Approximate Explicit Receding Horizon Control of Constrained Nonlinear Systems

Tor A. Johansen [a]

[a] *Department of Engineering Cybernetics, Norwegian University of Science and Technology, N-7491 Trondheim, Norway.*

Abstract

An algorithm for the construction of an explicit piecewise linear state feedback approximation to nonlinear constrained receding horizon control is given. It allows such controllers to be implemented via an efficient binary tree search, avoiding real-time optimization. This is of significant benefit in applications that requires low real-time computational complexity or software complexity. The method has *a priori* guarantee of asymptotic stability with region of attraction being a close inner approximation to the stabilizable set. This is achieved by ensuring that the approximation error does not exceed the stability margin.

1 Introduction

Receding horizon control (RHC) traditionally involves the solution of a constrained finite-horizon optimal control problem at each sampling instant (Keerthi and Gilbert 1988, Mayne and Michalska 1990, Michalska and Mayne 1993). Such real-time optimization is, however, restricted to applications that allow slow sampling and high-performance computers. Parisini and Zoppoli (1995) suggested to approximate the implicitly defined state feedback law of nonlinear constrained RHC by a nonlinear function using neural networks. Hence, the real-time optimization is replaced by a simpler function evaluation, extending the applicability of nonlinear RHC to applications that may require fast sampling and inexpensive computers. For linear constrained RHC an alternative function approximation method was suggested by Johansen and Grancharova (2002*a*). It has the advantage that the neural network approximation is replaced by a computationally more favorable piecewise linear (PWL) approximation implemented via a binary search tree. More importantly, it is guaranteed that stability is not lost due to the approximation error. This is achieved by choosing a suitable tolerance on the sub-optimal cost function error, and it is

⋆ Corresponding author: `Tor.Arne.Johansen@itk.ntnu.no`

given a constructive approximation algorithm that guarantees that this tolerance and the constraints are fulfilled. In contrast, the results in (Parisini and Zoppoli 1995) has some limitations as they focus on existence on function approximations, and no constructive method with a priori guarantees is given to determine the nonlinear function approximation such that asymptotic stability is not lost due to the approximation error, or that the constraints are not violated. In the present work we extend recent results for linear RHC by Johansen and Grancharova (2002*a*), see also (Johansen and Grancharova 2002*b*) to nonlinear constrained RHC problems. As we shall see, convexity is a simplifying property. We will, however, outline how global optimization can be utilized if this assumption is not fulfilled. Approximate explicit solutions can also be computed using dynamic programming approximations such as (Bertsekas and Tsitsiklis 1998, Rantzer 1999).

The ideas presented here can be viewed as a natural extension of explicit solutions recently derived for linear input and state constrained RHC and linear quadratic regulation (Bemporad *et al.* 2002, Bemporad *et al.* 2000*b*, Bemporad *et al.* 2000*a*, Bemporad and Filippi 2001). In this case the *exact* state feedback solution takes the form of a PWL function that can be computed off-line using multi-parametric quadratic programming (mp-QP) algorithms (Bemporad *et al.* 2002, Tøndel *et al.* 2001) or multi-parametric linear programming algorithms. In the nonlinear case no exact state feedback solution can be represented explicitly, in general, but approximations can be found using multi-parametric nonlinear programming (mp-NLP) (Fiacco 1983, Johansen 2002). The algorithm presented here can be utilized as an approximate mp-NLP algorithm with applicability beyond RHC. The mp-NLP algorithm in (Johansen 2002) generates an approximate PWL solution by locally approximating mp-NLPs with mp-QP sub-problems solved using the mp-QP solver presented in (Tøndel *et al.* 2001). In the present work we only solve NLP subproblems.

We take as a starting point a discrete-time nonlinear RHC formulation similar to (Chen and Allgöwer 1998) using also some elements of (Mayne *et al.* 2000) and dual-mode control, (Michalska and Mayne 1993). Since the contribution of the present work is essentially on an efficient implementation of RHC without real-time optimization, it is clear that similar algorithms can be derived along the same lines for alternative nonlinear RHC formulations such as those given in (Mayne and Michalska 1990, Keerthi and Gilbert 1988, Jadbabaie *et al.* 2001, De Nicolao *et al.* 1996) and others.

The following notation will be used throughout this paper. $A \succ 0$ means that the square matrix A is positive definite, and $A \succeq 0$ positive semi-definite. For $x \in \mathbb{R}^n$ the Euclidean norm is $||x|| = \sqrt{x^T x}$ and the weighted norm is defined for some symmetric matrix $A \succ 0$ as $||x||_A = \sqrt{x^T A x}$. The maximum and minimum eigenvalues of a square matrix A are denoted $\lambda_{max}(A)$ and $\lambda_{min}(A)$, respectively.

2 Nonlinear RHC formulation

Consider the discrete-time non-linear system

$$x(t+1) = f(x(t), u(t)) \tag{1}$$

where $x(t) \in \mathbb{R}^n$ is the state, and $u(t) \in \mathbb{R}^m$ is the input. We assume the control objective is regulation to the origin. For the current $x(t)$, a typical RHC algorithm, (Chen and Allgöwer 1998, Mayne *et al.* 2000), solves the optimization problem

$$V^*(x(t)) = \min_U J(U, x(t)) \tag{2}$$

subject to $x_{t|t} = x(t)$ and

$$\begin{array}{l} y_{\min} \le y_{t+k|t} \le y_{\max}, \quad k = 1, ..., N \\ u_{\min} \le u_{t+k} \le u_{\max}, \quad k = 0, 1, ..., N-1, \\ x_{t+N|t} \in \Omega \\ x_{t+k+1|t} = f(x_{t+k|t}, u_{t+k}), \quad k = 0, 1, ..., N-1 \\ y_{t+k|t} = Cx_{t+k|t}, \quad k = 1, 2, ..., N \end{array} \tag{3}$$

with $U = \{u_t, u_{t+1}, ..., u_{t+N-1}\}$ and the cost function given by

$$J(U, x(t)) = \sum_{k=0}^{N-1} \left(||x_{t+k|t}||_Q^2 + ||u_{t+k}||_R^2 \right) + ||x_{t+N|t}||_P^2 \tag{4}$$

N is a finite horizon, and the following assumptions are made:

A1. $P, Q, R \succ 0$.

A2. $y_{min} < 0 < y_{max}$ and $u_{min} < 0 < u_{max}$.

A3. The function f is twice continuously differentiable, with $f(0, 0) = 0$.

The compact and convex terminal set Ω is defined by

$$\Omega = \{x \in \mathbb{R}^n \mid x^T P x \le \alpha\} \tag{5}$$

where $P \in \mathbb{R}^{n \times n}$ and $\alpha > 0$ will be specified shortly. An optimal solution to the problem (2)-(3) is denoted $U^* = \{u_t^*, u_{t+1}^*, ..., u_{t+N-1}^*\}$, and the control input is chosen according to the receding horizon policy $u(t) = u_t^*$. This and similar optimization problems can be formulated in a concise form

$$V^*(x) = \min_U J(U,x) \quad \text{subject to} \quad G(U,x) \leq 0 \tag{6}$$

The problem (6) defines an mp-NLP, since it is an NLP in U parameterized by x. Define the set of N-step feasible initial states as follows

$$X_F = \{x \in \mathbb{R}^n \,|G(U,x) \leq 0 \text{ for some } U \in \mathbb{R}^{Nm}\} \tag{7}$$

Suppose Ω is a control invariant set, such that X_F is a subset of the N-step stabilizable set (Kerrigan and Maciejowski 2000). Notice that A2-A3 implies that the origin is an equilibrium and interior point in X_F. It remains to specify $P \succ 0$ and $\alpha > 0$ such that Ω is a control invariant set. For this purpose, we use the ideas of (Chen and Allgöwer 1998), where one simultaneously determine a linear feedback such that Ω is positively invariant under this feedback. Define the local linearization at the origin

$$A = \frac{\partial f}{\partial x}(0,0), \quad B = \frac{\partial f}{\partial u}(0,0) \tag{8}$$

and make the following assumption

A4. (A,B) is stabilizable.

Let K denote the associated LQ optimal gain matrix, such that $A_0 = A - BK$ is strictly Hurwitz. The following lemmas are discrete-time versions of Lemma 1 in (Chen and Allgöwer 1998):

Lemma 1 *If $\kappa > 0$ is such that $A_0 + \kappa I$ is strictly Hurwitz, the Lyapunov equation*

$$(A_0 + \kappa I)^T P (A_0 + \kappa I) - P = -Q - K^T R K \tag{9}$$

has a unique solution $P \succ 0$.

Proof. The result is trivial since $Q + K^T R K \succ 0$. □

Lemma 2 *Let κ and P satisfy the conditions in Lemma 1. Then there exists a constant $\alpha > 0$ such that Ω defined in (5) satisfies*

(1) $\Omega \subset \mathcal{C} = \{x \in \mathbb{R}^n \mid u_{min} \leq -Kx \leq u_{max}, y_{min} \leq Cx \leq y_{max}\}$.
(2) The autonomous nonlinear system

$$x(t+1) = f(x(t), -Kx(t)) \tag{10}$$

is asymptotically stable for all $x(0) \in \Omega$, i.e. Ω is positively invariant.
(3) The infinite-horizon cost for the system (10)

$$J_\infty(x(t)) = \sum_{k=0}^{\infty} \left(||x_{t+k|t}||_Q^2 + ||Kx_{t+k|t}||_R^2 \right) \tag{11}$$

satisfies $J_\infty(x) \leq x^T P x$ *for all* $x \in \Omega$.

Proof. Due to A2 one may define a set of the form

$$\Omega_{\alpha_1} = \{x \in \mathbb{R}^n \mid x^T P x \leq \alpha_1\} \tag{12}$$

with $\alpha_1 > 0$, such that $\Omega_{\alpha_1} \subseteq \mathcal{C}$, i.e. an ellipsoidal inner approximation Ω_{α_1} to the polyhedron $\mathcal{C}$ where the input and state constraints are satisfied. Hence, the first claim holds for all $\alpha \in (0, \alpha_1]$.

Define the positive definite function $W(x) = x^T P x$. Along trajectories of the autonomous system (10) we have

$$\begin{aligned} W(x(t+1)) - W(x(t)) &= (A_0 x(t) + \phi(x(t)))^T P (A_0 x(t) + \phi(x(t))) - x^T(t) P x(t) \\ &= x^T(t) \left(A_0^T P A_0 - P \right) x(t) + 2x^T(t) P \phi(x(t)) \end{aligned} \tag{13}$$

where $\phi(x) = f(x, -Kx) - A_0 x$ satisfies $\phi(0) = 0$. It is straightforward to show that

$$x^T P \phi(x) \leq L_\phi \left(\frac{\lambda_{max}(P)}{\lambda_{min}(P)} \right) ||x||_P^2 \tag{14}$$

where L_ϕ is a Lipschitz constant in Ω_{α_1} (which must exist because f is differentiable). We choose $\alpha \in (0, \alpha_1]$ such that $L_\phi \leq \kappa \lambda_{min}(P)/\lambda_{max}(P)$, which is possible because $\partial\phi/\partial x(0) = 0$ and ϕ is twice differentiable. From (9)

$$W(x(t+1)) - W(x(t)) \leq x^T(t) \left(A_0^T P A_0 - P + 2\kappa P \right) x(t) \tag{15}$$

$$= -x^T(t) \left(Q + K^T R K + \kappa^2 P \right) x(t) \tag{16}$$

and positive invariance of Ω follows since Ω is a level set of W.

Notice that from (16) we have

$$W(x(\infty)) - W(x(0)) \leq -J_\infty(x(0)) - \kappa^2 \sum_{t=0}^{\infty} ||x(t)||_P^2 \tag{17}$$

and the third claim holds because $W(x(\infty)) = 0$. □

It follows from (Mayne *et al.* 2000, Chen and Allgöwer 1998) that the RHC makes the origin asymptotically stable with region of attraction X_F. However, in parametric programming problems one seeks the solution $U^*(x)$ as an explicit function of the parameters x in some set $\mathbb{X} \subseteq X_F \subseteq \mathbb{R}^n$ (Fiacco 1983). The explicit solution allows us to replace the computationally expensive real-time optimization with a simple function evaluation. Unfortunately, for general nonlinear functions J and G an *exact* explicit solution cannot be found. In the remaining of this paper we will develop an algorithm for constructing an explicit approximate solution such that the approximation error that is introduced does not lead to loss of stability.

A procedure for selecting P, κ and α is given in (Chen and Allgöwer 1998). One feature of the explicit approach is that it is not generally desirable to select Ω as large as possible since this may lead to loss of performance and robustness. Moreover, any computational advantages of choosing Ω large are less important since the optimization will be carried out entirely off-line.

3 Nonlinear Parametric Programming

The numerical computations involved in constructing the approximate explicit PWL state feedback is simplified under the following assumption:

A5. J and G are jointly convex for all $(U, x) \in \mathbb{U} \times \mathbb{X}$, where $\mathbb{U} = [u_{min}, u_{max}]^N$ is the set of admissible inputs.

The optimal cost function can now be shown to have some regularity properties (Mangasarian and Rosen 1964):

Theorem 1 *X_F is a closed convex set, and $V^* : X_F \to \mathbb{R}$ is convex and continuous.* $\square$

Convexity of X_F and V^* is a direct consequence of A5, while continuity of V^* can be established under weaker conditions (Fiacco 1983). We remark that V^* is in general not differentiable, but properties such as local differentiability and directional differentiability can be investigated as shown in e.g. (Fiacco 1983). Regularity properties of the solution function U^* is a slightly more delicate issue, and essentially relies on stronger assumptions such as strict joint convexity that ensure uniqueness of the solution. Since this is not needed with our approach (in contrast to (Parisini and Zoppoli 1995) where continuity of U^* was assumed because their focus is on the approximation error in the solution rather than the value function V^*), we only refer to Appendix A and (Fiacco 1983) for some discussion on this issue.

The main idea is to construct a feasible PWL approximation to U^* on $\mathbb{X}$, where the constituent affine functions are defined on hypercubes covering $\mathbb{X}$. The accuracy of approximation will be measured by the difference between the optimal and sub-optimal cost functions rather than the difference between the exact and approximation solutions. Since the optimal cost function V^* cannot be assumed known, convexity is exploited to compute simple bounds to be used both for constructing the approximate solution, similar to chapter 9 in (Fiacco 1983), and quantify the associated perturbation in the stability analysis, see (Johansen and Grancharova 2002*a*) for similar results in the context of linear constrained RHC.

Consider the vertices $\mathcal{V} = \{v_1, v_2, ..., v_M\}$ of any bounded polyhedron $X_0 \subseteq X_F$. Define the affine function $\overline{V}(x) = \overline{V}_0 x + \bar{l}_0$ as the solution to the following linear program (LP):

$$\min_{\overline{V}_0, \bar{l}_0} \left(\overline{V}_0 v + \bar{l}_0 \right) \tag{18}$$

$$\text{subject to } \overline{V}_0 v_i + \bar{l}_0 \geq V^*(v_i), \quad \text{for all } i \in \{1, 2, ..., M\} \tag{19}$$

Likewise, define the convex PWL function

$$\underline{V}(x) = \max_{i \in \{1,2,...,M\}} \left(V^*(v_i) + \nabla^T V^*(v_i)(x - v_i) \right) \tag{20}$$

If V^* is not differentiable at v_i, then $\nabla V^*(v_i)$ is taken as any sub-gradient of V^* at v_i. $\underline{V}$ and $\overline{V}$ have the following properties, see also also chapter 9.2 of (Fiacco 1983):

Theorem 2 *Consider any bounded polyhedron* $X_0 \subseteq X_F$. *Then* $\underline{V}(x) \leq V^*(x) \leq \overline{V}(x)$ *for all* $x \in X_0$.

Proof. Let $x \in X_0$ be arbitrary, and consider the convex combination $x = \sum_i \alpha_i v_i$ where $\alpha_i \geq 0$ satisfies $\sum_i \alpha_i = 1$:

$$V^*(x) \leq \sum_{i=1}^{M} \alpha_i V^*(v_i) \leq \sum_{i=1}^{M} \alpha_i \left(\overline{V}_0 v_i + \bar{l}_0 \right) = \overline{V}_0 x + \bar{l}_0$$

The lower bound $\underline{V}$ follows from the convexity of V^*, since the sub-gradient inequality $V^*(x) \geq V^*(v) + \nabla^T V^*(v)(x - v)$ holds for all $v \in X_0$ (Rockafellar 1970).

□

We suggest to select a local linear approximation to the solution that minimizes the value function approximation error subject to feasibility of the solution, similar to (Bemporad and Filippi 2001):

Lemma 3 *Consider any bounded polyhedron $X_0 \subseteq X_F$ with vertices $\{v_1, v_2, ..., v_M\}$. If K_0 and g_0 solve the convex NLP*

$$\min_{K_0,g_0} \sum_{i=1}^{M} \Big(J(K_0 v_i + g_0, v_i) - V^*(v_i) + \beta ||K_0 v_i + g_0 - U^*(v_i)||_2^2 \Big) \tag{21}$$

$$\text{subject to } \; G(K_0 v_i + g_0, v_i) \leq 0, \quad i \in \{1, 2, ..., M\} \tag{22}$$

then $\hat{U}_0(x) = K_0 x + g_0$ is feasible for the mp-NLP (6) for all $x \in X_0$.

Proof. We remark that the NLP is convex because the cost function and constraints are convex functions, being the composition of convex functions with linear functions. Let $x \in X_0$ be arbitrary, and consider the convex combination $x = \sum_i \alpha_i v_i$ where $\alpha_i \geq 0$ satisfies $\sum_i \alpha_i = 1$:

$$G(K_0 x + g_0, x) = G\left(\sum_{i=1}^{M} \alpha_i (K_0 v_i + g_0), \sum_{i=1}^{M} \alpha_i v_i \right) \tag{23}$$

$$\leq \sum_{i=1}^{M} \alpha_i G\left(K_0 v_i + g_i, v_i\right) \; \leq \; 0 \tag{24}$$

□

In general, the NLP defined in this lemma need not have a feasible solution. As a partial remedy, the following result shows that at least for sufficiently small polyhedra X_0, feasibility can be guaranteed:

Lemma 4 *Let $X_0 \subseteq X_F$ be a sufficiently small bounded polyhedron with non-empty interior. Then there exists an affine function $\tilde{U}(x)$ such that $G(\tilde{U}(x), x) \leq 0$ for all $x \in X_0$.*

Proof. Since $X_0 \subseteq X_F$ is small, it follows from Theorem 5 in Appendix A that some unique and continuous feasible solution function $U(x)$ exists a neighborhood that contains X_0. Since G is convex it is straightforward to construct an affine support $\tilde{U}(x)$. □

Since $\hat{U}_0(x)$ defined in Lemma 3 is feasible in X_0, it follows that

$$\hat{V}(x) = J(\hat{U}_0(x), x) \tag{25}$$

is an upper bound on $V^*(x)$ in X_0 such that for all $x \in X_0$

$$0 \; \leq \; \hat{V}(x) - V^*(x) \leq \varepsilon_0 \tag{26}$$

where

$$\varepsilon_0 = -\min_{x \in X_0} \left(-\hat{V}(x) + \underline{V}(x) \right) \tag{27}$$

Computing ε_0 requires the solution of the NLP (27). If $\underline{V}$ is conservatively chosen as linear $\underline{V}(x) = V^*(v_i) + \nabla^T V^*(v_i)(x - v_i)$, cf. (20), this NLP is concave since $\hat{V}$ is convex. Hence the optimization can be done efficiently since X_0 is a polyhedron and it suffices to compare the solution at its extreme points due to the concavity (Horst and Tuy 1993).

4 mp-NLP algorithm

Consider a hypercube $\mathbb{X} \subset \mathbb{R}^n$ where we seek to approximate the solution function $U^*(x)$ to the mp-NLP (6). In order to keep the real-time computational complexity at a minimum, we require that the approximating function is PWL with a state space partition that is orthogonal and can be represented by a $k-d$-tree, (Bentley 1975), such that the real-time search complexity is logarithmic with respect to the number regions in the partition (Grancharova and Johansen 2002). The $k-d$-tree is a hierarchical data structure where a hypercube can be sub-divided into smaller hybercubes allowing the local resolution to be adapted. When searching the tree, only one scalar comparison is required at each level. Initially the algorithm will consider the whole region $X_0 = \mathbb{X}$. The main idea of the approximate mp-NLP algorithm is to compute the solution of the problem (6) at the 2^n vertices of the hypercube X_0, by solving up to 2^n NLPs. Based on these solutions, assuming they are all feasible, we compute a feasible local linear approximation function $\hat{U}_0$ to the optimal solution function U^*, restricted to the hypercube X_0, using Lemma 3. If such an approximation exists, and the maximal cost function error ε_0 in X_0 is smaller than some prescribed tolerance $\bar{\varepsilon} > 0$, no further refinement of the region X_0 is needed. Otherwise, we partition X_0 into two hypercubes, and repeat the procedure described above for each of these.

Algorithm 1 (approximate mp-QP)

1. Initialize the partition to the whole hypercube, i.e. $\mathcal{P} = \{\mathbb{X}\}$. Mark the hypercube $\mathbb{X}$ as unexplored.

2. Select any unexplored hypercube $X_0 \in \mathcal{P}$. If no such hypercube exists, the algorithm terminates successfully.

3. Solve the NLP (6) for x fixed to each of the vertices of the hypercube X_0 (some of these NLPs may have been solved in earlier steps). If all solutions are feasible, go to step 4. Otherwise, compute the size of X_0. If it is smaller than some tolerance, mark X_0 explored and infeasible. Otherwise, go to step 8.

4. If $0 \in X_0$, choose $\hat{U}_0(x) = -Kx$ and go to step 5 . Otherwise, go to step 6.

5. If $X_0 \subseteq \Omega$, mark X_0 as explored and go to step 2. Otherwise, go to step 8.

6. Compute an affine state feedback $\hat{U}_0$ using Lemma 3, as an approximation to be used in X_0. If no feasible solution was found, go to step 8.

7. Compute the error bound ε_0, using Theorem 3 and (27). If $\varepsilon_0 \leq \overline{\varepsilon}$, mark X_0 as explored and feasible, and go to step 2.

8. Split the hypercube X_0 into two hypercubes X_1 and X_2 using some heuristic rule. Mark both unexplored, remove X_0 from $\mathcal{P}$, add X_1 and X_2 to $\mathcal{P}$ and go to step 2.

□

The PWL approximation generated by Algorithm 1 is denoted $\hat{U} : \mathbb{X}' \to \mathbb{R}^{Nm}$, where $\mathbb{X}'$ is the union of hypercubes where a feasible solution has been found. It is an inner approximation to X_F and the approximation accuracy is determined by the tolerance in step 3.

We remark that $\hat{U}$ is generally not continuous. Due to step 4 we require that in a neighborhood of the origin the LQ optimal gain matrix is used, as in dual-mode RHC, (Michalska and Mayne 1993).

Step 8 needs further specification of how a hypercube is being partitioned. A hypercube is split into two equal parts by an axis-orthogonal hyperplane that goes through its center. As in (Grancharova and Johansen 2002), the main idea is to select the hyperplane where the error between the solutions on each side of the hyperplane is largest (before splitting). This is implemented by comparing the solutions at the vertices of the hypercube. It is reasonable to expect that this may give a significant reduction in the error in both hypercubes after splitting.

Theorem 3 *Assume the partitioning rule in step 8 guarantees that the error decreases by some minimum amount or factor at each split. Then Algorithm 1 terminates with an approximate solution function $\hat{U}$ that is feasible and satisfies*

$$0 \leq J(\hat{U}(x), x) - V^*(x) \leq \overline{\varepsilon} \tag{28}$$

for all $x \in \mathbb{X}'$.

Proof. If the algorithm terminates, the specified tolerance is met because of steps 7 and 8. Since V^* is continuous it is clear that a $k-d$-tree partition will lead to an approximation with arbitrary uniform accuracy provided the hypercubes are sufficiently small. According to lemma 4, this approximation will be feasible, and since the partitioning rule ensures that the error decreases by some minimum amount or factor at each step, the algorithm will indeed terminate after a finite number of steps. □

If convexity does not hold, global optimization is generally needed if theoretical guarantees are required:

(1) The NLP (6) must be solved using global optimization in step 3.
(2) The NLP (21)-(22) must be solved using global optimization in step 6.
(3) The computation of the error bound ε_0 in step 7 must rely on global optimization, or convex underestimation.
(4) The heuristics in step 8 may be modified to be efficient also in the non-convex case.

5 Stability

The exact RHC will make the origin asymptotically stable (Mayne *et al.* 2000, Chen and Allgöwer 1998). We show below that asymptotic stability is inherited by the approximate RHC under an assumption on the tolerance $\bar{\varepsilon}$:

A6. Assume the partition $\mathcal{P}$ generated by Algorithm 1 has the property that for any hypercube $X_0 \in \mathcal{P}$ that does not contain the origin

$$\bar{\varepsilon} \leq \gamma \min_{x \in X_0} ||x||_P^2 \tag{29}$$

where $\gamma \in (0, 1)$ is given.

Theorem 4 *The origin is an asymptotically stable equilibrium point for the system (1) in closed loop with the approximate explicit RHC given by Algorithm 1, for all* $x(0) \in \mathbb{X}'$.

Proof. Let $x(t) \in \mathbb{X}'$ be arbitrary and the associated optimal control be denoted U^*. At time $t+1$ consider $\tilde{U}_{t+1} = \{u^*_{t+1}, u^*_{t+2}, ..., u^*_{t+N-1}, -Kx^*_{t+N|t}\}$, where $x^*_{t+k|t}$ is the state at time $t+k$ associated with U^*. Since U^* is N-step feasible, $x^*_{t+N|t} \in \Omega$. Hence, $\tilde{U}_{t+1}$ is feasible and the tail of the trajectories remain feasible since Ω is positively invariant. Since $\hat{V}(x)$ is an upper bound on $V^*(x)$, standard arguments, (Bemporad and Filippi 2001), give

$$\begin{aligned} V^*(x(t+1)) &\leq \hat{V}(x(t+1)) \\ &= \hat{V}(x(t)) - ||x(t)||_Q^2 - ||u(t)||_R^2 - ||x^*_{t+N|t}||_P^2 \\ &\quad + ||f(x^*_{t+N|t}, -Kx^*_{t+N|t})||_P^2 + ||x^*_{t+N|t}||_Q^2 + ||Kx^*_{t+N|t}||_R^2 \\ &\leq \hat{V}(x(t)) - ||x(t)||_Q^2 - ||u(t)||_R^2 \end{aligned} \tag{30}$$

The first inequality is due to Theorem 2, while the second inequality is due to Lemma 2, eq. (16). Let $\Omega_{\alpha_2} = \{x \in \Omega \mid x^T P x \leq \alpha_2\}$ be such that $u = -Kx$ for all $x \in \Omega_{\alpha_2}$. Such a set with non-empty interior exists due to step 4 in Algorithm 1. Then, for $x \notin \Omega_{\alpha_2}$ it follows from (30) and assumption A6 that

$$V^*(x(t+1)) - V^*(x(t)) \leq \bar{\varepsilon} - ||x(t)||_Q^2 - ||u(t)||_R^2 \quad (31)$$
$$\leq -(1-\gamma)||x(t)||_Q^2 \ < \ 0 \quad (32)$$

It follows that $x(t) \rightarrow \Omega_{\alpha_2}$ as $t \rightarrow \infty$. Asymptotic stability of the origin can be concluded due to Lemma 2 because $u = -Kx$ in the positively invariant set Ω_{α_2}.

□

We remark that the tolerance $\bar{\varepsilon}$ can be chosen *a priori* for each hypercube X_0 to satisfy (29). Hence, one can guarantee *a priori* that the PWL feedback law generated by Algorithm 1 will be asymptotically stabilizing. The parameter γ in (29) determines the approximation accuracy and degree of sub-optimality. A γ close to one is sufficient for stability, but γ close to zero give less approximation error and sub-optimality.

6 Simulation example

The example is taken from (Chen and Allgöwer 1998), where the following system is studied

$$\dot{x}_1 = x_2 + u(1 + x_1)/2 \quad (33)$$
$$\dot{x}_2 = x_1 + u(1 - 4x_2)/2 \quad (34)$$

The origin is an unstable equilibrium point, with a stabilizable (but uncontrollable) linearization. We discretize this system using a sampling interval $T_s = 0.1$. The control objective is given by the weighting matrices

$$Q = \frac{T_s}{2} I_{2\times 2}, \qquad R = T_s \quad (35)$$

Notice the scaling with T_s that ensures the discrete-time formulation is comparable to the continuous-time formulation. The terminal penalty is given by the solution to the Lyapunov equation

$$P = \begin{pmatrix} 16.5926 & 11.5926 \\ 11.5926 & 16.5926 \end{pmatrix} \quad (36)$$

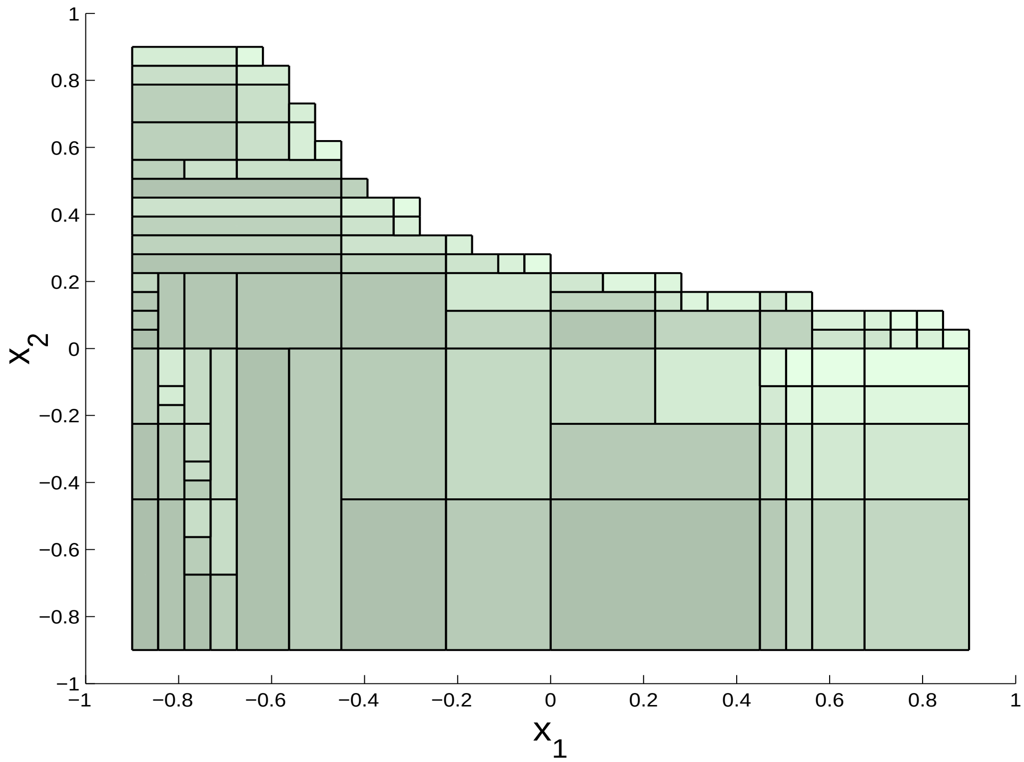

Fig. 1. State space partitioning of the approximate PWL explicit optimal control.

and a positively invariant terminal region is

$$\Omega = \{x \in \mathbb{R}^2 \mid x^T P x \leq 0.7\} \tag{37}$$

as in (Chen and Allgöwer 1998). The prediction horizon is $N = 1.5/T_s = 15$. In the approximate mp-NLP algorithm we choose $\gamma = 0.25$, and get the partition shown in Figure 1. It contains 105 regions and the associated solution and optimal cost functions are shown in Figures 2 and 3. For comparison, the same figures also show the exact solution and optimal cost, computed by gridding the state space. It is noticed that in some part of the set $\mathbb{X} = [-0.9, 0.9]^2$ a feasible solution does not exist. Figure 4 shows simulation results from the initial state $x(0) = (-0.683, -0.874)^T$ with the system in closed loop with both the exact and approximate RHC (in compliance with Fig. 2 in (Chen and Allgöwer 1998)). We notice that the difference is small, as the exact and approximate curves can hardly be distinguished in the plots. The computational complexity with the approximate approach is at most 14 arithmetic operations per sample, since there are at most 10 levels in the search tree. In contrast, (Chen and Allgöwer 1998) report typical computing times of about 6 seconds per sample using a state-of-the-art numerical optimization (NAG, e04ucf), which are several orders of magnitude slower. It is also worthwhile mentioning that the computations with the present approach can be carried out with sufficient accuracy using fixed-point arithmetic, while floating-point arithmetic is generally required in iterative numerical optimization.

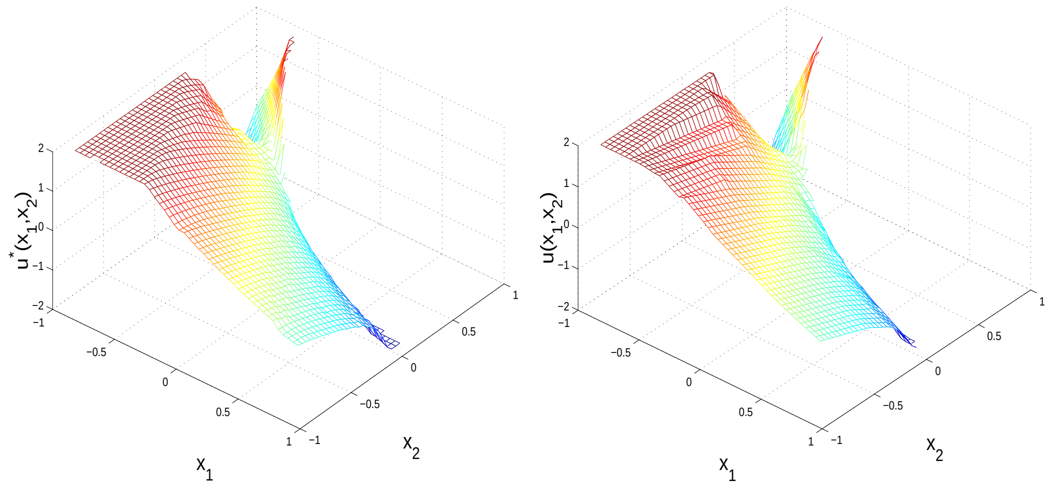

Fig. 2. Optimal solution $u(x)$, exact to the left and approximate to the right.

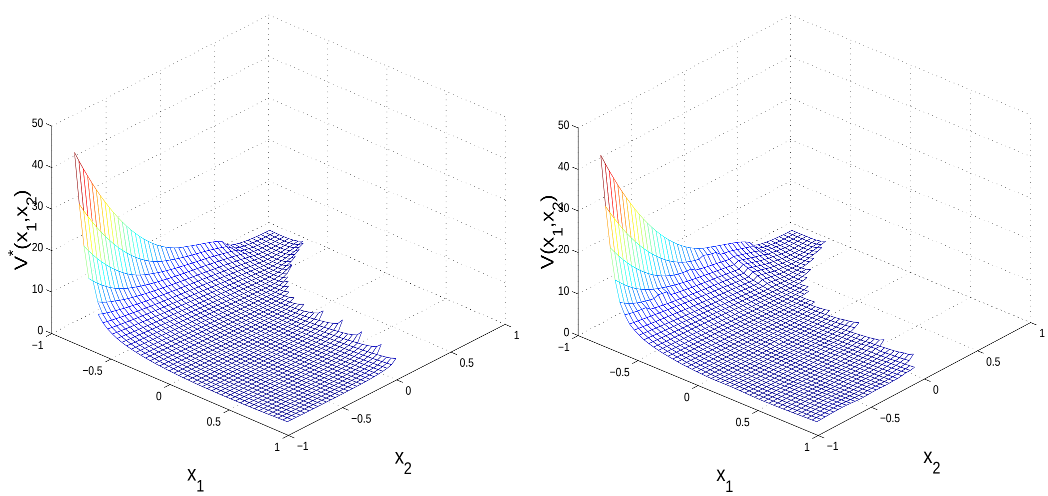

Fig. 3. Optimal cost function, exact $V^*(x)$ to the left and approximate $\hat{V}(x)$ to the right.

The partition found by Algorithm 1 can be further simplified without any loss of accuracy. For example, in all the regions in the lower left corner of the partition in Figure 1, the control is saturated at $u = 2$ as shown in Figure 2. This unnecessary partitioning is a consequence of the cost function approximation approach is taken. The solutions in these regions differ towards the end of the control trajectory and, hence, their cost differ. Still, it is straightforward to join neighboring regions with the same solution at the first sample of the control trajectory in a postprocessing step to reduce the complexity of the partition.

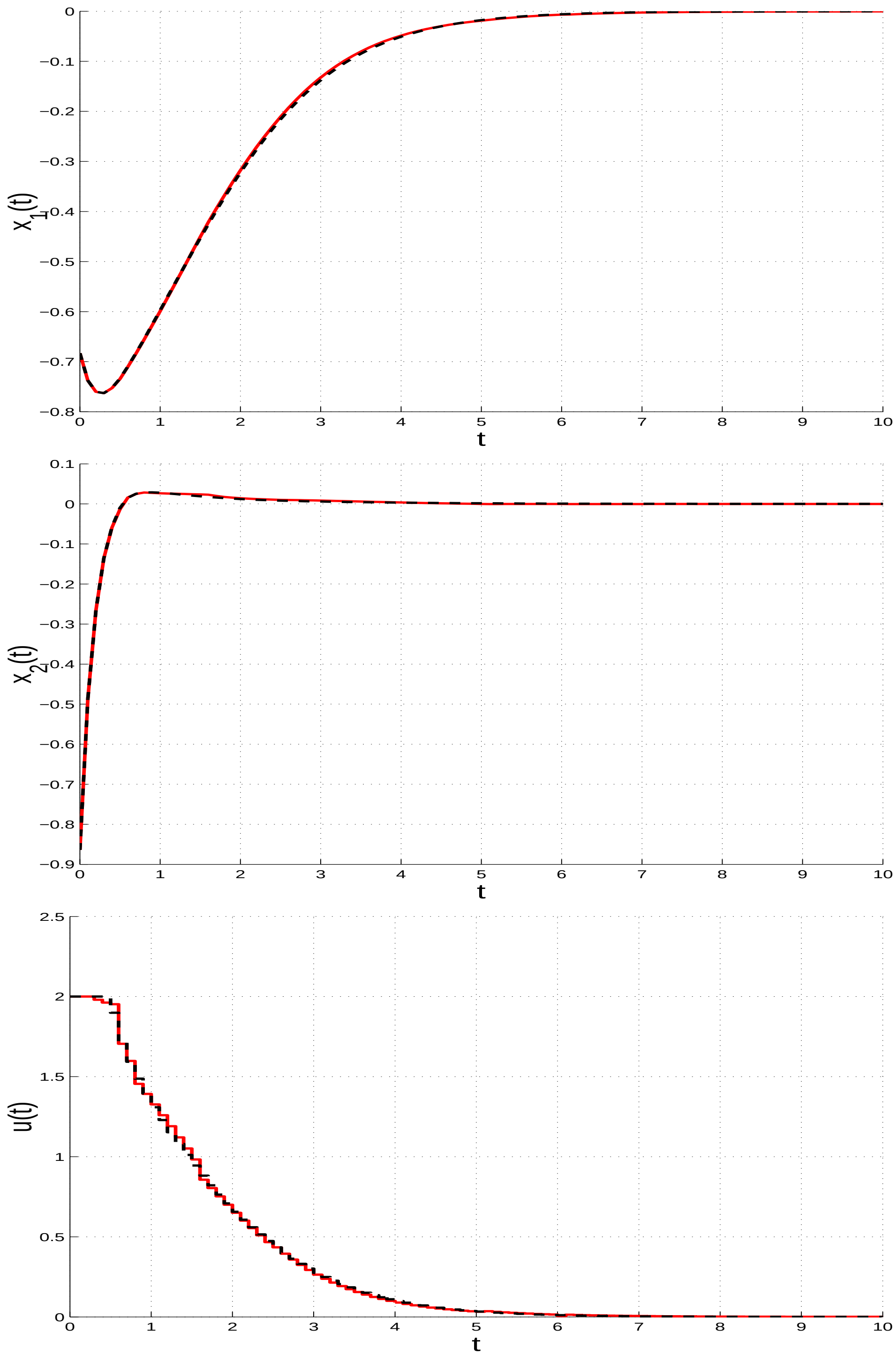

Fig. 4. Simulation results, where the exact solution are the dashed curves and the approximate solution are the solid curves.

7 Conclusions

An offline algorithm for the construction of a PWL explicit state feedback approximation to a general constrained nonlinear receding horizon control problem is given. It allows very efficient real-time implementation via a binary tree search, without real-time optimization. It is thus ideally suited for small embedded applications with low-cost (with only fixed-point arithmetics) real-time computers still operating at high sampling rates. Another important benefit of the explicit solution is that its simple implementation reduces software complexity, which is a key issue in safety-critical applications where real-time optimization is usually avoided.

A Background on parametric programming

For a given $x_0 \in X_F$ the well known Karush-Kuhn-Tucker (KKT) first-order conditions (Nocedal and Wright 1999)

$$\nabla_U L(U_0; x_0) = 0 \tag{A.1}$$
$$\text{diag}(\lambda_0) G(U_0; x_0) = 0 \tag{A.2}$$
$$\lambda_0 \geq 0 \tag{A.3}$$
$$G(U_0; x_0) \leq 0 \tag{A.4}$$

are necessary for a local minimum U_0, with associated Lagrange multiplier λ_0 and the Lagrangian defined as

$$L(U, \lambda; x) \triangleq J(U; x) + \lambda^T G(U; x) \tag{A.5}$$

Consider the optimal active set $\mathcal{A}_0$ at x_0, i.e. the set of indices to active constraints in (A.4). The above conditions are sufficient provided the following second order condition holds:

$$z^T \nabla^2_{UU} L(U_0, \lambda_0; x_0) z > 0, \quad \text{for all } z \in \mathcal{F} - \{0\} \tag{A.6}$$

with $\mathcal{F}$ being the set of all directions where it is not clear from first order conditions if the cost will increase or decrease:

$$\begin{aligned} \mathcal{F} = \{ z \in \mathbb{R}^p \mid & \nabla_U G_{\mathcal{A}_0}(U_0; x_0) z \geq 0, \\ & \nabla_U G_i(U_0; x_0) z = 0, \text{ for all } i \text{ with } (\lambda_0)_i > 0 \} \quad . \end{aligned} \tag{A.7}$$

The notation $G_{\mathcal{A}_0}$ means the rows of G with indices in $\mathcal{A}_0$. The following result gives local regularity conditions for the optimal solution, Lagrange multipliers and optimal cost as functions of x.

Theorem 5 *Consider the problem (6), and let $x_0 \in X_F$ and U_0 be given. If*

(1) V and G are twice continuously differentiable in a neighborhood of (U_0, x_0).
(2) The sufficient conditions (A.1)-(A.4) and (A.6) for a local minimum at U_0 hold.
(3) Linear independence constraint qualification (LICQ) holds, i.e. active constraint gradients $\nabla_U G_{\mathcal{A}_0}(U_0; x_0)$ are linearly independent.
(4) Strict complementary slackness holds, i.e. $(\lambda_0)_{\mathcal{A}_0} > 0$.

then

(1) U_0 is a local isolated minimum,
(2) For x in a neighborhood of x_0, there exists a unique continuous function $U^*(x)$ satisfying $U^*(x) = U_0$ and the sufficient conditions for a local minimum.
(3) Assume in addition A4 holds, and let x be in a neighborhood of x_0. Then $U^*(x)$ is differentiable and the associated Lagrange multipliers $\lambda^*(x)$ exists, and are unique and continuously differentiable. Finally, the set of active constraints is unchanged, and the active constraint gradients are linearly independent at $U^*(x)$.

□

Parts 1 and 2 are due to (Kojima 1980), while part 3 is due to Theorem 3.2.2 in (Fiacco 1983). For the fixed active set $\mathcal{A}_0$ the KKT conditions (A.1)-(A.2) reduces to the following system of equations parameterized by x:

$$\nabla_U J(U(x); x) + \sum_{i \in \mathcal{A}_0} \lambda_i(x) \nabla_U G_i(U(x); x) = 0 \tag{A.8}$$

$$G_{\mathcal{A}_0}(U(x); x) = 0 \tag{A.9}$$

The functions $U(x)$ and $\lambda(x)$ implicitly defined by (A.8)-(A.9) are optimal only for those x where the active set $\mathcal{A}_0$ is optimal. Assuming λ and U are well defined on X, we characterize the critical region $\mathcal{X}_{\mathcal{A}_0}$ where the solution corresponding to the fixed active set $\mathcal{A}_0$ is optimal:

$$\mathcal{X}_{\mathcal{A}_0} \triangleq \{x \in X \mid \lambda(x) \geq 0,\ G(U(x); x) \leq 0\} \tag{A.10}$$

We remark that in the nonlinear RHC setting, the first two assumption of Theorem 5 are satisfied if the cost function is required to be strictly jointly convex. This implies uniqueness of solutions, which is clearly necessary for continuity. It remains to discuss the third requirement, LICQ. As discussed in (Bemporad *et al.* 2002, Tøndel *et al.* 2001), the LICQ will generally be violated in degenerate critical regions, i.e. critical regions that are not full-dimensional. However, because the critical regions are closed sets this only reflects the fact that the dual solution may not be unique.

Hence, one may exploit the closedness of the critical regions together with the uniqueness of the solution to establish that U^* is continuous, as in (Bemporad *et al.* 2002). For an in-depth discussion of regularity properties of U^*, we refer to (Fiacco 1983).

References

Bemporad, A. and C. Filippi (2001). Suboptimal explicit MPC via approximate quadratic programming. In: *Proc. IEEE Conf. Decision and Control, Orlando*. pp. FrP08–5.

Bemporad, A., F. Borrelli and M. Morari (2000*a*). Optimal controllers for hybrid systems: Stability and piecewise linear explicit form. In: *Proc. Conference on Decision and Control, Sydney*.

Bemporad, A., M. Morari, V. Dua and E. N. Pistikopoulos (2000*b*). The explicit solution of model predictive control via multiparametric quadratic programming. In: *Proc. American Control Conference, Chicago*. pp. 872–876.

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

Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching. *Communications of the ACM* **18**, 509–517.

Bertsekas, D. P. and J. N. Tsitsiklis (1998). *Neuro-dynamic Programming*. Athena Scientific, Belmont.

Chen, H. and F. Allgöwer (1998). A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica* **34**, 1205–1217.

De Nicolao, G., L. Magni and R. Scattolini (1996). On the robustness of receding horizon control with terminal constraints. *IEEE Trans. Automatic Control* **41**, 451–453.

Fiacco, A. V. (1983). *Introduction to sensitivity and stability analysis in nonlinear programming*. Orlando, Fl: Academic Press.

Grancharova, A. and T. A. Johansen (2002). Approximate explicit model predictive control incorporating heuristics. In: *Proc. IEEE Conf. Computer Aided Control Design, Glasgow. http://www.itk.ntnu.no/ansatte/Johansen_Tor.Arne/cacsd_2002_1027.pdf*.

Horst, R. and H. Tuy (1993). *Global Optimization*. Springer-Verlag, Berlin.

Jadbabaie, A., J. Yu and J. Hauser (2001). Unconstrained receding-horizon control of nonlinear systems. *IEEE Trans. Automatic Control* **46**, 776–783.

Johansen, T. A. (2002). On multi-parametric nonlinear programming and explicit nonlinear model predictive control. Preprint, submitted for publication.

Johansen, T. A. and A. Grancharova (2002*a*). Approximate explicit model predictive control via orthogonal search tree. submitted for publication, http://www.itk.ntnu.no/ansatte/Johansen_Tor.Arne/mpqp-tree3.pdf.

Johansen, T. A. and A. Grancharova (2002*b*). Approximate explicit model predictive control implemented via orthogonal search tree partitioning. In: *Preprints, IFAC World Congress, Barcelona*. http://www.itk.ntnu.no/ansatte/Johansen_Tor.Arne/mpqp-tree-ifac.pdf.

Keerthi, S. S. and E. G. Gilbert (1988). Optimal infinite horizon feedback laws for a general class of constrained discrete-time systems: Stability and moving horizon approximations. *J. Optimization Theory and Applications* **57**, 265–293.

Kerrigan, E. and J. M. Maciejowski (2000). Invariant sets for constrained nonlinear discrete-time systems with application to feasibility in model predictive control. In: *Proc. IEEE Conf. Decision and Control, Sydney*.

Kojima, M. (1980). Strongly stable stationary solutions in nonlinear programs. In: *Analysis and Computation of Fixed Points* (S. M. Robinson, Ed.). pp. 93–138. Academic Press, New York.

Mangasarian, O. L. and J. B. Rosen (1964). Inequalities for stochastic nonlinear programming problems. *Operations Research* **12**, 143–154.

Mayne, D. Q. and H. Michalska (1990). Receding horizon control of nonlinear systems. *IEEE Trans. Automatic Control* **35**, 814–824.

Mayne, D. Q., J. B. Rawlings, C. V. Rao and P. O. M. Scokaert (2000). Constrained model predictive control: Stability and optimality. *Automatica*.

Michalska, H. and D. Q. Mayne (1993). Robust receding horizon control of constrained nonlinear systems. *IEEE Trans. Automatic Control* **38**, 1623–1633.

Nocedal, J. and S. J. Wright (1999). *Numerical Optimization*. Springer-Verlag, New York.

Parisini, T. and R. Zoppoli (1995). A receding-horizon regulator for nonlinear systems and a neural approximation. *Automatica* **31**, 1443–1451.

Rantzer, A. (1999). Dynamic programming via convex optimization. In: *Preprints IFAC World Congress, Beijing*. Vol. D. pp. 491–496.

Rockafellar, R. T. (1970). *Convex Analysis*. Princeton University Press, New Jersey.

Tøndel, P., T. A. Johansen and A. Bemporad (2001). An algorithm for multi-parametric quadratic programming and explicit MPC solutions. In: *Proc. IEEE Conf. Decision and Control, Orlando*. pp. TuP11–4.