Chapter 3
Explicit NMPC Using mp-QP Approximations of mp-NLP

Abstract. A numerical algorithm for approximate multi-parametric nonlinear programming (mp-NLP) is developed. The algorithm locally approximates the mp-NLP with a multi-parametric quadratic program (mp-QP). This leads to an approximate mp-NLP solution that is composed from the solution of a number of mp-QP solutions. The method allows approximate solutions to nonlinear optimization problems to be computed as explicit piecewise linear functions of the problem parameters. In control applications such as nonlinear constrained model predictive control this allows efficient online implementation in terms of an explicit piecewise linear state feedback without any real-time optimization.

3.1 Introduction

For multi-parametric nonlinear programs (mp-NLPs) one cannot expect to find exact solutions, in general. There is a large body of theory that develops local regularity conditions and local sensitivity results [7, 17], and algorithms for non-local parameter variations are derived for single-parametric problems [12]. Here we describe an approximate mp-NLP algorithm utilizing NLP and mp-QP algorithms to solve local sub-problems, first proposed in [13].

Before we describe the main idea behind the algorithm, we recall that a widely used family of algorithms for the numerical solution of nonlinear programs (NLPs) is Sequential Quadratic Programming (SQP) algorithms, e.g. [20]. They are iterative algorithms, where at each iteration the nonlinear program is locally approximated by a convex quadratic program (QP) at the current candidate solution point. This means that the nonlinear cost function is locally approximated by a positive definite quadratic function, and the nonlinear constraints are locally approximated by linear constraints. The QP is then solved to find a search direction towards a better point, a step in this direction is made, and the procedure is repeated and will eventually converge to a locally optimal solution for the NLP.

In the approximate mp-NLP algorithm described in this chapter, the idea is to locally approximate mp-NLPs with mp-QPs, similar to the use of QPs within SQP.

A. Grancharova & T.A. Johansen: Explicit Nonlinear Model Predictive Ctrl., LNCIS 429, pp. 71–85.
springerlink.com

An iterative (recursive) partitioning of the parameter space is used to control the accuracy of the approximation. It refines the partition in order to improve the accuracy of the local mp-QP approximation in the parts of the parameter space where this is needed in order to meet accuracy specifications in terms of sub-optimality bounds on the cost.

The proposed method is different from the approximate mp-NLP algorithm in Section 1.1.5, and the function approximation methods for non-linear optimal control are described in [1, 21, 22, 4, 23, 15, 2, 19]. While these references approximate the mp-NLP solution based on solution points computed for an extensive number of parameter values using an NLP algorithm, in the present chapter the mp-NLP is approximated by a number of mp-QPs that are solved using the mp-QP algorithm [28]. In [5], several alternative multi-parametric programming algorithms for explicit approximate solution of convex mp-NLP problems are compared, and a modification of the algorithm described in this section was found to be efficient. The main modifications is a different approach for the partitioning outside the mp-QP solutions.

The mp-NLP problem is formulated as follows:

$$\min_{z} V(z,x) \tag{3.1}$$

subject to

$$G(z,x) \leq 0 \tag{3.2}$$

for all $x \in X$, where X is some parameter set. Eqs. (3.1) - (3.2) define an mp-NLP, since it is an NLP in z parameterized by the parameter vector x. Assume the solution exists, and let it be denoted $z^*(x)$. In the special case when V and G are quadratic and linear, respectively, in both z and x, a solution can be found explicitly and exactly as a continuous PWL mapping $z^*(x)$ using mp-QP.

In [13] it is suggested to utilize an mp-QP algorithm to approximately solve the mp-NLP (3.1)-(3.2). In the mp-QP case, this algorithm will iteratively build a polyhedral partition of the state-space with an exact solution corresponding to a fixed active set within each polyhedral critical region. This leads to a PWL solution $z^*(x)$ since a fixed active set leads to a solution that is linear in x, [3]. In the mp-NLP case we keep the PWL structure of the solution, but in each polyhedral region we approximate the (exact) nonlinear solution by a PWL approximate solution found by solving a mp-QP constructed as a locally accurate quadratic approximation to V and linear approximation to G. Under regularity assumptions on V and G, one may expect that the approximation error and constraint violations will be small if each of the regions are sufficiently small. We therefore suggest to analyze the approximation error within each region and introduce a sub-partitioning of some regions when needed in order to keep the approximation error and constraint violations within specified bounds.

3.2 Local mp-QP Approximation to mp-NLP

In this section we study how the cost function and constraints can be locally approximated by mp-QP problems, based on [13]. Let $x_0 \in X$ be arbitrary and denote the corresponding optimal solution $z_0 = z^*(x_0)$. Taylor series expansions of V and G about the point (z_0, x_0) leads to the following locally approximate mp-QP problem:

$$V_0(z,x) \triangleq \frac{1}{2}(z-z_0)^T H_0(z-z_0) + (D_0 + F_0(x-x_0))(z-z_0) + Y_0(x) \quad (3.3)$$

subject to

$$G_0(z-z_0) \leq E_0(x-x_0) + T_0 \quad (3.4)$$

The cost and constraints are defined by the matrices

$$\begin{gathered} H_0 \triangleq \nabla^2_{zz} V(z_0,x_0), \;\; F_0 \triangleq \nabla^2_{xz} V(z_0,x_0) \\ D_0 \triangleq \nabla_z V(z_0,x_0), \;\; G_0 \triangleq \nabla_z G(z_0,x_0) \\ E_0 \triangleq -\nabla_x G(z_0,x_0), \;\; T_0 \triangleq -G(z_0,x_0) \\ Y_0(x) \triangleq V(z_0,x_0) + \nabla_x V(z_0 z x_0)(x-x_0) + \tfrac{1}{2}(x-x_0)^T \nabla^2_{xx} V(z_0,x_0)(x-x_0) \end{gathered}$$

Let the PWL solution to the mp-QP (3.3) - (3.4) be denoted $z_{QP}(x)$ with associated Lagrange multipliers $\lambda_{QP}(x)$. This solution satisfies the following KKT conditions

$$\begin{aligned} H_0\,(z_{QP}(x)-z_0) + F_0(x-x_0) + D_0 + G_0^T \lambda_{QP}(x) &= 0 && (3.5)\\ \mathrm{diag}(\lambda_{QP}(x))\,(G_0(z_{QP}(x)-z_0) - E_0(x-x_0) - T_0) &= 0 && (3.6)\\ \lambda_{QP}(x) &\geq 0 && (3.7)\\ G_0\,(z_{QP}(x)-z_0) - E_0(x-x_0) - T_0 &\leq 0 && (3.8) \end{aligned}$$

Consider the optimal active set $\mathscr{A}$ of the QP (3.3) - (3.4) at a given $x \in X$, and let $G_{0,\mathscr{A}}$ and $\lambda_{QP,\mathscr{A}}$ denote the rows of G_0 and λ_{QP}, respectively, with indices in $\mathscr{A}$. Eqs. (3.5) - (3.6) define the following linear equations

$$\begin{pmatrix} H_0 & G_{0,\mathscr{A}}^T \\ G_{0,\mathscr{A}} & 0 \end{pmatrix} \begin{pmatrix} z_{QP,\mathscr{A}}(x) - z_0 \\ \lambda_{QP,\mathscr{A}}(x) \end{pmatrix} = \begin{pmatrix} F_0(x-x_0) + D_0 \\ E_0(x-x_0) + T_0 \end{pmatrix} \quad (3.9)$$

The following results is an extension of Theorem 2 in [3] (where $H_0 > 0$ was assumed in addition to LICQ).

Assumption 3.1. *V and G are twice continuously differentiable in a neighborhood of (z_0, x_0).*

Assumption 3.2. *The sufficient conditions (1.7)-(1.10) and (1.12) for a local minimum at z_0 hold.*

Assumption 3.3. *Linear independence constraint qualification (LICQ) holds, i.e. the active constraint gradients $\nabla_z G_{\mathscr{A}_0}(z_0, x_0)$ are linearly independent.*

Assumption 3.4. *Strict complementary slackness holds, i.e. $(\lambda_0)_{\mathscr{A}_0} > 0$.*

Assumption 3.5. *For an optimal active set $\mathscr{A}$, the matrix $G_{0,\mathscr{A}}$ has full row rank (LICQ) and $Z_{0,\mathscr{A}}^T H_0 Z_{0,\mathscr{A}} > 0$, where the columns of $Z_{0,\mathscr{A}}$ is a basis for null$(G_{0,\mathscr{A}})$.*

Theorem 3.1. *Consider the problem (3.3)-(3.4), and let X be a polyhedral set with $x_0 \in X$. The system of linear equations (3.9) has a unique solution*

$$\begin{pmatrix} z_{QP,\mathscr{A}}(x) - z_0 \\ \lambda_{QP,\mathscr{A}}(x) \end{pmatrix} = \begin{pmatrix} H_0 & G_{0,\mathscr{A}}^T \\ G_{0,\mathscr{A}} & 0 \end{pmatrix}^{-1} \begin{pmatrix} F_0^T(x - x_0) + D_0 \\ E_0(x - x_0) + T_0 \end{pmatrix} \tag{3.10}$$

and the critical region where the solution is optimal is given by the polyhedral set

$$\mathscr{X}_{0,\mathscr{A}} \triangleq \left\{ x \in X \mid \lambda_{QP,\mathscr{A}}(x) \geq 0, G_0(z_{QP,\mathscr{A}}(x) - z_0) \leq E_0(x - x_0) + T_0 \right\}$$

Hence, $z_{QP}(x) = z_{QP,\mathscr{A}}(x)$ and $\lambda_{QP}(x) = \lambda_{QP,\mathscr{A}}(x)$ if $x \in \mathscr{X}_{0,\mathscr{A}}$, and the solution z_{QP} is a continuous, PWL function of x defined on a polyhedral partition of X.

Proof ([13]). Non-singularity of the matrix on the left-hand-side of (3.9) follows from standard 2nd order considerations such as Lemma 16.1 in [20], due to Assumption 3.5. The rest of the proof is similar to [3]. □

Algorithms for solving such an mp-QP (with straightforward modifications to account for the relaxed second-order condition of Assumption 3.5) are given in Section 1.2. The following result compares the primal and dual local QP solution with the global NLP solution.

Theorem 3.2. *Consider the problem (3.1)-(3.2). Let $x_0 \in X$ and suppose there exists a z_0 satisfying the above assumptions. Then for x in a neighborhood of x_0*

$$z_{QP}(x) - z^*(x) = \mathscr{O}(||x - x_0||_2^2) \tag{3.11}$$

$$\lambda_{QP}(x) - \lambda^*(x) = \mathscr{O}(||x - x_0||_2^2) \tag{3.12}$$

Proof ([13]). Let the neighborhood of x_0 under consideration be restricted to $\mathscr{X}_{0,\mathscr{A}_0}$, where $\mathscr{A}_0$ is the optimal active set at x_0. This is without loss of generality since the assumptions imply that x_0 is an interior point in $\mathscr{X}_{0,\mathscr{A}_0}$. The first KKT condition for the QP is

$$H_0 \left(z_{QP}(x) - z_0 \right) + F_0(x - x_0) + \left(D_0 + G_0^T \lambda_{QP}(x) \right) = 0 \tag{3.13}$$

Since $z_0 = z^*(x_0)$ we have $z^*(x) - z_0 = \mathscr{O}(||x - x_0||_2)$, and the first KKT condition (1.7) for the NLP can be rewritten as follows using a Taylor series expansion

$$
\begin{aligned}
0 &= \nabla_z V(z^*(x),x) + \nabla_z^T G(z^*(x),x)\lambda^*(x) && (3.14)\\
&= \nabla_z V(z_0,x_0) + \nabla_{zz}^2 V(z_0,x_0)(z^*(x)-z_0)\\
&\quad + \nabla_{xz}^2 V(z_0,x_0)(x-x_0)\\
&\quad + \left(\nabla_z^T G(z_0,x_0) + \mathscr{O}(||x-x_0||_2)\right)\lambda^*(x) + \mathscr{O}(||x-x_0||_2^2) && (3.15)\\
&= D_0 + H_0(z^*(x)-z_0) + F_0(x-x_0) + G_0^T \lambda_{QP}(x)\\
&\quad + G_0^T(\lambda^*(x)-\lambda_{QP}(x)) + \mathscr{O}(||x-x_0||_2^2)\\
&\quad + \mathscr{O}(||x-x_0||_2)(\lambda^*(x)-\lambda_{QP}(x)) && (3.16)
\end{aligned}
$$

Comparing (3.13) and (3.16) we get

$$H_0\left(z_{QP}(x)-z^*(x)\right) + G_0^T\left(\lambda_{QP}(x)-\lambda^*(x)\right) = \mathscr{O}(||x-x_0||_2^2) \tag{3.17}$$

From Theorem 1.1, part 3, it is known that the set of active constraints is unchanged in a neighborhood of x_0. Hence, for the QP we have

$$G_0\left(z_{QP}(x)-z_0\right) = E_0(x-x_0) + T_0 \tag{3.18}$$

When x is in a neighborhood of x_0, Taylor expanding the NLP constraints gives

$$
\begin{aligned}
0 &= G(z^*(x),x) && (3.19)\\
&= G(z_0,x_0) + \nabla_z G(z_0,x_0)(z^*(x)-z_0) + \nabla_x G(z_0,x_0)(x-x_0) + \mathscr{O}(||x-x_0||_2^2)\\
&= G_0(z^*(x)-z_0) - E_0(x-x_0) - T_0 + \mathscr{O}(||x-x_0||_2^2)
\end{aligned}
$$

Comparing (3.18) and (3.19) it follows that

$$G_0\left(z_{QP}(x)-z^*(x)\right) = \mathscr{O}(||x-x_0||^2) \tag{3.20}$$

and the result follows by inverting the system (3.17) and (3.20). This system is indeed invertible: Due to Assumption 3.4 it follows that $\nabla_z G_{\mathscr{A}_0}(z_0,x_0)\zeta = 0$ for all $\zeta \in \mathscr{F}$. Since $G_{0,\mathscr{A}_0} = \nabla_z G_{\mathscr{A}_0}(z_0,x_0)$, it is clear that $\mathscr{F} = \mathrm{null}(G_{0,\mathscr{A}_0})$ and Assumptions 3.2 and 3.3 (and in particular eq. (1.12)) ensures that Assumption 3.5 holds and non-singularity of

$$\begin{pmatrix} H_0 & G_0^T \\ G_0 & 0 \end{pmatrix}$$

follows from Lemma 16.1 in [20]. □

Theorem 3.2 concerns only a small neighborhood of x_0 and is therefore of limited computational use. However, it provides a qualitative indication that the mp-QP approximation of the mp-NLP is locally accurate, under some assumptions. We therefore proceed by deriving some quantitative estimates and bounds on the cost and solution errors, as well as the maximum constraint violation. The solution error bound is defined as

$$\rho \triangleq \max_{x \in X_0} |w^T (\mu(0, z_{QP}(x)) - \mu(0, z^*(x)))| \tag{3.21}$$

where $X_0 \subset X$ is arbitrary, and w is a vector with positive weights. Likewise, we define the cost error bound

$$\varepsilon \triangleq \max_{x \in X_0} |V(z_{QP}(x), x) - V^*(x)| \tag{3.22}$$

where $V^*(x) \triangleq V(z^*(x); x)$. In addition, one may compute the maximum constraint violation

$$\delta \triangleq \max_{x \in X_0} \omega^T G(z_{QP}(x), x) \tag{3.23}$$

where ω is a vector of non-negative weights. Typically, the elements of w corresponding to the first sample of the trajectory will be positive, while the remaining will be zero since in receding horizon control the primary interest is the first sample of the trajectory. The maximum constraint violation (3.23) can be computed by solving an NLP, while the solution and cost error bounds (3.21) and (3.22) are not easily computed without introducing additional assumptions or allowing underestimation. A further problem is that they require computation of the exact $z^*(x)$ for several x, which relies on the solution of several NLPs and is therefore expensive. Obvious estimation techniques for ρ and ε is to take the maximum over a finite number of points X_0, such as extreme points (vertices), points generated by Monte Carlo methods, or combinations. It should be emphasized that these methods can underestimate the bounds, in general.

3.3 Convexity

For the case when V and G are convex functions, it is possible to derive a guaranteed bound on ε from knowledge of $z^*(x)$ only at all the vertices $\mathscr{V} = \{v_1, v_2, ..., v_M\}$ of the bounded polyhedron X_0, see section 1.1.5.1. This immediately gives the following bounds on the cost function error $-\varepsilon_1 \leq V^*(x) - V(z_{QP}(x), x) \leq \varepsilon_2$, where

$$\varepsilon_1 = \max_{x \in X_0} (V(z_{QP}(x), x) - \underline{V}(x)) \tag{3.24}$$

$$\varepsilon_2 = \max_{x \in X_0} \left(\overline{V}(x) - V(z_{QP}(x), x)\right) \tag{3.25}$$

Hence, the cost error bound $\tilde{\varepsilon} \triangleq \max(\varepsilon_1, \varepsilon_2) \geq \varepsilon$ can be computed by solving two NLPs. A solution error bound can be shown to exist as in Chapter 9.7 of [7].

3.4 Algorithm

So far it has been established that under some regularity conditions, local mp-QP solutions give accurate approximation to the mp-NLP solution when restricted to a sufficiently small subset $X_0 \subset X$. It remains to determine a sub-partition of the polyhedral region X such that the local mp-QP solutions associated with each region are sufficiently accurate. In [13] the following algorithm was suggested to approximate the mp-NLP solution, based on recursive sub-partitioning guided by the approximation errors discussed above.

Algorithm 3.1. Approximate mp-NLP.

Step 1. Let $X_0 := X$.

Step 2. Select x_0 as the Chebychev center of X_0, by solving an LP.

Step 3. Compute $z_0 = z^*(x_0)$ by solving the NLP (3.1)-(3.2) with $x(0) = x_0$.

Step 4. Compute the local mp-QP problem (3.3) - (3.4) at (z_0, x_0). If H_0 is not positive definite, then modify H_0 such that it is positive definite (e.g. by an eigenvalue decomposition where negative eigenvalues are replaced by small positive numbers) and the mp-QP is convex.

Step 5. Estimate the approximation errors ε, ρ and δ on X_0.

Step 6. If $\varepsilon > \overline{\varepsilon}$, $\rho > \overline{\rho}$, or $\delta > \overline{\delta}$, then sub-partition X_0 into polyhedral regions using the heuristic rules described in Section 1.1.5.2.

Step 7. Select a new X_0 from the partition. If no further sub-partitioning is needed, go to step 8. Otherwise, repeat Steps 2-7 until the tolerances $\overline{\varepsilon}$, $\overline{\rho}$ and $\overline{\delta}$ are respected in all polyhedral regions in the partition of X.

Step 8. For all sub-partitions X_0, solve the mp-QP (3.3) - (3.4) using the mp-QP solver [28, 26].

Computation of the approximation errors in Step 5 are carried out based on the results in Section 3.3 if the cost function and constraints are known to be convex. If not, we suggest to estimate error bounds by solving NLPs at a number of points in X_0, typically the vertices and possibly other points, as in [10]. If the convexity assumption does not hold, this seems to be a fairly robust strategy. The sub-partitioning in Step 6 is based on heuristic criteria, where the purpose is to select one axis-orthogonal hyperplane to split X_0 such that the approximation error after splitting is minimized (as described in Section 1.1.5.2). Alternatively, the hyperplane is selected such that the change of error at the vertices (before splitting) across the hyperplane is maximal (as used in [9]).

3.5 Example: Compressor Surge Control

3.5.1 NMPC Formulation

Consider the following 2nd-order compressor model [11, 8] with x_1 being normalized mass flow, x_2 normalized pressure and u normalized mass flow through a close coupled valve in series with the compressor

$$\dot{x}_1 = B\left(\Psi_e(x_1) - x_2 - u\right) \tag{3.26}$$

$$\dot{x}_2 = \frac{1}{B}\left(x_1 - \Phi(x_2)\right) \tag{3.27}$$

The following compressor and valve characteristics are used

$$\Psi_e(x_1) = \psi_{c0} + H\left(1 + 1.5\left(\frac{x_1}{W} - 1\right) - 0.5\left(\frac{x_1}{W} - 1\right)^3\right)$$
$$\Phi(x_2) = \gamma \operatorname{sign}(x_2)\sqrt{|x_2|}$$

with $\gamma = 0.5$, $B = 1$, $H = 0.18$, $\psi_{c0} = 0.3$ and $W = 0.25$. The control objective is to avoid surge, i.e. stabilize the system. This may be formulated as

$$J(u[0,T], x[0,T]) \triangleq \int_0^T l(x(t), u(t), t)dt + S(x(T), T) + Rv^2 \tag{3.28}$$

where

$$l(x,u) = \alpha(x - x^*)^T(x - x^*) + \kappa u^2$$
$$S(x) = \beta(x - x^*)^T(x - x^*)$$

with $\alpha, \beta, \kappa, \rho \geq 0$ and the setpoint $x_1^* = 0.40$, $x_2^* = 0.60$ corresponds to an unstable equilibrium point, subject to the inequality constraints for $t \in [0, T]$

$$u_{min} \leq u(t) \leq u_{max} \tag{3.29}$$

$$-x_2 + 0.4 \leq v \tag{3.30}$$

$$-v \leq 0 \tag{3.31}$$

and the ordinary differential equation (ODE) given by

$$\frac{d}{dt}x(t) = f(x(t), u(t)) \tag{3.32}$$

with given initial condition $x(0) \in X \subset \mathbb{R}^n$. Valve capacity requires the constraint $0 \leq u(t) \leq 0.3$ to hold, and the pressure constraint $x_2 \geq 0.4 - v$ avoids operation too far left of the operating point. The variable $v \geq 0$ is a slack variable introduced in order to avoid infeasibility and $R = 8$ is a large weight. The input signal $u[0,T]$ is assumed to be piecewise constant and parameterized by a vector $U \in \mathbb{R}^p$ such that $u(t) = \mu(t, U) \in \mathbb{R}^r$ is piecewise constant. The solution to (3.32) is assumed in the form $x(t) = \phi(t, U, x(0))$ for $t \in [0, T]$ and some piecewise continuous function ϕ. Relaxing the inequality constraints (3.30) to hold only at N time instants $\{t_1, t_2, ..., t_N\} \subset [0, T]$, we can rewrite the optimization problem in the following standard parametric form (direct single shooting, Section 2.2.2.1) where the ODE constraint (3.32) has been eliminated by substituting its solution ϕ into the cost and constraints; minimize with respect to $z = (U, v)$ the cost

$$V(z,x(0)) \triangleq \int_0^T l(\phi(t,U,x(0)),\mu(t,U),t)dt + S(\phi(T,U,x(0)),T) + Rv^2 \quad (3.33)$$

subject to

$$G(z,x(0)) \triangleq \begin{pmatrix} \tilde{G}(U;x(0)) \\ U - U_{max} \\ U_{min} - U \\ -v \end{pmatrix} \leq 0 \quad (3.34)$$

with blocks $\tilde{G}_i(U;x(0)) \triangleq \hat{g}(\phi(t_i,U,x(0)),\mu(t_i,U))$ as defined by (3.30). Eqs. (3.33) - (3.34) define an mp-NLP, since it is an NLP in z parameterized by the initial state vector $x(0)$.

3.5.2 Tuning and Settings

We have chosen $\alpha = 1$, $\beta = 0$, and $\kappa = 0.08$. The horizon is chosen as $T = 12$, which is split into $N = p = 15$ equal-sized intervals, leading to a piecewise constant control input parameterization. Numerical analysis of the cost function shows that it is non-convex. It should be remarked that the constraints on u and v are linear, such that any mp-QP solution is feasible for the mp-NLP. The bounds ε and ρ are estimates by computing the errors at the vertices only, and the tolerances $\overline{\varepsilon} = 0.5$ and $\overline{\rho} = 0.03$ were applied.

3.5.3 Results

The mp-NLP contains 16 free variables, 47 constraints and 2 parameters. The partition contains 379 regions, resulting from 45 mp-QPs, cf. Fig. 3.1. This can be reduced to 101 polyhedral regions without loss of accuracy in a postprocessing step, where regions with the same solution at the first sample are joined whenever their union remains polyhedral, as in [3]. The computed approximate PWL feedback is shown in Fig. 3.2, together with the exact feedback computed by solving the NLP on a dense grid. The corresponding optimal costs are shown in Fig. 3.3, and simulation results are shown in Fig. 3.4, where the controller is switched on after $t = 20$. We note that it quickly stabilizes the deep surge oscillations. Euler integration with step size 0.02 is applied to solve the ODE.

By generating a search tree using the method of [27], the PWL mapping with 379 regions can be represented as a binary search tree with 329 nodes, of depth 9. Real-time evaluation of the controller therefore requires 49 arithmetic operations, in the worst case, and 1367 numbers needs to be stored in real-time computer memory.

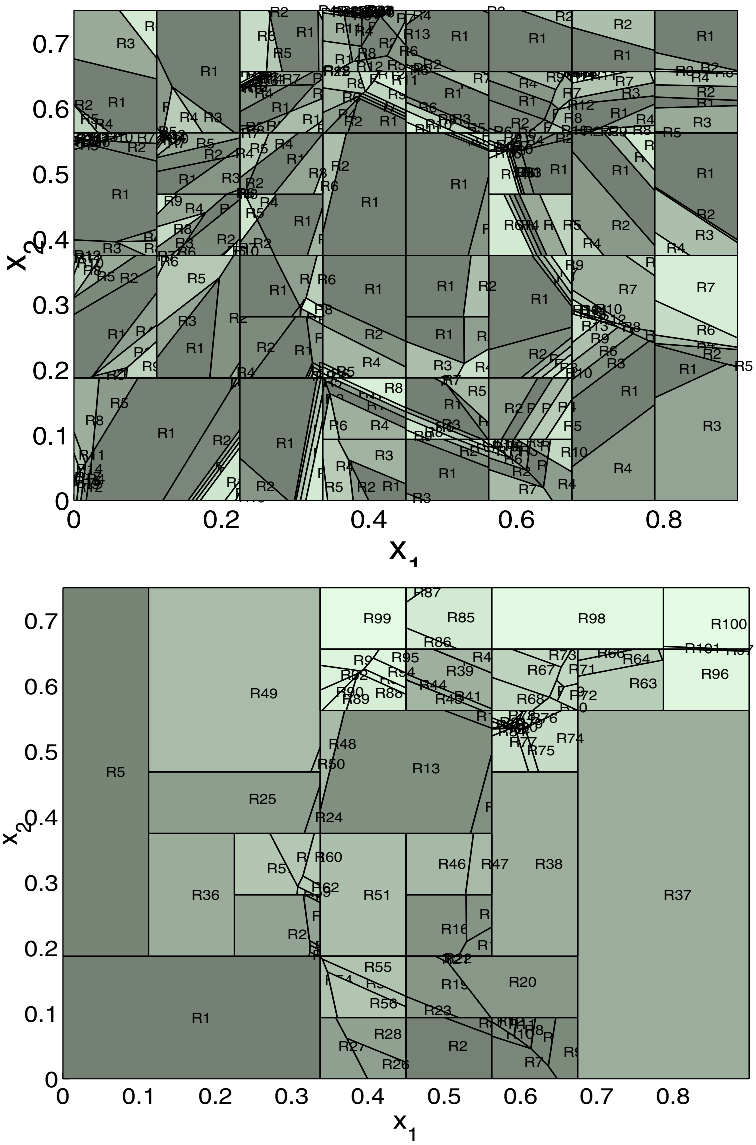

Fig. 3.1 State space partition (top), and after reduction (bottom).

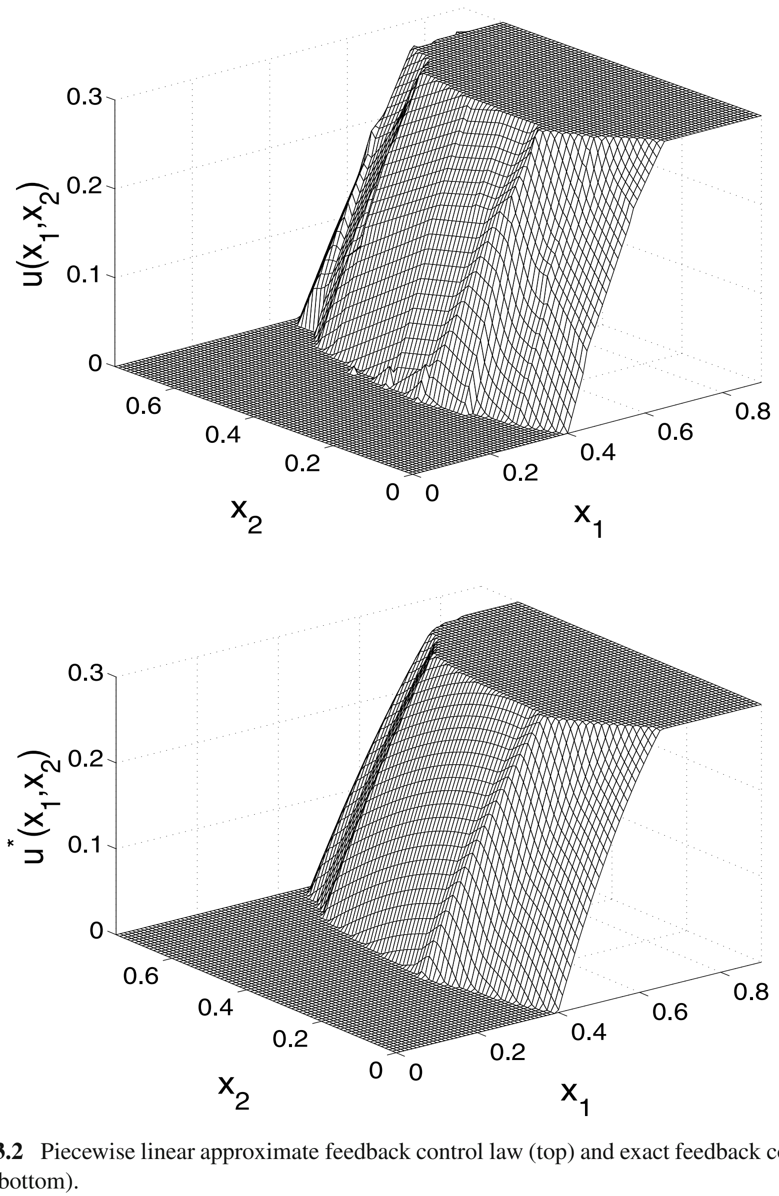

Fig. 3.2 Piecewise linear approximate feedback control law (top) and exact feedback control law (bottom).

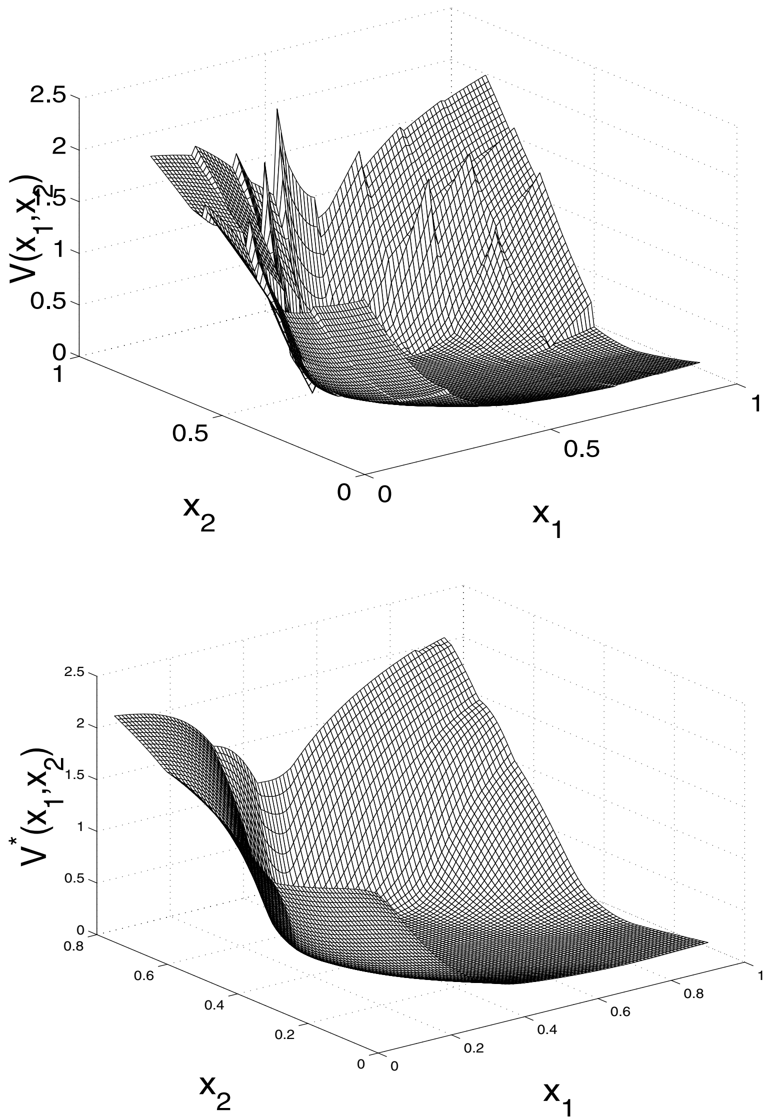

Fig. 3.3 Optimal costs of the approximate feedback control law (top) and exact feedback control law (bottom).

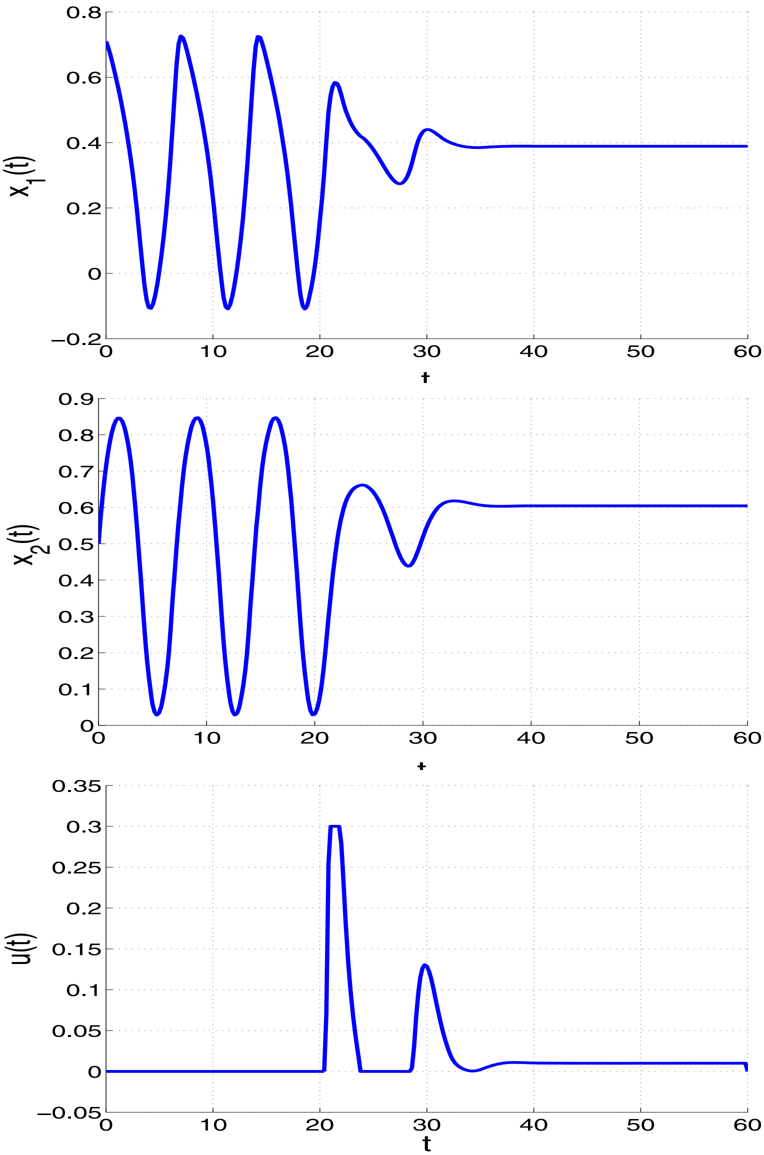

Fig. 3.4 Simulation of compressor with approximate explicit nonlinear MPC. The solution with the exact explicit MPC cannot be distinguished graphically.

References

1. Bemporad, A., Filippi, C.: Suboptimal explicit RHC via approximate multiparametric quadratic programming. J. Optimization Theory and Applications 117, 9–38 (2003)
2. Bemporad, A., Filippi, C.: An algorithm for approximate multiparametric convex programming. Computational Optimization and Applications 35, 87–108 (2006)
3. Bemporad, A., Morari, M., Dua, V., Pistikopoulos, E.N.: The explicit linear quadratic regulator for constrained systems. Automatica 38, 3–20 (2002)
4. Bertsekas, D.P., Tsitsiklis, J.N.: Neuro-dynamic programming. Athena Scientific, Belmont (1998)
5. Domínguez, L.F., Narciso, D.A., Pistikopoulos, E.N.: Recent advances in multiparametric nonlinear programming. Computers and Chemical Engineering 34, 707–716 (2010)
6. Domínguez, L.F., Pistikopoulos, E.N.: Quadratic approximation algorithm for multiparametric nonlinear programming problems. Technical report. Imperial College London (2009)
7. Fiacco, A.V.: Introduction to sensitivity and stability analysis in nonlinear programming. Academic Press, Orlando (1983)
8. Gravdahl, J.T., Egeland, O.: Compressor surge control using a close-coupled valve and backstepping. In: Proceedings of the American Control Conference, Albuquerque, NM, vol. 2, pp. 982–986 (1997)
9. Grancharova, A., Johansen, T.A.: Approximate explicit model predictive control incorporating heuristics. In: Proceedings of IEEE International Symposium on Computer Aided Control System Design, Glasgow, Scotland, U.K., pp. 92–97 (2002)
10. Grancharova, A., Johansen, T.A., Tøndel, P.: Computational aspects of approximate explicit nonlinear model predictive control. In: Findeisen, R., Allgöwer, F., Biegler, L. (eds.) Assessment and Future Directions of Nonlinear Model Predictive Control. LNCIS, vol. 358, pp. 181–192. Springer, Heidelberg (2007)
11. Greitzer, E.M.: Surge and rotating stall in axial flow compressors, part i: Theoretical compression system model. J. Engineering for Power 98, 190–198 (1976)
12. Guddat, J., Guerra Vazquez, F., Jongen, H.T.: Parametric optimization: Singularities, pathfollowing and jumps. Wiley (1990)
13. Johansen, T.A.: On multi-parametric nonlinear programming and explicit nonlinear model predictive control. In: Proceedings of the IEEE Conference on Decision and Control, Las Vegas, NV, vol. 3, pp. 2768–2773 (2002)
14. Johansen, T.A., Grancharova, A.: Approximate explicit constrained linear model predictive control via orthogonal search tree. IEEE Trans. Automatic Control 48, 810–815 (2003)
15. Johansen, T.A.: Approximate explicit receding horizon control of constrained nonlinear systems. Automatica 40, 293–300 (2004)
16. Kojima, M.: Strongly stable stationary solutions in nonlinear programs. In: Robinson, S.M. (ed.) Analysis and Computation of Fixed Points, pp. 93–138. Academic Press, London (1980)
17. Levitin, E.S.: Perturbation theory in mathematical programming. Wiley (1994)
18. Mangasarian, O.L., Rosen, J.B.: Inequalities for stochastic nonlinear programming problems. Operations Research 12, 143–154 (1964)
19. Narciso, D.: Developments in nonlinear multiparametric programming and control. PhD thesis. London, U.K. (2009)
20. Nocedal, J., Wright, S.J.: Numerical optimization. Springer, New York (1999)
21. Parisini, T., Zoppoli, R.: A receding-horizon regulator for nonlinear systems and a neural approximation. Automatica 31, 1443–1451 (1995)

22. Parisini, T., Zoppoli, R.: Neural approximations for multistage optimal control of non-linear stochastic systems. IEEE Trans. on Automatic Control 41, 889–895 (1996)
23. Pistikopoulos, E.N., Georgiadis, M.C., Dua, V.: Multi-parametric programming: Theory, algorithms, and applications. Wiley-VCH (2007)
24. Ralph, D., Dempe, S.: Directional derivatives of the solution of a parametric nonlinear program. Mathematical Programming 70, 159–172 (1995)
25. Rockafellar, R.T.: Convex analysis. Princeton University Press, New Jersey (1970)
26. Spjøtvold, J., Kerrigan, E.C., Jones, C.N., Tøndel, P., Johansen, T.A.: On the facet-to-facet property of solutions to convex parametric quadratic programs. Automatica 42, 2204–2209 (2006)
27. Tøndel, P., Johansen, T.A., Bemporad, A.: Evaluation of piecewise affine control via binary search tree. Automatica 39, 743–749 (2003)
28. Tøndel, P., Johansen, T.A., Bemporad, A.: An algorithm for multi-parametric quadratic programming and explicit MPC solutions. Automatica 39, 489–497 (2003)