STABILITY & FEASIBILITY OF CONSTRAINED RECEDING HORIZON CONTROL

Pascal Grieder*†, Marco Lüthi*, Pablo A. Parrilo* and Manfred Morari*

* Institut für Automatik, ETH - Swiss Federal Institute of Technology, Physikstrasse 3, CH-8092 Zürich
† Corresponding Author: E-mail: grieder@control.ee.ethz.ch, Tel. +41 01 632 7313

Keywords: Optimal Control, Receding Horizon Control, Model Predictive Control, Constrained Control

Abstract

In this paper we present an approach to test stability and feasibility of discrete-time LTI systems subject to constrained optimal control. The analysis is based on the feedback solution of the constrained optimal control problem formulated in a Receding Horizon fashion. The validation procedure consists of two independent parts. For stability, a piecewise quadratic Lyapunov function is computed by solving LMIs. Sufficient conditions for the existence of such functions are given. To guarantee feasibility, an efficient algorithm for computing invariant subsets of piecewise affine systems is presented. Finally, we demonstrate how this method may serve to obtain controllers of low complexity.

1 Introduction

The main focus of this paper will be on receding horizon control (RHC) of constrained linear systems. In RHC, at each time step an optimal control problem is solved over a finite horizon, but only the first element of the computed control sequence is applied. Traditionally, this optimization was carried out on line and the resulting method has become known as Model Predictive Control (MPC). More recently, it has been shown [2] how to compute explicitly the piecewise affine (PWA) state feedback control law corresponding to the optimal controller by solving a *multi-parametric Quadratic Program (mp-QP)*. The analysis techniques proposed in this paper require the explicit state feedback control law, but after successful analysis, the controller can be implemented in the equivalent MPC form if this is preferred. The main problem of RHC is that it does not, in general, guarantee stability. Furthermore, RHC might drive the state to a part of the state space where no solution to the finite time optimal control problem satisfying the constraints exists. This existence of an input sequence satisfying the constraints is referred to as feasibility. The lack of feasibility and stability guarantees has been addressed by imposing terminal set constraints [9, 3] or by solving the infinite-horizon problem [5]. The constraints on the terminal set necessary to show stability tend to result in small terminal sets. This inadvertently leads to large horizons if the distance between the initial state and the terminal set is large and the input is bounded. Large horizons inherently result in significant computational complexity.

The method proposed in this paper consists of two algorithmic building blocks, which together allow for a proof of stability and feasibility for finite-time optimal control applied in a RHC manner. A simple controller is computed and analyzed for stability and feasibility. If no properties can be derived, controller complexity is increased until stability and feasibility can be guaranteed. The stability analysis is based on the results in [7, 4] and the invariant set computation is based on procedures in [8]. As will be shown in Section 5, the proposed analysis scheme may be used to obtain controllers of significantly lower complexity than those obtained by other approaches in the literature [9, 3].

2 Problem Formulation

We will consider optimal control problems for discrete-time linear time-invariant systems

$$x(t+1) = Ax(t) + Bu(t), \tag{1}$$

with $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$. Let $x(t)$ denote the measured state at time t and $x_{t+k|t}$ denote the predicted state at time $t+k$ given the state at time t. For brevity we will denote $x_{k|0}$ as x_k.

Assume now that the states and the inputs of system (1) are subject to the following constraints

$$x \in \mathbb{X} \subseteq \mathbb{R}^n, \qquad u \in \mathbb{U} \subseteq \mathbb{R}^m, \tag{2}$$

where $\mathbb{X}$ and $\mathbb{U}$ are compact polytopic sets containing the origin in their interior, and consider the constrained finite-time optimal control problem

$$\begin{aligned} J_N^*(x(0)) &= \min_{u_0,\dots,u_{N-1}} \sum_{k=0}^{N-1} \left(u_k'\mathcal{R}u_k + x_k'\mathcal{Q}x_k\right) \quad &(3a)\\ &\quad +x_N'\mathcal{Q}_f x_N \\ \text{subj. to} \quad & x_k \in \mathbb{X} \quad \forall k \in \{1,\dots,N\}, &(3b)\\ & u_k \in \mathbb{U} \quad \forall k \in \{0,\dots,N-1\}, &(3c)\\ & x_{k+1} = Ax_k + Bu_k, \quad x_0 = x(0), &(3d)\\ & \mathcal{Q} \succeq 0,\ \mathcal{Q}_f \succeq 0,\ \mathcal{R} \succ 0. &(3e) \end{aligned}$$

The solution $U_N^*(x(0)) = [u_0,\dots,u_{N-1}]$ to problem (3) is a function of the initial condition $x(0)$. Before going further, we will introduce the following definitions:

Definition 1 *We define the N-step feasible set $\mathcal{X}_f^N \subseteq \mathbb{R}^n$ as the set of initial states $x(0)$ for which the optimal control problem (3) is feasible, i.e.*

$$\mathcal{X}_f^N = \{x(0) \in \mathbb{R}^n | \exists U_N \in \mathbb{R}^{Nm},\ GU_N \le W + Ex(0)\},$$

where m denotes the number of inputs and N the prediction horizon.

For a specific $x(0)$ problem (3) is a quadratic program. As shown in [2] problem (3) can be solved for all $x(0)$ within a polyhedral set of values by considering (3) as a mp-QP.

Theorem 1 *[2, 3] Consider the constrained finite time optimal control problem (3). Then, the set of feasible parameters $\mathcal{X}_f^N$ is polyhedral, the optimizer $U_N^* : \mathcal{X}_f^N \mapsto \mathbb{R}^s$ is continuous and piecewise affine (PWA), i.e.*

$$U_N^*(x(0)) = \tilde{F}_r x(0) + \tilde{G}_r \quad \text{if} \quad x(0) \in \mathcal{P}_r \tag{4a}$$

$$\mathcal{P}_r = \{x \in \mathbb{R}^n | H_r x \leq K_r\}, \tag{4b}$$

and the optimal solution $J_N^* : \mathcal{X}_f^N \mapsto \mathbb{R}$ is continuous, convex and piecewise quadratic (PWQ).

Note that the evaluation of the PWA solution (4) of the mp-QP provides the same result as solving the quadratic program, i.e., when $x(0)$ is known, the optimal input sequence $U_N^*(x(0))$ in (4) is identical to the optimal input sequence obtained by solving the quadratic program (3) for $x(0)$.

Henceforth, we will denote the RHC feedback law which provides the first input as $u_0 = F_r x + G_r$. Analysis of the mp-QP solution [2] yields that $\{\mathcal{P}_r\}_{r=1}^R$ is a polyhedral partition of $\mathcal{X}_f^N$. We will denote the polyhedron $\mathcal{P}_r$ as region r.

In RHC the constraints are only enforced for N steps ahead, which can lead to infeasibility of (3). Even in case of no model mismatch, the optimal open-loop trajectory is different from the trajectory which results from the closed-loop control scheme [9, 3]. This may therefore affect not only feasibility but stability as well. As stated in the introduction, this problem is commonly dealt with by enforcing terminal set constraints [9, 3] which tend to result in large optimization problems which are unsuitable for fast systems. In the following an analysis scheme is presented which allows for simple controllers with stability and feasibility guarantees. The scheme consists of computing simple controllers with short prediction horizons and subsequently analyzing stability and feasibility of the closed-loop system.

3 Stability and Feasibility of RHC

In order for RHC to be applicable we must ascertain that the controlled system is stable and that the system constraints will always be satisfied, i.e., stability and feasibility must be proven. The following definition and lemma will address the issue of feasibility:

Definition 2 *We define the control invariant subset $\mathcal{X}_I$ of the feasible set $\mathcal{X}_f^N$ computed in (3) ($\mathcal{X}_I \subseteq \mathcal{X}_f^N$) as:*

$$\mathcal{X}_I = \{x(0) \in \mathcal{X}_f^N | x(k) \in \mathcal{X}_f^N \ \ \forall k \geq 0\}.$$

The controller is given in Theorem 1 and the associated closed-loop dynamics are given by

$$x(k+1) = (A + BF_r)x(k) + BG_r \ \text{ if } \ H_r x(k) \leq K_r. \tag{5}$$

Lemma 1 *For a given feasible compact set $\mathcal{X}_f^N$ as presented in Theorem 1, a control invariant subset $\mathcal{X}_I \subseteq \mathcal{X}_f^N$ induces feasibility of (3) for all time if $x(0) \in \mathcal{X}_I$ and the controller given in Theorem 1 is applied in a receding horizon manner.*

From Lemma 1 we can conclude that a sufficient condition for infinite horizon feasibility is positive invariance of the regions obtained with the methods in [2]. Note that positive invariance does not induce stability since limit cycles cannot be ruled out. Therefore we require a proof of stability as well. These two aspects will be dealt with separately in the following two subsections.

3.1 Feasibility

As previously stated, a stabilizing control law does not guarantee feasibility for all time. According to Lemma 1 invariance implies feasibility. We will therefore present an efficient method for computing the control invariant subset $\mathcal{X}_I$ as well as a method for checking whether $\mathcal{X}_I$ is equal to the maximum control invariant set $\mathcal{C}_\infty$. For a more elaborate overview of invariant set computation for PWA systems we refer the reader to [8].

3.1.1 Invariant Set Computation

The following procedure allows the exact computation of the control invariant set $\mathcal{X}_I \subseteq \mathcal{X}_f^N$ of a given controller partition. It should be noted, that in general $\mathcal{X}_I$ is non-convex and does not even need to be connected. The underlying principle of the algorithm is to iteratively remove states from which the trajectory exits the feasible set $\mathcal{X}_f^N$ until no such states can be found.

To reduce the computational effort to a minimum, two data structures are introduced so that redundant computation is avoided. The transition matrix $\mathcal{T} \in \{0,1\}^{R \times R}$ (where R denotes the number of regions) and the modification vector $\mathcal{M} \in \{0,1\}^R$ store the feasible transitions and keep track of set modifications respectively. The exact definitions are

$$\mathcal{T}(s,t) = 1, \text{ if } \exists x(k) \in \mathcal{P}_s, \text{ such that } x(k+1) \in \mathcal{P}_t,$$
$$\text{else } \ \mathcal{T}(s,t) = 0;$$
$$\mathcal{M}(r) = 1, \text{ if } \mathcal{P}_r^{i+1} \neq \mathcal{P}_r^i, \ \text{ else } \ \mathcal{M}(r) = 0.$$

Here, the dynamics are given by (5) and i denotes the iteration number in the algorithm given below:

1. Given a controller partition $\{\mathcal{P}_r\}_{r=1}^R$, compute the entries of the transition matrix $\mathcal{T}$. Subsequently create the transition set $\Pi = \{s, t \in \{1, \ldots, R\} \mid \mathcal{T}(s,t) = 1\}$, initialize $\mathcal{M}(r) = 1, \forall r \in \{1, \ldots, R\}$ and set the iteration counter $i = 0$.

2. For all $s, t \in \Pi$, do:
If $\mathcal{M}(t) = 1$ compute and store the subset $\mathcal{S}_{s,t}^{i+1} = \{x(k) \in \mathbb{R}^n | x(k) \in \mathcal{P}_s^i,\ x(k+1) \in \mathcal{P}_t^i\}$ for the dynamics in (5). If $M(t) = 0$, set $\mathcal{S}_{s,t}^{i+1} = \mathcal{S}_{s,t}^i$.

3. For each start region $\mathcal{P}_s^i$, attempt to create the union $\mathcal{P}_s^{i+1} = \bigcup_{(s,t)\in\Pi} S_{s,t}^{i+1}$. $\mathcal{P}_s^{i+1} \subseteq \mathcal{P}_s^i$ corresponds to the set of points $x \in \mathcal{P}_s^i$ which remains within the partition $\bigcup_{r=\{1,\dots,R\}} \mathcal{P}_r$ in one time step. The algorithm for computing the union of polytopes is described in [1].

   (a) The union $\mathcal{P}_s^{i+1}$ is convex:
   Set $\mathcal{M}(s) = 0$ if $\mathcal{P}_s^{i+1} \equiv \mathcal{P}_s^i$ and set $\mathcal{M}(s) = 1$ if $\mathcal{P}_s^{i+1} \subset \mathcal{P}_s^i$.

   (b) The union $\mathcal{P}_s^{i+1}$ is non-convex:
   Add new regions to the controller partition, i.e. for all $S_{s,t}^{i+1} \neq \varnothing$, set $\mathcal{P}_{R+1}^{i+1} = S_{s,t}^{i+1}$ and $R = R + 1$. In addition, the transition matrices $\mathcal{T}$, Π and the modification vector $\mathcal{M}$ are updated accordingly.

4. If no region has been modified ($\mathcal{M}(r) = 0\ \forall r \in \{1, \dots, R\}$), the algorithm has converged and the invariant subset is found. Otherwise set $i = i + 1$ and goto step 2.

The union of all remaining regions is equal to $\mathcal{X}_I$.

Lemma 2 *Given a compact initial set $\mathcal{X}_f^N$ over which a PWA feedback law is defined, the presented polyhedral-invariant-set algorithm always converges.*

Proof It follows from (2) that the initial volume of $\mathcal{X}_f^N$ is finite. At each iteration, this volume is reduced by $\Delta Vol > 0$ until $\Delta Vol = 0$, i.e., the volume is strictly decreasing. If any point contained in the resulting invariant set does not belong to $\mathcal{X}_I$, ΔVol would not be zero which implies that the resulting set is positive invariant. Therefore the algorithm always converges, though not necessarily in finite time. □

It should be noted that in extensive simulations the algorithm always converged in finite time.

3.1.2 Computing the Maximum Control Invariant Set $\mathcal{C}_\infty$

The following definition is derived from [8]:

Definition 3 *The set $\mathcal{C}_\infty$ is the maximal control invariant set contained in $\mathbb{R}^n$ for the system $x_{k+1} = f(x_k, u_k)$ subject to the constraints in (2) if and only if $\mathcal{C}_\infty$ is control invariant and contains all the control invariant sets contained in $\mathbb{R}^n$ which uphold the constraints in (2).*

An important question which arises in practical problems, is whether the obtained invariant set $\mathcal{X}_I$ is the maximum control invariant set $\mathcal{C}_\infty$ for a given problem, i.e., does there exist an alternative feedback law which produces a larger invariant set $\mathcal{X}_I' \supset \mathcal{X}_I$. As was stated in [5], the set $\mathcal{C}_\infty$ is convex and polyhedral, independent of the weight matrices $(\mathcal{Q}, \mathcal{R})$ and solely depends on the system dynamics and constraints. We can therefore discard all invariant sets $\mathcal{X}_I$ which are not convex. Convexity can be checked with the methods in [1].

If $\mathcal{X}_I$ is convex, there is an easy way to check whether the invariant set $\mathcal{X}_I$ is also the maximum control invariant set $\mathcal{C}_\infty$. Solving the following problem as an mp-QP

$$J_{N=1}^*(x(0)) = \min_{u_0} u_0' u_0, \quad (6a)$$

$$\text{subj. to} \quad x_1 \in \mathbb{X},\ x_1 \in \mathcal{X}_I \text{ and } u_0 \in \mathbb{U}, \quad (6b)$$

$$x_1 = Ax_0 + Bu_0,\ x_0 = x(0), \quad (6c)$$

yields a convex partition $\mathbb{X}_{\text{mpQP}}$. If $\mathcal{X}_I \equiv \mathbb{X}_{\text{mpQP}}$, then $\mathcal{X}_I$ is $\mathcal{C}_\infty$. The proof follows from the fact that all states which can enter the set $\mathcal{X}_I$ in one step are contained in $\mathcal{X}_I$, i.e., $Pre(\mathcal{X}_I) = \mathcal{X}_I$ [8]. Note that this computation is very efficient since the mp-QP is solved over a prediction horizon $N = 1$ only.

3.2 Stability

In order to guarantee stability, we identify a PWQ Lyapunov function over the invariant set $\mathcal{X}_I$ which was computed with the methods in Section 3.1.1. The procedure is identical to the LMI procedure presented in [4] but will be restated here for completeness. Furthermore, we will show that the considered Lyapunov function always exists for sufficiently large prediction horizons N in (3).

First, certain solution properties need to be established:

Lemma 3 *[11, 5] Given system (1) with (A, B) stabilizable and (C, A) detectable ($C'C = Q$), there exists a finite horizon $\bar{N}_\mathcal{S}$ defined over a compact set $\mathcal{S}$ of initial states such that the receding horizon solution obtained in (3) for horizon $\bar{N}_\mathcal{S}$ and terminal weight $Q_f = P_{LQR}$ equals the open loop solution for $N \to \infty$. The terminal weight P_{LQR} is obtained from the Algebraic Riccati Equation (ARE). The equality also holds for all horizons $N \geq \bar{N}_\mathcal{S}$. The procedure of computing $\bar{N}_\mathcal{S}$ is explained in [5].*

Lemma 4 *[5] Given a horizon $N \geq \bar{N}_\mathcal{S}$ and terminal weight $Q_f = P_{\text{LQR}}$ in (3), the PWA partition obtained by solving an mp-QP is convex and positive invariant. The terminal weight P_{LQR} can be computed by the Algebraic Riccati Equation (ARE).*

Theorem 2 *The PWQ value function J_N^* in (3) is a Lyapunov function for all $N \geq \bar{N}_\mathcal{S}$, if $Q_f = P_{\text{LQR}}$. The function $J_N^*(x)$ is piecewise-quadratic if $x \in \mathcal{X}_f^N$ and quadratic if $x \in \mathcal{P}_0$, where $\mathcal{P}_0$ is the region containing the origin.*

Proof Follows from Lemma 4 and Theorem 1. Since the solution obtained for a horizon $N \geq \bar{N}_\mathcal{S}$ is infinite horizon optimal, the solution is stabilizing, infinite time feasible and the cost $J_N^*(x_k)$ is strictly decreasing from k to $k+1$. $J_N^*(x)$ is quadratic around the origin since there the optimal feedback law corresponds to the Riccati LQR and therefore $J_N^*(x) = x' P_{LQR} x$ in a neighborhood of the origin. It follows that the value function is only zero at the origin ($J_N^*(0) = 0$). Since $J_N^*(x)$ is quadratic around the origin and the set $\mathcal{S}$ is compact, a quadratic upper and lower bound on $J_N^*(x)$ exists. The PWQ cost function J_N^* in (3) is therefore a Lyapunov function guaranteeing exponential stability. Since the set $\mathcal{X}_f^N$ is also positive invariant if computed with horizon $N \geq \bar{N}_{\mathcal{S}}$ [5], $J_N^*(x_k)$ is defined for all $k \geq 0$, iff $x_0 \in \mathcal{X}_f^N$. □

In the following, we will consider a PWQ Lyapunov function of the form:

$$P_{\mathrm{PWQ}}(x) = x' Q_r x + x' 2 L_r + C_r, \tag{7a}$$

$$\text{if} \quad H_r x \leq K_r. \tag{7b}$$

As shown in [4], this function does not need to be continuous to imply stability for discrete time systems. If a state moves from region s ($x_0 \in \mathcal{P}_s$) to region t ($x_1 \in \mathcal{P}_t$) in one time step, the decay rate of the Lyapunov function ΔV_{st} is defined as:

$$\begin{aligned} \Delta V_{st}(x_0) &= P_{\mathrm{PWQ}}(x_1) - P_{\mathrm{PWQ}}(x_0), \\ &= x_0' \, \Delta Q_{st} \, x_0 + 2 x_0' \, \Delta L_{st} + \Delta C_{st}. \end{aligned}$$

We will define here the set of states contained in region s that enter region t in one time step.

$$\mathcal{T}_{st} = \{x_0 \in \mathbb{R}^n | H_s x_0 \leq K_s, H_t((A+BF_s)x_0 + BG_s) \leq K_t\} \tag{8}$$

The sets $\mathcal{T}_{st}$ can be efficiently computed by applying the methods presented in Section 3.1.1 and will be described by $\mathcal{T}_{st} = \{x \in \mathbb{R}^n | \; H_{st} x \leq K_{st}\}$. The set of all feasible transitions st is stored in the set denoted as $\Pi = \{st \in \mathbb{N} \mid \mathcal{T}_{st} \neq \varnothing\}$. Concisely, we aim to prove stability by identifying a Lyapunov function $P_{\mathrm{PWQ}}(x)$ defined over the control invariant set $\mathcal{X}_I$ which satisfies the following conditions:

$$\text{find} \quad P_{\mathrm{PWQ}} \;\; \text{s.t.} \tag{9a}$$

$$P_{\mathrm{PWQ}}(x) \geq \epsilon ||x||^2, \; \forall x \in \mathcal{X}_I, \; \epsilon > 0 \tag{9b}$$

$$\Delta V_{st}(x) \leq -\rho ||x||^2, \; \forall x \in \mathcal{X}_I, \; \rho > 0. \tag{9c}$$

In order to find a function $PWQ(x)$ which satisfies (9c) we can formulate the following inequality with $\bar{x} = [x_0 \;\; 1]'$ and $\rho > 0$ for region s:

$$\Delta V_{st}(x_0) = PWQ(x_1) - PWQ(x_0) \tag{10a}$$

$$= \bar{x}' \begin{bmatrix} \Delta Q_{st} & \Delta L_{st}, \\ \Delta L_{st}' & \Delta C_{st} \end{bmatrix} \bar{x} \tag{10b}$$

$$\leq \bar{x}' \Bigg(- \begin{bmatrix} -H_{st}' \\ K_{st}' \end{bmatrix} N_{st} [-H_{st} \;\; K_{st}] - \rho \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \Bigg) \bar{x} \tag{10c}$$

$$\leq -\rho x_0' x_0. \tag{10d}$$

Note that the term in (10c) is smaller than the term in (10d), if the state x_0 is inside the set $\mathcal{T}_{st} = \{x \in \mathbb{R}^n | \; H_{st} x \leq K_{st}\}$, since N_{st} is a matrix consisting of nonnegative elements only. The formulation corresponds to the S-procedure also applied in [7, 4].

With (10) equation (9) can easily be reformulated to obtain a suitable LMI problem:

$$\text{find} \;\; P_{\mathrm{PWQ}}, \; N_r, \; N_{st}, \; \rho, \; \epsilon, \; \text{s.t.}$$

$$\forall r \in \{1, \ldots, R\} \text{ and } \forall \; st \in \Pi \tag{11a}$$

$$\begin{bmatrix} -\Delta Q_{st} - \rho I & -\Delta L_{st} \\ -\Delta L_{st}' & -\Delta C_{st} \end{bmatrix} \succeq \begin{bmatrix} -H_{st}' \\ K_{st}' \end{bmatrix} N_{st} \, [-H_{st} \;\; K_{st}], \tag{11b}$$

$$\begin{bmatrix} Q_r - \epsilon I & L_r \\ L_r' & C_r \end{bmatrix} \succeq \begin{bmatrix} -H_r' \\ K_r' \end{bmatrix} N_r \, [-H_r \;\; K_r], \tag{11c}$$

$$N_{st} \geq 0, \;\; N_r \geq 0, \;\; \rho > 0, \;\; \epsilon > 0, \tag{11d}$$

$$N_r = N_r', N_r \in \mathbb{R}^{d_r \times d_r}, N_{st} = N_{st}', N_{st} \in \mathbb{R}^{d_{st} \times d_{st}}$$

$$C_1 = 0, \;\; L_1 = 0 \in \mathbb{R}^n. \tag{11e}$$

Here (11b) induces $\Delta V_{st} \leq 0$, (11c) ascertains that the PWQ Lyapunov function is positive and (11d) ensures that all elements of N_r and N_{st} are nonnegative while d_r and d_{st} denote the number of rows of H_r and H_{st}. Equation (11b) is sufficient (not necessary) for (9c), as becomes clear if we multiply (11b) with $[x \;\; 1]$ from the left and $[x \;\; 1]'$ from the right. The scalar parameters ϵ and ρ are arbitrarily small and greater than zero in order to enforce a strictly positive PWQ function and exponential stability, respectively. Equation (11e) ensures that a quadratic upper bound for the Lyapunov function exists, by enforcing a quadratic Lyapunov function around the origin, which is covered by region $r = 1$.

Theorem 3 *[4] If a PWQ function $P_{\mathrm{PWQ}}(x)$ is obtained with (11) and with region $r = 1$ containing the origin, the system is exponentially stable.*

4 Algorithm

The proposed algorithm provides the minimal prediction horizon N for which the resulting controller covers the maximum control invariant set with stability and feasibility guarantees.

Algorithm A

1. Initialize the prediction horizon to $N = 1$.
2. Solve (3) for prediction horizon N as an mp-QP.
3. Compute the polyhedral invariant set $\mathcal{X}_I$ as described in Section 3.1.1.
4. Apply the method in 3.1.2 to check if $\mathcal{X}_I$ is the maximal control invariant set. If not, set $N = N + 1$ and goto 1.
5. Compute a PWQ Lyapunov function over $\mathcal{X}_I$ as described in Section 3.2 to verify stability. If not successful, set $N = N + 1$ and goto 1, else end.

The existence of a PWQ Lyapunov function is guaranteed for sufficiently large horizons N (Theorem 2). However, since the LMI computation is merely sufficient, there is no guarantee that a PWQ function guaranteeing stability will be found. It is therefore advisable to abort this procedure once step 5 has been executed a large number of times.

5 Numerical Examples

Extensive simulations were performed in order to assess the potential gain of the post-processing procedure introduced in this paper. The post-analysis tools were used to obtain the minimum prediction horizon N which provides a stabilizing and infinite-time feasible controller covering the maximum control invariant set $\mathcal{C}_\infty$ (i.e., Algorithm A). The infinite horizon controller in [5] was used as a basis for comparison. Both controllers cover the same set of states and provide stability and feasibility properties. In this section we will compare complexity and performance of the two controllers. A total of 20 random stable systems were created for $n = 2$ and $n = 3$ states respectively whereby the number of inputs was fixed at $m = 2$. The inputs for all systems were constrained to $-1 \leq u_{1,2} \leq 1$ and the states were limited to $-10 \leq x_i \leq 10$ ($i = 1, 2, 3$). Two different performance objectives were considered in the original problem formulation (3). We covered the cases of small and large weights on the input, i.e., $\mathcal{R}_1 = 0.1I$ and $\mathcal{R}_2 = 10I$ with $\mathcal{Q} = I$ for both cases. Therefore a total of 80 cases were studied.

The results for the 2 and 3 state systems are given in Figures 1 and 2 for $\mathcal{R}_2 = 0.1I$, where the prediction horizon, number of regions and performance is compared to the infinite horizon solution. The run-time for Algorithm A is also given.

Figures 1(a) and 2(a) depict the decrease in the necessary prediction horizon. Note that knowledge of the minimum prediction horizon necessary to stabilize the maximum control invariant set can also serve to speed up procedures which rely on on-line solution of the optimization problem (MPC). The average decrease in the prediction horizon over all 80 simulation runs is 72% which corresponds to an average decrease in the number of regions of 69%. The decrease in number of regions is depicted in Figures 1(b) and 2(b). Figures 1(c) and 2(c) depict the decrease in closed-loop performance compared to the infinite horizon solution. The performance index was obtained by gridding the state space and subsequently computing the closed-loop cost for all of the initial states. The average decrease in performance is 0.06%. As previously stated, the LMI analysis in Section 3.2 is not guaranteed to provide a solution due to the conservative formulation. However, we have observed that the LMI was successful in 100% of the analyzed cases.

One important aspect is the computational complexity of the proposed off line computations. We found that most of the time is spent on solving the LMI problem in (11), followed by the invariant set computation and the mp-QP (see Figures 1(d) and 2(d)). The entire procedure may take from under a minute (No. of regions $R < 100$) up to several hours[1] (No. of regions $R > 1000$). We do not believe that going beyond $R = 1000$ regions in dimensions above $n = 3$ makes sense, since the necessary runtime becomes prohibitive. We have been able to successfully apply algorithm A to several fourth order systems with more than 500 regions. However, the maximum control invariant set for random fourth order systems may easily consist of more than 10000 regions. For problems of this size we have run into numerous numerical difficulties in all three components of Algorithm A (i.e. mp-QP, invariant set computation and LMI) such that we decided to abort simulation of systems beyond a dimension of $n = 3$. It should be stressed, however, that we do not see the bottleneck in the dimension alone, it is the combination of large prediction horizons and higher dimensional systems which make excessive computation necessary.

	$n=2$ $\mathcal{R}_1$ A / [5]	$n=3$ $\mathcal{R}_1$ A / [5]	$n=2$ $\mathcal{R}_2$ A / [5]	$n=3$ $\mathcal{R}_2$ A / [5]
Horizon N	1.2/8.4	2.5/8.5	1.4/4.6	2.3/4.9
No. Regions R	19/218	224/897	27/66	172/267
Performance	0.20%	0.01%	0.03%	0.02%

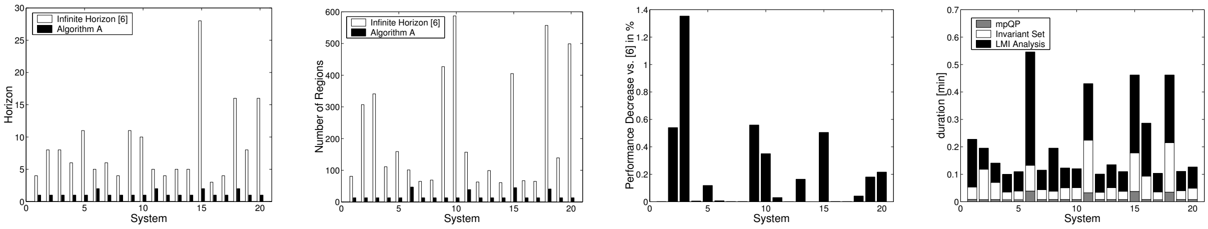
Table 1: Average prediction horizon N, number of regions R and performance decrease of Algorithm A versus [5] for the combinations of cost weights $\mathcal{R}_1 = 0.1I$, $\mathcal{R}_2 = 10I$ ($\mathcal{Q} = I$) and state space dimension $n = 2$ and $n = 3$. 20 Simulations were run for each combination.

Differences in the solutions for the different cost objectives $\mathcal{R}_1 = 0.1I$ and $\mathcal{R}_2 = 10I$ were also observed. For the infinite horizon controller in [5], the prediction horizon which is needed for the controller to cover the maximum control invariant set is much smaller for large weights on the input, i.e., for $\mathcal{R}_2$ (see Table 1). This is due to the fact that the unconstrained region $\mathcal{X}_I$ is significantly larger if the weight on the input is large (this also follows from monotonicity of the Riccati equation). However, the different cost objectives have little impact on the controllers obtained with Algorithm A and the differences in relative performance versus [5] are mainly due to the strong variations in the complexity of the infinite horizon controllers when different cost weights $\mathcal{R}$ in (3) are used.

6 Conclusion

We have shown, how to verify infinite time feasibility and stability for constrained optimal control applied as RHC. The procedure requires the off-line computation of the feedback solution to the optimal control problem, but the results remain valid for on-line computations as well. Though the focus of this paper was on linear systems, the procedure can be applied to other PWA (e.g. hybrid) systems without any modification to the proposed algorithm. The extensive examples have shown that the proposed method allows for significantly

[1] 2GHz PC using NAG for LPs and SeDuMi [10] for the LMI problems.



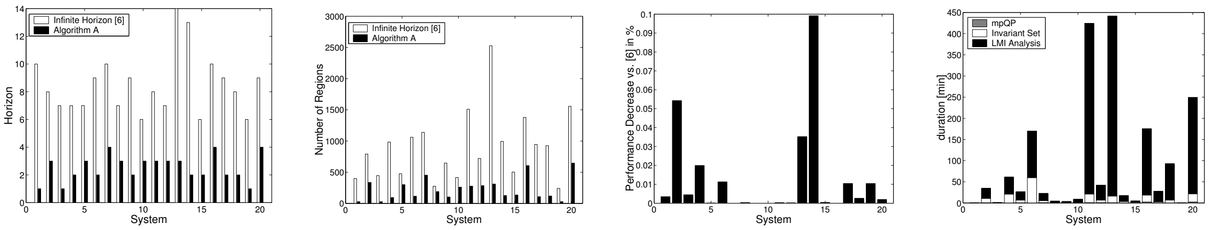
(a) Prediction Horizon Algorithm A vs [5].

(b) Region Number Algorithm A vs [5].

(c) Performance Decrease Algorithm A vs [5].

(d) Runtime Algorithm A.

Figure 1: Simulation results for 20 second-order systems with $\mathcal{R}_1 = 0.1I$.



(a) Prediction Horizon Algorithm A vs [5].

(b) Region Number Algorithm A vs [5].

(c) Performance Decrease Algorithm A vs [5].

(d) Runtime Algorithm A.

Figure 2: Simulation results for 20 third-order systems with $\mathcal{R}_1 = 0.1I$.

shorter prediction horizons than other RHC methods without a significant decrease in performance. This in turn results in a significantly lower computational burden for both off-line and on-line solutions of the optimization problem. However, the numerical results have also shown that the proposed Algorithm is not applicable to most higher order systems. Our recent work has focussed on making the presented techniques tractable for larger problems [6].

The presented algorithms can be downloaded from:

`http://control.ethz.ch/~grieder`

References

[1] A. Bemporad, K. Fukuda, and F.D. Torrisi. Convexity recognition of the union of polyhedra. *Computational Geometry*, 18:141–154, April 2001.

[2] A. Bemporad, M. Morari, V. Dua, and E.N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, January 2002.

[3] F. Borrelli. *Constrained Optimal Control Of Linear And Hybrid Systems*, volume 290 of *Lecture Notes in Control and Information Sciences*. Springer, 2003.

[4] G. Ferrari-Trecate, F.A. Cuzzola, D. Mignone, and M. Morari. Analysis of discrete-time piecewise affine and hybrid systems. *Automatica*, 38:2139–2146, 2002.

[5] P. Grieder, F. Borrelli, F.D. Torrisi, and M. Morari. Computation of the constrained infinite time linear quadratic regulator. In *Proc. 2003 American Control Conference*, Denver, Colorado, USA, June 2003.

[6] P. Grieder and M. Morari. Complexity reduction of receding horizon control. Technical report, Automatic Control Lab, ETHZ, Switzerland, December 2002.

[7] M. Johannson and A. Rantzer. Computation of piece-wise quadratic Lyapunov functions for hybrid systems. *IEEE Trans. Automatic Control*, 43(4):555–559, 1998.

[8] E. C. Kerrigan. *Robust Constraints Satisfaction: Invariant Sets and Predictive Control*. PhD thesis, Department of Engineering, The University of Cambridge, Cambridge, England, 2000.

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

[10] J.F. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, pages 625–653, October 1999.

[11] M. Sznaier and M.J. Damborg. Suboptimal control of linear systems with state and control inequality constraints. In *Proc. 26th IEEE Conf. on Decision and Control*, volume 1, pages 761–762, December 1987.