend on time t, so the feasible set does not depend on the time either. We denote the feasible set by $\mathbb{F}_{\mathbb{X}_0}$ to emphasize its dependence on the set of terminal constraints $\mathbb{X}_0$. For $x \in \mathbb{F}_{\mathbb{X}_0}$, let $\bar{\mathbf{u}}_t^{[0,N-1]} = \left\{\bar{u}_t^0, \ldots, \bar{u}_t^{N-1}\right\}$ be a minimizer of (2). We assume that a minimizer exists, even though it may not be unique. This is the case e.g. if $J(t, \cdot, \cdot, \cdot)$ and f are continuous, due to the compactness of sets involved in (2). Then, we denote $\lambda_{\mathbb{X}_0}(t, x)$ the optimal value of (2) and the control action to be effectively implemented as the system input is selected as the first component of an optimal argument, i.e. $u_t \in \mu_{\mathbb{X}_0}(t, x_t)$, where:

$$\mu_{\mathbb{X}_0}(t, x) = \left\{ \bar{u}_t^0 \in \mathbb{U} \middle| \begin{array}{c} \bar{\mathbf{u}}_t^{[0,N-1]} = \left\{\bar{u}_t^0, \ldots, \bar{u}_t^{N-1}\right\} \\ \text{is a minimizer of (2)} \end{array} \right\}. \tag{3}$$

We can establish some monotonicity properties with respect to the terminal constraint $\mathbb{X}_0$ as follows.

Proposition 1. *Given a set $\mathbb{X}_0'$, such that $\mathbb{X}_0 \subseteq \mathbb{X}_0' \subseteq \mathbb{X}$, then*

- $\mathbb{F}_{\mathbb{X}_0} \subseteq \mathbb{F}_{\mathbb{X}_0'}$
- *For all $t \in \mathbb{N}$, for all $x \in \mathbb{F}_{\mathbb{X}_0}$*

$$\lambda_{\mathbb{X}_0'}(t, x) \le \lambda_{\mathbb{X}_0}(t, x).$$

Proof. These claims follow from the fact that $\mathbb{X}_0 \subseteq \mathbb{X}_0' \subseteq \mathbb{X}$ enlarges the feasible set. All the optimal solutions with terminal constraints $\mathbb{X}_0$ are feasible solutions with terminal constraints $\mathbb{X}_0'$. □

Consider now the closed-loop system given by (1) and (2) with

$$u_t \in \mu_{\mathbb{X}_0}(t, x_t). \tag{4}$$

We denote by $\bar{\mathbb{F}}_{\mathbb{X}_0}$ the *recursive feasible set* of (1)-(2)-(4), that is the set of states $x_0 \in \mathbb{X}$ such that $x_t \in \mathbb{F}_{\mathbb{X}_0}$ for all $t \in \mathbb{N}$. Clearly, $\bar{\mathbb{F}}_{\mathbb{X}_0} \subseteq \mathbb{F}_{\mathbb{X}_0}$. To show that the closed-loop system is well-posed and satisfies the constraints at all time, it is important to guarantee that $\bar{\mathbb{F}}_{\mathbb{X}_0} \neq \emptyset$.

2.2. Recursive feasibility using control invariant sets

Let us recall a classical result on the use of control invariant sets as terminal constraints in (2), see e.g. [6].

Definition 1. $\mathbb{X}_0 \subseteq \mathbb{X}$ *is a control invariant set of* (1) *if*

$$\forall x \in \mathbb{X}_0,\ \exists u \in \mathbb{U},\ f(x, u) \in \mathbb{X}_0.$$

A control invariant set $\mathbb{X}_0$ is said to be the maximal control invariant set of (1) *if any control invariant set is a subset of $\mathbb{X}_0$.*

The following proposition is reminiscent from classical results (see e.g. [27, Theorem 5.4]). It is therefore stated without proof.

Proposition 2. *If* $\mathbb{X}_0 \subseteq \mathbb{X}$ *is a control invariant set of* (1) *then* $\mathbb{X}_0 \subseteq \mathbb{F}_{\mathbb{X}_0}$ *and* $\bar{\mathbb{F}}_{\mathbb{X}_0} = \mathbb{F}_{\mathbb{X}_0}$. *Moreover, if* $\mathbb{X}_0$ *is the maximal control invariant set, then* $\mathbb{F}_{\mathbb{X}_0} = \mathbb{X}_0$.

Control invariant sets are thus a powerful tool to enforce recursive feasibility in NMPC scheme. From Proposition 1, it follows that considering larger control invariant sets leads to larger feasible sets and better performances. Hence, it is important to consider a control invariant set that is as close as possible to the maximal one. Unfortunately, for nonlinear systems with potentially nonconvex sets of constraints, the maximal control invariant set maybe very complex. Accurate approximations require complex set representations (see e.g. [7, 8, 9]) and are therefore impractical to consider in real-time optimization.

3. NMPC scheme based on *K*-step control invariant sets

In this section, we present an alternative NMPC scheme based on K-step control invariant sets. We then compare the feasibility and the performances with respect to the approach presented in the previous section.

3.1. K-step control invariant sets

Let us consider system (1) with state and input constraints given by $\mathbb{X}$ and $\mathbb{U}$. Given a set $\mathbb{T} \subseteq \mathbb{X}$ we define the set of *controllable predecessors* of $\mathbb{T}$ as

$$\mathsf{CPre}(\mathbb{T}) = \{x \in \mathbb{X} | \ \exists u \in \mathbb{U},\ f(x,u) \in \mathbb{T}\}.$$

Then, for $k \in \mathbb{N}$, we define $\mathsf{CPre}_k(\mathbb{T})$ inductively as follows:

$$\mathsf{CPre}_0(\mathbb{T}) = \mathbb{T},\ \mathsf{CPre}_{k+1}(\mathbb{T}) = \mathsf{CPre}\left(\mathsf{CPre}_k(\mathbb{T})\right),\ k \in \mathbb{N}.$$

Finally, for $K \geq 1$, let

$$\mathsf{CPre}_{[1:K]}(\mathbb{T}) = \bigcup_{k=1}^{K} \mathsf{CPre}_k(\mathbb{T}).$$

Intuitively, $\mathsf{CPre}_{[1:K]}(\mathbb{T})$ denotes the set of states from which (1) can reach the set $\mathbb{T}$ in K steps or less, while respecting state and input constraints.

Definition 2. $\mathbb{T} \subseteq \mathbb{X}$ *is a* K-step control invariant set *of* (1) *if* $\mathbb{T} \subseteq \mathsf{CPre}_{[1:K]}(\mathbb{T})$.

In words, $\mathbb{T}$ is K-step control invariant set if from any state of $\mathbb{T}$ there exists a control sequence allowing to come back to $\mathbb{T}$ in at most K time steps (see the weak invariance notion in [22]).

Proposition 3. *Let* $\mathbb{T} \subseteq \mathbb{X}$ *be a K-step control invariant set of* (1). *Then,* $\mathsf{CPre}_{[1:K]}(\mathbb{T})$ *is a control invariant set of* (1).

Proof. Let us consider $x_0 \in \mathsf{CPre}_{[1:K]}(\mathbb{T})$. Then, there exists an input sequence $u_0, \dots, u_{k-1} \in \mathbb{U}$ with $k \in \{1, \dots, K\}$ such that the associated trajectory of (1), $x_0,\ x_1, \dots, x_k$ satisfies $x_i \in \mathbb{X}$, for all $i \in \{1, \dots, k\}$ and $x_k \in \mathbb{T}$. Then, it follows that $x_1 \in \mathsf{CPre}_{k-1}(\mathbb{T})$. If $k > 1$, we get that $x_1 \in \mathsf{CPre}_{[1:K]}(\mathbb{T})$. If $k = 1$, then $x_1 \in \mathbb{T}$, which since $\mathbb{T}$ is K-step control invariant set yields $x_1 \in \mathsf{CPre}_{[1:K]}(\mathbb{T})$. It follows that $\mathsf{CPre}_{[1:K]}(\mathbb{T})$ is a control invariant set of (1). □

3.2. A novel NMPC scheme

We now present the NMPC scheme based on K-step control invariant sets. We consider at all time $t \in \mathbb{N}$ the following optimization problem parameterized by $x \in \mathbb{X}$:

$$\begin{array}{ll} & \min_{\mathbf{u}_t^{[0,N+K-1]}} J(t, x_t^0, \mathbf{u}_t^{[0,N-1]}, \mathbf{x}_t^{[1,N]}) \\ & x_t^0 = x, \\ & x_t^{i+1} = f(x_t^i, u_t^i), \quad i = 0, \dots, N+K-1 \\ \text{s.t.} & u_t^i \in \mathbb{U}, \quad i = 0, \dots, N+K-1 \\ & x_t^i \in \mathbb{X}, \quad i = 0, \dots, N+K \\ & \exists k \in \{1, \dots, K\}, \quad x_t^{N+k} \in \mathbb{T}. \end{array} \tag{5}$$

In (5), N and K are positive integers, J, $\mathbb{X}$ and $\mathbb{U}$ are the same as in (2) and $\mathbb{T} \subseteq \mathbb{X}$ is a compact set. Let us remark that while the constraints in (5) involves the predicted behavior of (1) on a time horizon of length $N + K$, the cost function only involves the value of the input and state on the first N time steps.

We denote by $\mathbb{F}_{\mathbb{T}}^K$ the *feasible set* of (5), that is the set of states $x \in \mathbb{X}$ such that (5) has a feasible solution. Note that constraints defined by sets $\mathbb{X}$, $\mathbb{U}$ and $\mathbb{T}$ in (5) are time-invariant so the feasible set does not depend on the time t. For $x \in \mathbb{F}_{\mathbb{T}}^K$, let $\bar{\mathbf{u}}_t^{[0,N+K-1]}$ be a minimizer of (5). We assume that a minimizer exists, even though it may not be unique. As for (2), this is the case e.g. if $J(t, ., ., .)$ and f are continuous, due to the compactness of sets involved in (5). Then, we denote $\lambda_{\mathbb{T}}^K(t,x)$ the optimal value of (5) and the non-empty set of optimizers:

$$\mu_{\mathbb{T}}^K(t,x) = \left\{ \bar{u}_t^0 \in \mathbb{U} \left| \begin{array}{c} \bar{\mathbf{u}}_t^{[0,N+K-1]} = \left\{\bar{u}_t^0, \dots, \bar{u}_t^{N+K-1}\right\} \\ \text{is a minimizer of (5)} \end{array} \right. \right\} \tag{6}$$

Consider now the closed-loop system given by (1) with

$$u_t \in \mu_{\mathbb{T}}^K(t, x_t). \tag{7}$$

We denote by $\bar{\mathbb{F}}_{\mathbb{T}}^K$ the *recursive feasible set* of (1)-(5)-(7), that is the set of all states $x_0 \in \mathbb{X}$ such that $x_t \in \mathbb{F}_{\mathbb{T}}^K$ for all $t \in \mathbb{N}$ subject to $x_{t+1} = f(x_t, u_t)$ when $u_t \in \mu_{\mathbb{T}}^K(t, x_t)$. Clearly, $\bar{\mathbb{F}}_{\mathbb{T}}^K \subseteq \mathbb{F}_{\mathbb{T}}^K$. The next result describes the main property of the proposed NMPC scheme.

Theorem 1. *Let* $\mathbb{T} \subseteq \mathbb{X}$ *be a K-step control invariant set of* (1). *Then,*

- $\bar{\mathbb{F}}_{\mathbb{T}}^K = \mathbb{F}_{\mathbb{T}}^K = \mathbb{F}_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}$*;*
- *For all* $x_0 \in \mathbb{F}_{\mathbb{T}}^K$,

$$\mu_{\mathbb{T}}^K(t, x_t) = \mu_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}(t, x_t)$$

  and

$$\lambda_{\mathbb{T}}^K(t, x_t) = \lambda_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}(t, x_t)$$

  for all $t \in \mathbb{N}$.

Proof. Let $x \in \mathbb{F}_{\mathbb{T}}^K$, let $\bar{\mathbf{u}}_t^{[0,N+K-1]}$ be a minimizer of (5). Then, considering the associated optimal sequence $\bar{\mathbf{x}}_t^{[0,N+K]}$ with $\bar{x}_t^0 = x$, it is clear from the last constraint of (5) that $\bar{x}_t^N \in \mathsf{CPre}_{[1:K]}(\mathbb{T})$. Hence, $\bar{\mathbf{u}}_t^{[0,N-1]}$ is a feasible solution of (2) with

$$\mathbb{X}_0 = \mathsf{CPre}_{[1:K]}(\mathbb{T}).$$

This implies that $x \in \mathbb{F}_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}$ and that

$$\lambda^K_{\mathbb{T}}(t,x) \geq \lambda_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}(t,x).$$

Conversely, consider $x \in \mathbb{F}_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}$ and let $\bar{\mathbf{u}}_t^{[0,N-1]}$ be a minimizer of (2) with $\mathbb{X}_0 = \mathsf{CPre}_{[1:K]}(\mathbb{T})$ and let $\bar{\mathbf{x}}_t^{[0,N]}$ with $\bar{x}_t^0 = x$ be the associated trajectory of (1). Then, $\bar{x}_t^N \in \mathsf{CPre}_{[1:K]}(\mathbb{T})$ which means that there exists an input sequence $\bar{\mathbf{u}}_t^{[N,N+k-1]} \in \mathbb{U}$, with $k \in \{1,\ldots,K\}$ such that $\bar{\mathbf{x}}_t^{[N,N+k]} \in \mathbb{X}$ and $\bar{x}_t^{N+k} \in \mathbb{T}$. Since $\mathbb{T}$ is a K-step controlled invariant we have that $\mathbb{T} \subseteq \mathsf{CPre}_{[1:K]}(\mathbb{T})$. Moreover, from Proposition 3, we have that $\mathsf{CPre}_{[1:K]}(\mathbb{T})$ is a control invariant set of (1). If $k < K$, it follows that there exists an input sequence $\bar{\mathbf{u}}_t^{[N+k,N+K-1]} \in \mathbb{U}$ such that

$$\bar{\mathbf{x}}_t^{[N+k+1,N+K]} \in \mathsf{CPre}_{[1:K]}(\mathbb{T}) \subseteq \mathbb{X}.$$

Hence, $\bar{\mathbf{u}}_t^{[0,N+K-1]}$ is a feasible solution of (5). This implies that $x \in \mathbb{F}^K_{\mathbb{T}}$ and that

$$\lambda^K_{\mathbb{T}}(t,x) \leq \lambda_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}(t,x).$$

Consequently, one can conclude that

$$\mathbb{F}^K_{\mathbb{T}} = \mathbb{F}_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}$$

and that for all $x \in \mathbb{F}^K_{\mathbb{T}}$, $\lambda^K_{\mathbb{T}}(t,x) = \lambda_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}(t,x)$. Moreover, it follows from above that the first N elements of the minimizers of (5) coincide with the minimizers of (2) with $\mathbb{X}_0 = \mathsf{CPre}_{[1:K]}(\mathbb{T})$. This yields for all

$$\mu^K_{\mathbb{T}}(t,x) = \mu_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}(t,x), \forall x \in \mathbb{F}^K_{\mathbb{T}}.$$

Then, the closed-loop behaviors of (1)-(2)-(4) coincide with those of (1)-(5)-(7). This yields that

$$\bar{\mathbb{F}}^K_{\mathbb{T}} = \bar{\mathbb{F}}_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}.$$

Moreover, since $\mathsf{CPre}_{[1:K]}(\mathbb{T})$ is a control invariant set of (1), we get from Proposition 2 that

$$\bar{\mathbb{F}}_{\mathsf{CPre}_{[1:K]}(\mathbb{T})} = \mathbb{F}_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}$$

which together with $\mathbb{F}^K_{\mathbb{T}} = \mathbb{F}_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}$ gives $\bar{\mathbb{F}}^K_{\mathbb{T}} = \mathbb{F}^K_{\mathbb{T}}$. □

The previous result shows the equivalence in term of closed-loop behaviors, between NMPC scheme (1)-(5)-(7) and (1)-(2)-(4) with $\mathbb{X}_0 = \mathsf{CPre}_{[1:K]}(\mathbb{T})$. It also implies the following result:

Corollary 1. *Let $\mathbb{X}_0 \subseteq \mathbb{X}$ be a control invariant set of* (1)*, let us consider $\mathbb{T} \subseteq \mathbb{X}_0$ such that $\mathbb{X}_0 \subseteq \mathsf{CPre}_{[1:K]}(\mathbb{T})$. Then,*

- $\bar{\mathbb{F}}^K_{\mathbb{T}} = \mathbb{F}^K_{\mathbb{T}}$ *and* $\mathbb{F}_{\mathbb{X}_0} \subseteq \mathbb{F}^K_{\mathbb{T}}$*;*
- *For all* $x_0 \in \mathbb{F}_{\mathbb{X}_0}$, $\lambda_{K,\mathbb{T}}(t,x_t) \leq \lambda_{\mathbb{X}_0}(t,x_t), \forall t \in \mathbb{N}$.

Proof. Since $\mathbb{T} \subseteq \mathbb{X}_0 \subseteq \mathbb{X}$ and $\mathbb{X}_0 \subseteq \mathsf{CPre}_{[1:K]}(\mathbb{T})$, it follows that $\mathbb{T}$ is a K-step control invariant set. Then, we get from Theorem 1 that

$$\bar{\mathbb{F}}^K_{\mathbb{T}} = \mathbb{F}^K_{\mathbb{T}} = \mathbb{F}_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}$$

and for all $t \in \mathbb{N}$, for all $x \in \mathbb{F}^K_{\mathbb{T}}$, $\lambda^K_{\mathbb{T}}(t,x) = \lambda_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}(t,x)$. Then, $\mathbb{X}_0 \subseteq \mathsf{CPre}_{[1:K]}(\mathbb{T})$ gives from Proposition 1 that

$$\mathbb{F}_{\mathbb{X}_0} \subseteq \mathbb{F}_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}$$

and for all $t \in \mathbb{N}$, for all $x \in \mathbb{F}_{\mathbb{X}_0}$, $\lambda_{\mathsf{CPre}_{[1:K]}(\mathbb{T})}(t,x) \leq \lambda_{\mathbb{X}_0}(t,x)$. From above, it follows that

$$\mathbb{F}_{\mathbb{X}_0} \subseteq \mathbb{F}^K_{\mathbb{T}}$$

and for all $t \in \mathbb{N}$ and $x \in \mathbb{F}_{\mathbb{X}_0}$, $\lambda^K_{\mathbb{T}}(t,x) \leq \lambda_{\mathbb{X}_0}(t,x)$. □

In this section, we have compared the properties of the established NMPC schemes based on control invariant sets and those of the novel MPC scheme based on K-step control invariant sets. Corollary 1 indicates that the approach based on K-step control invariant sets can lead to larger recursive feasible sets and to better performances than an approach based on classical $1-$step control invariant sets. This discussion rejoins the analysis in [21] on the inner-outer approximation of robust control invariant sets.

4. Algorithms for effective implementation

In this section, we discuss the computational aspects of the NMPC scheme based on K-step control invariant sets. Firstly, we focus on offline computations and describe a method to compute K-steps control invariant sets using symbolic control techniques. Secondly, we discuss the online computation aspects and we describe a method to initialize the optimization problem (5) with a feasible solution.

4.1. Offline computation

This subsection describes the offline computational part required to apply the framework introduced in the previous section. This includes the use of symbolic control to compute the K-step control invariant set and symbolic controllers that will be used in online computations. First, we briefly recall the symbolic control technique (see e.g. [24, 25] for more details).

4.1.1. Symbolic control

Symbolic control involves discretizing the state space and the control input, representing the set of states as a finite number of symbols that transition between each other under the influence of control input. The first step involves constructing the symbolic system dynamics of (1). Let us consider:

- A finite partition $(\mathbb{X}_q)_{q\in\mathbb{Q}}$ of $\mathbb{R}^n$ such that $\mathbb{Q} = \mathbb{Q}_{\mathbb{X}} \cup \{q_{\text{out}}\}$ and
$$\bigcup_{q\in\mathbb{Q}_{\mathbb{X}}} \mathbb{X}_q \subseteq \mathbb{X}$$
where $\{q_{out}\}$ is the complement set of $\mathbb{Q}_{\mathbb{X}}$ which represents the set of unsafe symbolic states.

- A finite number of samples $(u_p)_{p\in\mathbb{P}}$ providing a finite subset of $\mathbb{U}$.

The symbolic model is described by the dynamics:

$$q_{t+1} \in F(q_t, p_t),\ q_t \in \mathbb{Q},\ p_t \in \mathbb{P} \tag{8}$$

where $\mathbb{Q}$ and $\mathbb{P}$ are the finite sets of symbolic states and inputs; $\mathrm{F} : \mathbb{Q} \times \mathbb{P} \to 2^{\mathbb{Q}}$ is the set-valued transition map defined by

$$q_+ \in F(q, p) \iff \mathbb{X}_{q_+} \cap \hat{f}(\mathbb{X}_q, u_p) \neq \emptyset$$

where $\hat{f}$ is an over-approximation of f ensuring:

$$f(\mathbb{X}_q, u_p) \subseteq \hat{f}(\mathbb{X}_q, u_p),\ \forall q \in \mathbb{Q},\ p \in \mathbb{P}.$$

Such over-approximations can be computed efficiently using for instance interval reachability analysis (see [28]). For each symbolic state q and input p, we compute the over-approximation of the successor of $\mathbb{X}_q$ under the influence of each input u_p using system dynamics. One can show that the dynamics of (8) is an over-approximation of that of (1), so control invariant sets and K-step control invariant sets of (1) can be obtained from those of (8), see e.g. [26].

In order to compute K-step control invariant sets of (8), we need to first compute its control invariant set.

4.1.2. Computation of control invariant set

Let us remark that (8) has finite sets of states and inputs. Then, its maximal control invariant set can be easily computed using a fixed-point algorithm [24], where starting from the set $\mathbb{Q}_{\mathbb{X}}$, we iteratively remove the states from which we cannot keep the system state in $\mathbb{Q}_{\mathbb{X}}$. Then, since the dynamics of (8) is an over-approximation of that of (1), the maximal control invariant set of (8) provides a control invariant set of (1).

4.1.3. Computation of K-step control invariant sets

Once we obtain the maximum control invariant set, our next step is to select a subset as large as possible as a candidate for the K-step control invariant set. Choosing a K-step control invariant set with a simple shape will facilitate its integration into the NMPC framework. Therefore, one can choose a hyperrectangle inside the control invariant set that is as large as possible[1] and denote it as $\mathbb{T}$. Using symbolic control, the set $\mathsf{CPre}_{[1:K]}(\mathbb{T})$ can be effectively computed. Further, a progressive increase of the value of K is considered up to the validation of the relationship in Definition 2: $\mathbb{T} \subseteq \mathsf{CPre}_{[1:K]}(\mathbb{T})$, which allows to certify that $\mathbb{T}$ is a K-step control invariant set.

4.1.4. Computation of symbolic controllers

As with most complex optimization problems, initial values have a considerable impact on the results in (5). Therefore we would like to provide the optimization problem with an initial value that satisfies the constraints at each moment, and allow the iterative procedure to focus on its improvement. One needs to compute the following controllers offline using a symbolic approach:

$$\begin{array}{l} C_i : Q \to U, \text{ such that} \\ \forall x \in \mathsf{CPre}_i(\mathbb{T}),\ F(q, C_i(q(x))) \in \mathsf{CPre}_{i-1}(\mathbb{T}) \\ i \in [1, K] \\ C^* : Q \to U, \text{ such that} \\ \forall x \in \mathsf{CPre}_{[1:K]}(\mathbb{T}),\ F(q, C^*(q(x))) \in \mathsf{CPre}_{[1:K]}(\mathbb{T}) \end{array} \tag{9}$$

where $q(x)$ denote the symbolic state $q \in Q$ such that $x \in \mathbb{X}_q$. Without entering into algorithmic details, is worth metioning that both controllers can be obtained while computing the K-step control invariant sets. Generally there is freedom in the design of such symbolic controllers that satisfy the above requirements (depending on the number of control symbols and the structure of the dynamics). No particular criterion is imposed here, a random choice or ad hoc criteria can be employed.

4.2. Online computation

This section presents the online computational part of NMPC based on the K-step control invariant sets.

4.2.1. Solving optimization problem (5)

If we denote problem (5) as $O(t, x_t)$ we can define a collection of simpler[2] optimization problems $O_k(t, x_t), k \in \{1, \ldots, K\}$:

$$\begin{array}{ll} & \min_{\mathbf{u}_t^{[0,N+K-1]}} J(t, x_t^0, \mathbf{u}_t^{[0,N-1]}, \mathbf{x}_t^{[1,N]}) \\ & x_t^0 = x, \\ & x_t^{i+1} = f(x_t^i, u_t^i), \quad i = 0, \ldots, N+K-1 \\ \text{s.t.} & u_t^i \in \mathbb{U}, \quad i = 0, \ldots, N+K-1 \\ & x_t^i \in \mathbb{X}, \quad i = 0, \ldots, N+K \\ & x_t^{N+k} \in \mathbb{T}. \end{array} \tag{10}$$

Figure 1 illustrates the fact that solving these K optimization problems online can be done either sequentially or concurrently. Then, we select the index k that minimizes the costs $\lambda_{\mathbb{T}}^k(t, x_t)$ and we return the first component of the sequence of inputs corresponding to the minimizer.

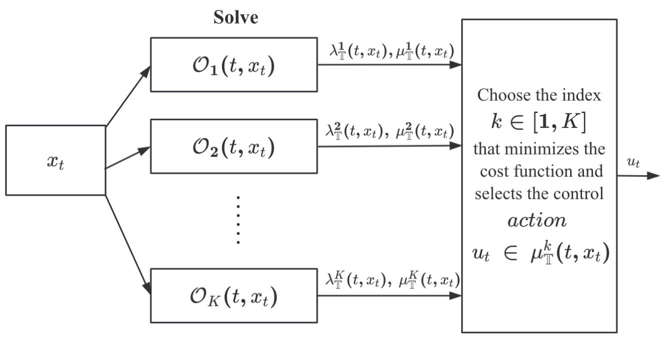


Figure 1: Practical implementation of (5)

[1]A measure for the enlargement needs to be defined and the largest candidate may not be unique thus providing degrees of freedom for the design. We don't dwell here on this aspect and simply assume the largest hyperrectagle has been selected by enlarging each direction sequentially.

[2]Simpler is understood here in terms of terminal constraints.

Algorithm 1: Initialization for t=0

Input : x_0 s.t. $x_0 \in \mathsf{CPre}_{[1:K]}(\mathbb{T}) \subseteq \mathbb{X}$
Output: Feasible control sequence at time 0: $\hat{\mathbf{u}}_0^{[0,N+K-1]}$

// Initialize $\hat{\mathbf{u}}_0^{[0,N-1]}$
$\hat{x}_0^0 \leftarrow x_0$;
for $j \leftarrow 1$ **to** N **do**
  $\hat{u}_0^{j-1} \leftarrow C^*(q(\hat{x}_0^{j-1}))$;
  $\hat{x}_0^{j} \leftarrow f(\hat{x}_0^{j-1}, \hat{u}_0^{j-1})$;
// Initialize $\hat{\mathbf{u}}_0^{[N,N+\bar{i}-1]}$
// We have $\hat{x}_0^{N+\bar{i}} \in \mathbb{T}$, $\bar{i}$ is the minimal step to reach $\mathbb{T}$
$\bar{i} \leftarrow min\{i|\hat{x}_0^N \in \mathsf{CPre}_i(\mathbb{T})\}$;
for $i \leftarrow 1$ **to** $\bar{i}$ **do**
  $\hat{u}_0^{N+i-1} \leftarrow C_{\bar{i}-i+1}(\hat{x}_0^{N+i-1})$;
  $\hat{x}_0^{N+i} \leftarrow f(\hat{x}_0^{N+i-1}, \hat{u}_0^{N+i-1})$;
// Initialize $\hat{\mathbf{u}}_0^{[N+\bar{i},N+K-1]}$
for $j \leftarrow N+\bar{i}$ **to** $N+K-1$ **do**
  $\hat{u}_0^{j} \leftarrow C^*(q(\hat{x}_0^{j}))$;
  $\hat{x}_0^{j+1} \leftarrow f(\hat{x}_0^{j}, \hat{u}_0^{j})$;

4.2.2. Initialization strategy for NMPC

For nonlinear optimization solvers to be successful, it is often required to provide a feasible initial point on which the solver can improve. The following proposition shows how this can be done using the symbolic controllers computed offline.

Proposition 4. *Given an initial position x_0 located within the set $\mathsf{CPre}_{[1:K]}(\mathbb{T})$, an initial control sequence for the optimization problem that ensures constraint satisfaction can be computed at time* 0 *using Algorithm 1, and at time $t \geq 1$ using Algorithm 2.*

Proof. For $t = 0$, $q(x_0)$ is the initial symbolic state, and the associated control $\hat{u}_0^0 = C^*(q(x_0))$. According to the definition of the controller C^*, we have $\hat{u}_0^0 \in \mathbb{U}$, and $\hat{x}_0^1 = f(x_0, \hat{u}_0^0) \in \mathsf{CPre}_{[1:K]}(\mathbb{T})$. In the same way one can calculate $\hat{\mathbf{u}}_0^{[1,N-1]}$, and all states in $\hat{\mathbf{x}}_0^{[1:N]} \in \mathsf{CPre}_{[1:K]}(\mathbb{T})$. Then the minimum number of steps to reach the set $\mathbb{T}$: $\bar{i} = min\{i|\hat{x}_0^N \in \mathsf{CPre}_i(\mathbb{T})\} \in [1, K]$ is available and $\hat{u}_0^N = C_{\bar{i}}(q(\hat{x}_0^N)$. According to the definition of controller C_i, we have $\hat{u}_0^N \in \mathbb{U}$ and $\hat{x}_0^{N+1} = f(\hat{x}_0^N, \hat{u}_0^N) \in \mathsf{CPre}_{\bar{i}-1}(\mathbb{T})$. By iteratively employing these controllers, we can obtain the control sequence $\hat{\mathbf{u}}_0^{[N,N+\bar{i}-1]}$ and we can ensure that the set $\mathbb{T}$ can be reached at $\hat{x}_0^{N+\bar{i}}$. Since $\mathbb{T} \subseteq \mathsf{CPre}_{[1:K]}(\mathbb{T})$ the available control C^* is used to fill in the sequence $\hat{\mathbf{u}}_0^{[N+\bar{i},N+K-1]}$.

For any instant $t \geq 1$, we have the solution of the optimization problem at instant $t-1$, which yields a control sequence $\mathbf{u}_{t-1}^{[0,N+K-1]} \in \mathbb{U}$, the corresponding states $\mathbf{x}_{t-1}^{[0,N+K]} \in \mathbb{X}$ and the state $x_{t-1}^{N+k} \in \mathbb{T}$. We let $\hat{\mathbf{u}}_t^{[0,N+k-2]} = \mathbf{u}_{t-1}^{1,N+k-1}$ then we can get $\hat{x}_t^{N+k-1} \in \mathbb{T}$ with $k \in [1, K]$. If k is equal to 1, one can find $\bar{i} = min\{i|\hat{x}_t^N \in \mathsf{CPre}_i(\mathbb{T})\} \in [1, K]$ then use the previous method to calculate the control sequence $\hat{\mathbf{u}}_t^{[N,N+K-1]}$. If k is not equal to 1, one can calculate $\hat{\mathbf{u}}_t^{[N+k-1,N+K-1]}$ with the controller C^*. Such initial control sequence obtained at time t will satisfy all the constraints. □

Algorithm 2: Initialization at any $t \geq 1$

Input : $\mathbf{u}_{t-1}^{[0,N+K-1]}$, $\mathbf{x}_{t-1}^{[0,N+K]}$, with $x_{t-1}^{N+k} \in \mathbb{T}$
Output: Feasible control sequence at time t: $\hat{\mathbf{u}}_t^{[0,N+K-1]}$

$\hat{\mathbf{u}}_t^{[0,N+k-2]} \leftarrow \mathbf{u}_{t-1}^{[1,N+k-1]}$;
// $\hat{x}_t^{N+k-1} = x_{t-1}^{N+k}$
if $k \neq 1$ **then**
  for $j \leftarrow N+k-1$ **to** $N+K-1$ **do**
    $\hat{u}_t^{j} \leftarrow C^*(q(\hat{x}_t^{j}))$;
    $\hat{x}_t^{j+1} \leftarrow f(\hat{x}_t^{j}, \hat{u}_t^{j})$;
else
  // if $k = 1$ it means that $\hat{x}_t^N \in \mathbb{T}$
  $\bar{i} \leftarrow min\{i|\hat{x}_t^N \in \mathsf{CPre}_i(\mathbb{T})\}$;
  for $i \leftarrow 1$ **to** $\bar{i}$ **do**
    $\hat{u}_t^{N+i-1} \leftarrow C_{\bar{i}-i+1}(q(\hat{x}_t^{N+i-1}))$;
    $\hat{x}_t^{N+i} \leftarrow f(\hat{x}_t^{N+i-1}, \hat{u}_t^{N+i-1})$;
  for $j \leftarrow N+\bar{i}$ **to** $N+K-1$ **do**
    $\hat{u}_t^{j} \leftarrow C^*(q(\hat{x}_t^{j}))$;
    $\hat{x}_t^{j+1} \leftarrow f(\hat{x}_t^{j}, \hat{u}_t^{j})$;

5. Numerical example

Consider a mobile cart model as a numerical example, similar to [29]. The state vector $x \in \mathbb{R}^3$ and the control input vector $u \in \mathbb{R}^2$. The system dynamics are given as:

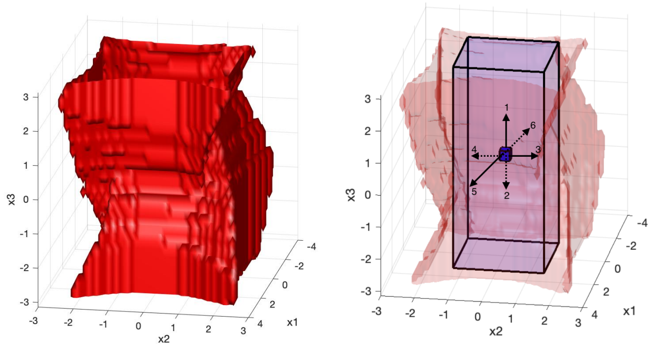
$$\begin{aligned} x_1(t+1) &= x_1(t) + u_1(t)\cos(x_3(t)) \\ x_2(t+1) &= x_2(t) + u_1(t)\sin(x_3(t)) \\ x_3(t+1) &= x_3(t) + u_2(t)\ (\mathrm{mod} 2\pi) \end{aligned} \tag{11}$$

The model describes the behavior of a mobile cart with (x_1, x_2) representing the 2D Cartesian coordinates in meters. x_3 represents the angular orientation of the velocity vector with respect to the x_1 axis in radians, u_1 is the linear velocity in m/s, and u_2 is the angular velocity in rad/s. Note that by convention, we consider the angle $x_3 \in [-\pi, \pi)$. The system is subjected to state and control input constraints:

$$\mathbb{X} = \left\{ (x_1, x_2)^T \in \mathbb{R}^2 \,\middle|\, \begin{array}{l} x_1^2 - x_2^2 \leq 4 \\ 4x_2^2 - x_1^2 \leq 16 \end{array} \right\} \tag{12}$$

$$\mathbb{U} = [0.2, 2] \times [-1, 1]. \tag{13}$$

Figure 2a illustrates the approximation of the maximal control invariant set computed using symbolic techniques. Its construction procedure used a symbolic model with 109,200 symbolic states and 40 symbolic inputs. The computation time is about 2 minutes with a PC 1.4 GHz Intel Core i5. This control invariant set has a relatively complex shape and is obviously not suitable for direct use as terminal constraint in nonlinear MPC



(a) (b)

Figure 2: On the left is the approximation of the maximal control invariant set computed using symbolic control, on the right is the stretching method, the direction of stretching is in order 1-6.

design according to the framework (2). Aside the fact that the control invariant set is highly nonlinear and thus impacts the real-time optimization performances, it should be noted that the symbolic procedure represents it in terms of a large union of convex sets.

Figure 2b shows that a large hyperrectangle can be found inside the maximum control invariant set. Its practical construction follow an intuitive procedure, starting with a small rectangle and stretching it in different directions. We use $[x_l, x_u]$, the coordinates of the minimal and maximal vertices to represent this hyperrectangle. In this 3 dimensional case $x_l = [-1.3; -1.3; -\pi]$, $x_u = [1.3; 1.4; \pi]$.

Once we have identified this large hyperrectangle candidate for K-step control invariance, we can proceed to find its corresponding index K. In Figure 3, we show that $\mathbb{T} \subseteq \mathsf{CPre}_{[1:K]}(\mathbb{T})$ is obtained when $K = 5, 6$ and 7. When $K = 7$, $\mathsf{CPre}_{[1:7]}(\mathbb{T})$ is identical to the control invariant set shown in Figure 2a. This numerical certification of K-step control invariance was obtained in 14s on the same PC, an attractive off-line computation time given the complexity of the maximal control invariant set. We will use the minimum value of K by default in the following.

For the optimization problem (5), the prediction horizon was chosen to be $N = 20$, the penalty terms for position and control input were set to 100 and 1, respectively. $P\mathbf{x}_{\mathbf{t}}^{[0,\mathbf{N}]}$ is the projection of $\mathbf{x}_{\mathbf{t}}^{[0,\mathbf{N}]}$ on the first two coordinates (x_1, x_2), $x_{ref} \in \mathbb{R}^2$ is the reference point. With the previously computed hyperrectangle $[x_l, x_u]$, we have the following optimization problem :

$$
\begin{array}{ll}
 & \min_{\mathbf{u}_t^{[0,N+K-1]}} 100\|P\mathbf{x}_t^{[0,N]} - x_{ref}\|^2 + \|\mathbf{u}_t^{[0,N-1]}\|^2 \\
 & x_t^0 = x, \\
 & x_t^{i+1} = f(x_t^i, u_t^i), \quad i = 0, \ldots, N+K-1 \\
\text{s.t.} & u_t^i \in \mathbb{U}, \quad i = 0, \ldots, N+K-1 \\
 & x_t^i \in \mathbb{X}, \quad i = 0, \ldots, N+K \\
 & \exists k \in \{1, \ldots, K\}, \quad x_t^{N+k} \in [x_l, x_u].
\end{array}
\tag{14}
$$

Figure 4 presents a comparison between NMPC without terminal constraints and NMPC based on K-step control invariant sets.

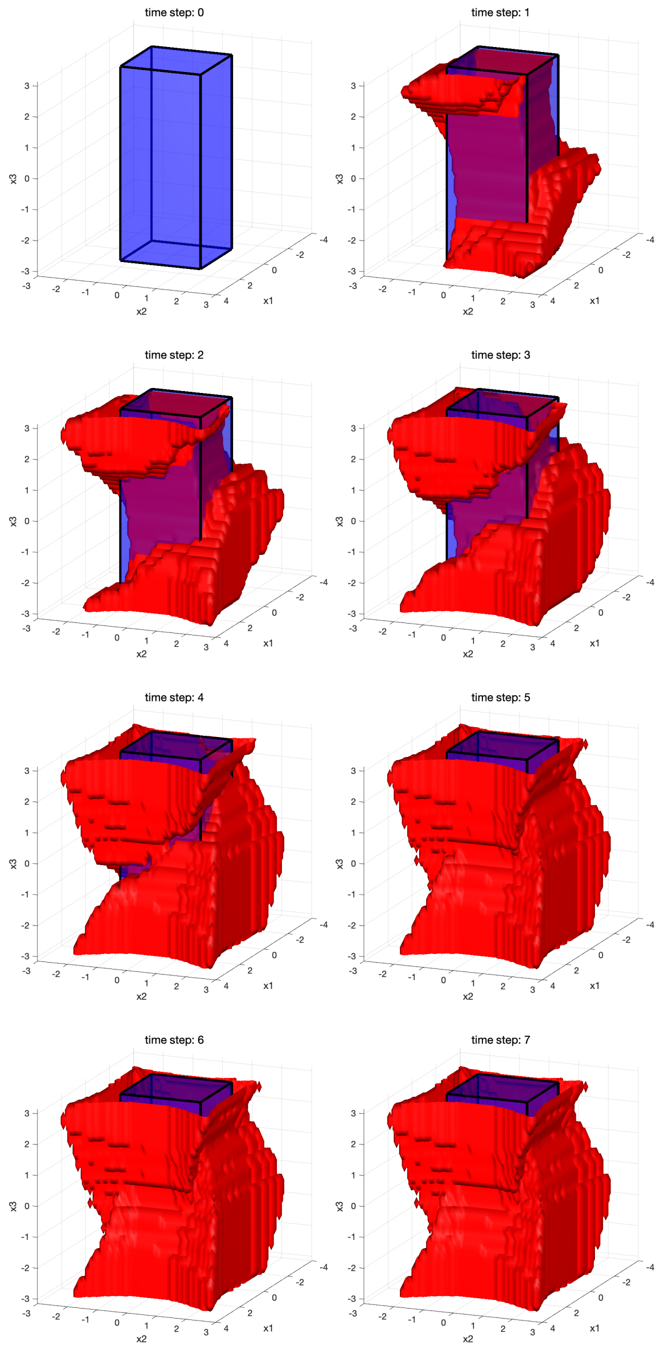
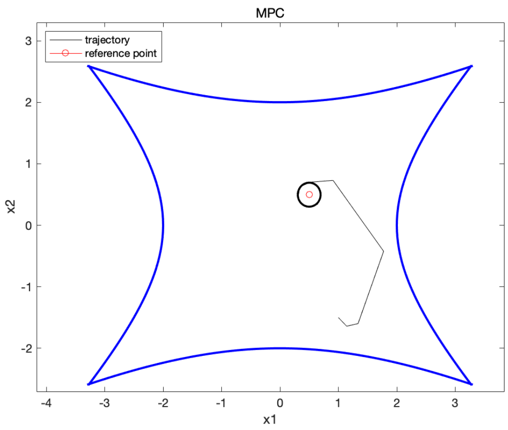
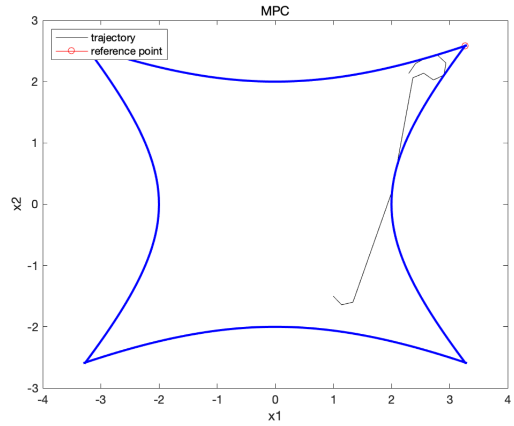


Figure 3: The evolution of the set $\mathsf{CPre}_{[1:K]}$ (red parts) with K is depicted in the figure, where $K = 0, 1, 2, ...7$. By the fifth iteration step, the hyperrectangle is completely contained in the computed states, and by the seventh iteration step, the computed states are identical to the maximum control invariant set.

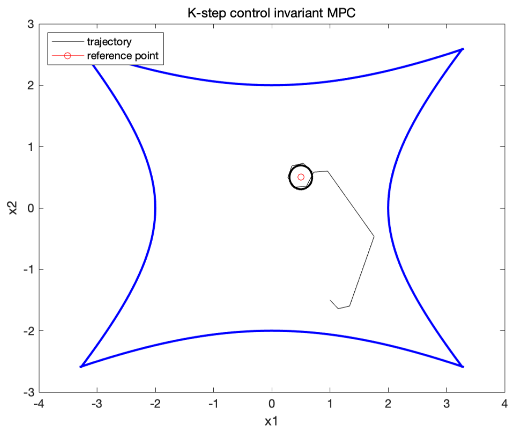
A reference point for the position $x_{ref} = (0.5, 0.5)$ is considered in a first case, the third component (orientation) not being penalized in the cost index. One can observe that the two methods exhibit equivalent performance. It should be noted that the optimal trajectory after transitory should be a circle around the reference point (due to speed constraints $u_1 \geq 0.2m/s$). The closed loop may however not reach such a behaviour when the receding optimization problem is initialized with an arbitrary control sequence, due to convergence to a local minimum.



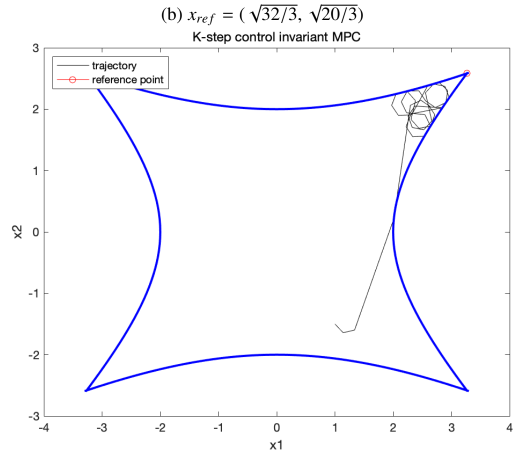
(a) $x_{ref} = (0.5, 0.5)$



(b) $x_{ref} = (\sqrt{32/3}, \sqrt{20/3})$



(c) $x_{ref} = (0.5, 0.5)$



(d) $x_{ref} = (\sqrt{32/3}, \sqrt{20/3})$

Figure 4: Trajectory of moving carts. The upper two figures show the NMPC approach without terminal constraints and the two figures below show the NMPC using a K-step control invariant set (K=5) as terminal constraint.

Using the symbolic control-based initialization method described in the previous section, the trajectories in Figure 4(c) are obtained with same performances of a costly NMPC formulation (2), see trajectories in Figure 4(a).

For the reference $x_{ref} = (\sqrt{32/3}, \sqrt{32/3})$, the NMPC without terminal constraints fails in the tracking process of target point which is located at the corner point, see Figure 4(b). Indeed, the optimization problem became infeasible at time instant $t = 13$. In comparison, the NMPC strategy based on the K-step control invariant set enables the cart to move continuously without encountering safety issues, thus confirming the recursive feasibility properties.

The time evolution of the state and inputs signals for the position trajectories depicted in Figure 4(d) are represented in Figure 5.

6. Conclusion

Ensuring the recursive feasibility of MPC is a widely discussed challenge and this paper offers a novel perspective to address this issue. By further developing computational methods based on symbolic control, we effectively compute the K-step control invariant sets and utilize them as constraints within the NMPC framework. This approach, which is straightforward and easy to implement, was shown to achieve promising results. The approach can be extended to solve path planning problems

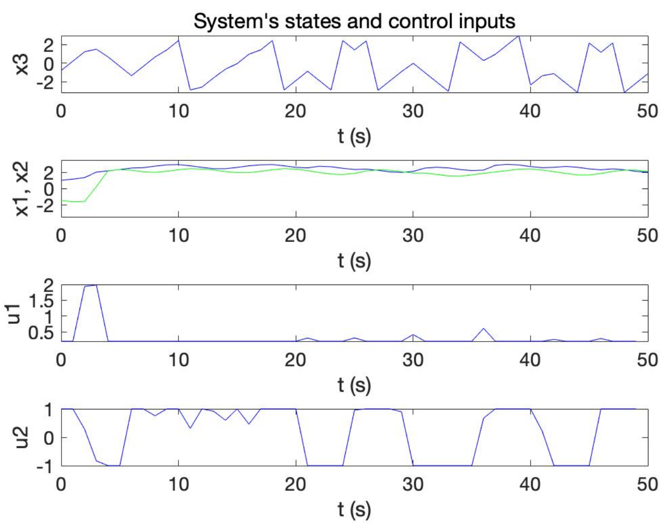


Figure 5: Evolution of system's states and control inputs for (d) in Figure 4

in complex environments. From the theoretical point of view, the robust $K-$step invariance and its used in MPC deserves further attention.

References

[1] E. F. Camacho, C. Bordons, Model predictive control (advanced textbooks in control and signal processing), Springer, 1999.

[2] G. Goodwin, M. M. Seron, J. A. De Doná, Constrained control and estimation: an optimisation approach, Springer Science & Business Media, 2006.

[3] J. M. Maciejowski, Predictive Control: With Constraints, Pearson Education, 2002.

[4] L. Grüne, J. Pannek, Nonlinear model predictive control, Springer, 2017.

[5] F. Blanchini, Set invariance in control, Automatica 35 (11) (1999) 1747–1767.

[6] D. Q. Mayne, J. B. Rawlings, C. V. Rao, P. O. Scokaert, Constrained model predictive control: Stability and optimality, Automatica 36 (6) (2000) 789–814.

[7] M. Oishi, I. Mitchell, C. Tomlin, P. Saint-Pierre, Computing viable sets and reachable sets to design feedback linearizing control laws under saturation, in: Proceedings of the 45th IEEE Conference on Decision and Control, IEEE, 2006, pp. 3801–3807.

[8] J. M. Bravo, D. Limón, T. Alamo, E. F. Camacho, On the computation of invariant sets for constrained nonlinear systems: An interval arithmetic approach, Automatica 41 (9) (2005) 1583–1589.

[9] B. Decardi-Nelson, J. Liu, Computing robust control invariant sets of constrained nonlinear systems: A graph algorithm approach, Computers & Chemical Engineering 145 (2021) 107177.

[10] M. Korda, D. Henrion, C. N. Jones, Convex computation of the maximum controlled invariant set for discrete-time polynomial control systems, in: 52nd IEEE Conference on Decision and Control, IEEE, 2013, pp. 7107–7112.

[11] M. Fiacchini, T. Alamo, E. F. Camacho, On the computation of convex robust control invariant sets for nonlinear systems, Automatica 46 (8) (2010) 1334–1338.

[12] M. A. Ben Sassi, A. Girard, Controller synthesis for robust invariance of polynomial dynamical systems using linear programming, Systems & control letters 61 (4) (2012) 506–512.

[13] L. Schäfer, F. Gruber, M. Althoff, Scalable computation of robust control invariant sets of nonlinear systems, IEEE Transactions on Automatic Control (2023).

[14] F. Blanchini, W. Ukovich, Linear programming approach to the control of discrete-time periodic systems with uncertain inputs, Journal of Optimization Theory and Applications 78 (1993) 523–539.

[15] Y. I. Lee, M. Cannon, B. Kouvaritakis, Extended invariance and its use in model predictive control, Automatica 41 (12) (2005) 2163–2169.

[16] Y. I. Lee, B. Kouvaritakis, Constrained robust model predictive control based on periodic invariance, Automatica 42 (12) (2006) 2175–2181.

[17] C. Böhm, M. Lazar, F. Allgöwer, Stability of periodically time-varying systems: Periodic Lyapunov functions, Automatica 48 (10) (2012) 2663–2669.

[18] W. Lombardi, S. Olaru, G. Bitsoris, S.-I. Niculescu, Cyclic invariance for discrete time-delay systems, Automatica 48 (10) (2012) 2730–2733.

[19] S. V. Raković, B. Kern, R. Findeisen, Practical set invariance for decentralized discrete time systems, in: 49th IEEE Conference on Decision and Control (CDC), IEEE, 2010, pp. 3283–3288.

[20] R. Comelli, S. Olaru, M. Seron, E. Kofman, Application of a stabilizing model predictive controller to path following for a car-like agricultural robot, Optimal Control Applications and Methods (Mar. 2024). doi:10.1002/oca.3126.

[21] R. Comelli, S. Olaru, E. Kofman, Inner–outer approximation of robust control invariant sets, Automatica 159 (2024) 111350.

[22] S. Olaru, M. Soyer, Z. Zhao, C. E. Dórea, E. Kofman, A. Girard, From relaxed constraint satisfaction to $p-$ invariance of sets, IEEE Transactions on Automatic Control (2024).

[23] N. Athanasopoulos, A. I. Doban, M. Lazar, On constrained stabilization of discrete-time linear systems, in: 21st Mediterranean Conference on Control and Automation, IEEE, 2013, pp. 830–839.

[24] P. Tabuada, Verification and control of hybrid systems: a symbolic approach, Springer Science & Business Media, 2009.

[25] G. Reissig, A. Weber, M. Rungger, Feedback refinement relations for the synthesis of symbolic controllers, IEEE Transactions on Automatic Control 62 (4) (2016) 1781–1796.

[26] A. Girard, A. Eqtami, Least-violating symbolic controller synthesis for safety, reachability and attractivity specifications, Automatica 127 (2021) 109543.

[27] E. C. Kerrigan, Robust constraint satisfaction: Invariant sets and predictive control, Ph.D. thesis, University of Cambridge UK (2001).

[28] P.-J. Meyer, A. Devonport, M. Arcak, Interval reachability analysis: Bounding trajectories of uncertain systems with boxes for control and verification, Springer Nature, 2021.

[29] Z. Azaki, A. Girard, S. Olaru, Predictive and symbolic control: Performance and safety for non-linear systems, IFAC-PapersOnLine 55 (16) (2022) 290–295.