Nominal Stability of the Real-Time Iteration Scheme for Nonlinear Model Predictive Control

Moritz Diehl † ⋆ Rolf Findeisen ‡
Frank Allgöwer ‡ Hans Georg Bock † Johannes Schlöder †

†Interdisciplinary Center for Scientific Computing (IWR), University of Heidelberg, Germany
{moritz.diehl,bock,schloeder}@iwr.uni-heidelberg.de
‡Institute for Systems Theory in Engineering, University of Stuttgart, Germany,
{findeisen,allgower}@ist.uni-stuttgart.de
⋆Visitor at the Institute for Mathematics and its Applications, University of Minnesota, USA

February 28, 2003

Abstract

We present and investigate a Newton type method for online optimization in nonlinear model predictive control, the so called "real-time iteration scheme". In this scheme only one Newton type iteration is performed per sampling instant, and the control of the system and the solution of the optimal control problem are performed in parallel. In the resulting combined dynamics of system and optimizer, the actual feedback control in each step is based on the current solution estimate, and the solution estimates are at each sampling instant refined and transferred to the next optimization problem by a specially designed transition. This approach yields an efficient online optimization algorithm that has already been successfully tested in several applications. Due to the close dovetailing of system and optimizer dynamics, however, stability of the closed-loop system is not implied by standard nonlinear model predictive control results. In this paper, we give a proof of nominal stability of the scheme which builds on concepts from both, NMPC stability theory and convergence analysis of Newton type methods. The principal result is that – under some reasonable assumptions – the combined system-optimizer dynamics can be guaranteed to converge towards the origin from significantly disturbed system-optimizer states.

1 Introduction

Nonlinear model predictive control (NMPC) is a feedback control technique that is based on the real-time optimization of a nonlinear dynamic process model. It has attracted increasing attention over the past decade, in particular in chemical engineering [QB01, Hen98, MRRS00]. Among the advantages of NMPC are the flexibility provided in formulating the objective and the process model and the capability to directly handle equality and inequality constraints on states and inputs.

One important precondition for the application of NMPC, however, is the availability of reliable and efficient numerical dynamic optimization algorithms, since at every sampling time a nonlinear dynamic optimization problem must be solved. Solving such an optimization problem efficiently and fast, however, is not a trivial task and has attracted strong research interest in recent years (see e.g. [Wri96, BWB00, TR01, Bie00, LB89, OB95b, TWR02, MBF02]).

Most approaches use classical off-line dynamic optimization algorithms to solve the optimization problems arising in NMPC. They do this as fast as possible, and once the solution has been computed, the obtained control is applied to the system to be controlled. If the system is slow and the computer fast, the feedback delay due to the computation time is short compared to the timescale of the system, and classical stability theory for NMPC [MM90, ABQ+99, DMS00] can be assumed to hold true. In practical applications, however, in particular for large-scale systems, the optimizer cannot be assumed to be infinitely fast compared to the system. A possible approach to take account of the computation time is to predict the state at the time we expect the optimization to be finished and carry out the optimization for this prediction [FA03, CBO00], allowing to prove nominal stability; however, this approach may still result in a considerable delay of the feedback response to disturbances.

In contrast to the classical approaches, the "real-time iteration" scheme [DBS+02, Die02, DFS+02] – that is the focus of this paper – reduces sampling times and feedback delay by a dovetailing of the dynamics of the system with the dynamics of the optimization algorithm. In principle only one optimization iteration is performed per sampling instant and the obtained estimate for the optimal solution is shifted suitably to allow overall fast convergence. The approach allows to efficiently treat large-scale systems [FDU+02] or systems with short timescales [DBS03] on standard computers, thus pushing forward the frontier of practical applicability of NMPC. In its actual implementation for continuous time systems, the scheme is based on the direct multiple shooting method within the optimal control package MUSCOD-II (Leineweber [Lei99]), and it has already been successfully applied for the NMPC of a real pilot plant distillation column [DUF+01, DFS+03].

However, to concentrate on the essential features of the method and – most important – on a proof of nominal stability of the scheme, we restrict the presentation in this paper to a strongly simplified NMPC scheme for discrete time systems, as follows.

1.1 Discrete Time Nonlinear Model Predictive Control

Throughout this paper, we consider the following nonlinear discrete time system:

$$x^{k+1} = f(x^k, u^k), \quad k = 0, 1, 2, \dots, \tag{1}$$

with system states $x^k \in \mathbb{R}^{n_x}$ and controls $u^k \in \mathbb{R}^{n_u}$. We assume that $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ is twice continuously differentiable, and, without loss of generality, that the origin is a steady state for (1), i.e. $f(0, 0) = 0$.

The aim of NMPC is to find controls $u^k = u(x^k)$ that depend on the current system state x^k and that are optimal with respect to a specified objective on a moving horizon, which implicitly captures the desire that the system converges towards the steady state. We will denote the predicted states and controls by s_i and q_i, in order to distinguish them from the states x^k and controls u^k of the real system. For the derivations considered in this paper we assume that the objective minimized at every time instant k is given by

$$\sum_{i=0}^{N} L(s_i, q_i),$$

where $s_i,\ i = 0, \dots, N$ is the predicted state over the fixed prediction horizon $N \in \mathbb{N}$ starting from x^k

considering a predicted input sequence $(q_0, q_1, \ldots, q_N)$:

$$s_{i+1} = f(s_i, q_i),\ i = 0, \ldots, N, \qquad s_0 = x^k.$$

We assume that the stage cost $L : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}$ is twice continuously differentiable, that $L(0,0) = 0$, and that there is a $m > 0$ such that

$$L(x,u) \geq m\|x\|^2, \quad \forall\, x \in \mathbb{R}^{n_x}, u \in \mathbb{R}^{n_u}. \tag{2}$$

A typical choice for L is e.g. $L(x,u) = x^T Q x + u^T R u$ with positive definite matrices Q and R.

Given this setup, the input applied in NMPC is defined as the first input q_0^* of the optimal[1] predicted input sequence $(q_0^*, \ldots, q_N^*)$:

$$u(x^k) := q_0^*(x^k). \tag{3}$$

The closed loop system then obeys the "ideal-NMPC dynamics"

$$x^{k+1} = f(x^k, u(x^k)), \tag{4}$$

and one central question in NMPC is if the closed loop system (4) is stable. This question has been examined extensively over recent years and a variety of NMPC schemes exist that can guarantee stability, see e.g. [MM90, ABQ$^+$99, DMS00]. For the purposes of this paper we enforce stability using a so called zero terminal constraint in the prediction, i.e.

$$s_{N+1} = 0, \quad \text{(or, equivalently,} \quad f(s_N, q_N) = 0)$$

and we will provide a nominal stability result in Theorem 4.1.

Summarizing, in NMPC we proceed by solving a sequence of optimization problems $P(x^k)$ of the following form:

Definition 1.1 ($P(x)$)

$$\min_{\substack{s_0, \ldots, s_N, \\ q_0, \ldots, q_N}} \quad \sum_{i=0}^{N} L(s_i, q_i) \tag{5a}$$

subject to

$$x - s_0 = 0, \tag{5b}$$

$$f(s_i, q_i) - s_{i+1} = 0, \quad i = 0, \ldots, N-1, \tag{5c}$$

$$f(s_N, q_N) = 0. \tag{5d}$$

As said, the vectors s_i, q_i are introduced to avoid confusion with the real system states x and the inputs u. Note that the optimal solution $(s_0^*(x), \ldots, s_N^*(x), q_0^*(x), \ldots, q_N^*(x))$ of $P(x)$, if it exists, satisfies $s_0^*(x) = x$, and – because of the definition of the "ideal NMPC control" in (3) – also $q_0^*(x) = u(x)$.

[1] Optimal values are in the following denoted by a star.

Assumption 1 *For all initial values x in an open set $X \subset \mathbb{R}^{n_x}$ that contains the origin, problem $P(x)$ has a unique optimal solution $(s_0^*(x), \ldots, s_N^*(x), q_0^*(x), \ldots, q_N^*(x))$, and the value function $V(x)$ which is defined via the optimal cost for every x by*

$$V(x) := \sum_{i=0}^{N} L(s_i^*(x), q_i^*(x)) \tag{6}$$

is continuous on this set X. Furthermore, there is a (possibly large) $M > 0$ such that $V(x) \leq M\|x\|^2$ $\forall x \in X$.

Note that the steady state trajectory $(0, 0, \ldots, 0)$ is the solution of $P(0)$ and has optimal cost $V(0) = 0$, and that because of $V(x) \geq L(x, q_0^*(x)) \geq m\|x\|^2$ we also have $V(x) > 0,\ \forall x \in X \backslash \{0\}$. In the remainder of this paper we are not interested in the set X, but rather in the largest compact level set of V that is contained in X. Thus in the following we consider a fixed $\alpha > 0$ such that

$$X_\alpha := \{x \in X | V(x) \leq \alpha\} \subset X, \tag{7}$$

is maximal and that X_α is compact. Clearly, X_α contains a neighborhood of the origin. This set X_α corresponds to the region of attraction of the ideal NMPC controller: for all $x^0 \in X_\alpha$ we can prove asymptotic stability of the ideal NMPC dynamics (4), i.e., $\lim_{k\to\infty} x_k = 0$, as will be stated in Theorem 4.1 in Section 4.1.

Remark: In practical applications, inequality path constraints of the form $h(x_i, q_i) \geq 0$, like bounds on controls or states, are of major interest, and should be included in the formulation of the optimization problems $P(x)$. For the purpose of this paper we leave such constraints unconsidered, since general convergence results for Newton type methods with changing active sets are difficult to establish. However, we note that in the practical implementation of the real-time iteration scheme they are included.

1.2 Sequential versus Simultaneous Solution Approaches

Existing numerical schemes for NMPC optimization can roughly be subdivided into *sequential* and *simultaneous* solution strategies [BBB$^+$01, BR91b, Pyt99]. In the *sequential* approach, the system equations (5b) and (5c) are used to eliminate the states $(s_0, \ldots, s_N)$ from the optimization problem, regarding them as a function of the controls $(q_0, \ldots, q_N)$, and substitutes these functions into the objective (5a) and the terminal constraint (5d); thus, the system equations and the optimization problem are treated sequentially, one after the other, in each optimization iteration. Many real-time optimization schemes for NMPC are based on this approach. However, sequential optimization schemes for NMPC often suffer from the drawback that poor initial guesses for the control trajectory may lead the predicted state trajectories far away from the desired reference trajectory; in particular, it may be difficult to satisfy the terminal constraint (5d); therefore, the sequential approach often causes an unnecessarily strong nonlinearity of the resulting optimization problem and poor convergence behaviour, especially for unstable systems. In some cases, an open-loop simulation on a longer horizon is even impossible.

In contrast to this, the *simultaneous* approach avoids this difficulty by keeping both, the control *and* the state in the optimization problem, and treating the problem $P(x)$ exactly as it is formulated in (5), thus solving system equations and optimization problem simultaneously. Though the resulting optimization

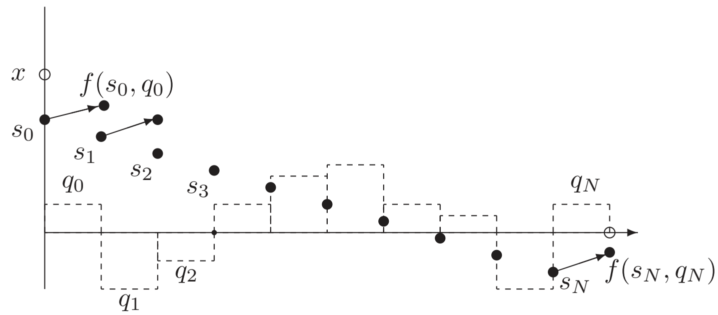

Figure 1: Problem $P(x)$: initial value x and NLP variables $s_0, \ldots, s_N$ and $q_0, \ldots, q_N$.

problem in the variables $(s_0, \ldots, s_N, q_0, \ldots, q_N)$ may be large-scale, it has a favourable structure and can be efficiently solved, and instability and nonlinearity of the dynamic model can be better controlled. Note that for some guess of the optimization variables, the state trajectory $(s_0, \ldots, s_N)$ need not necessarily satisfy the system equations (5b) and (5c) (for a visualization, see Fig. 1), but that a solution trajectory of course satisfies all constraints. The real-time iteration scheme is based on this simultaneous approach.

1.3 Online NMPC and System-Optimizer Dynamics

In ideal NMPC it is assumed that the feedback $u(x^k)$ is available instantaneously at every sampling time k. However, in practice usually no explicit solution to the problem $P(x^k)$ is available, and the numerical solution requires a non negligible computation time and involves some numerical errors. We typically know each initial value x^k only at the time k when the corresponding control u^k is already required for implementation. Thus, instead of implementing the ideal NMPC control $u(x^k)$ we have to use some quickly available approximation $\tilde{u}(x^k, w^k)$, where the additional argument w^k indicates a data vector $w^k \in \mathbb{R}^n$ that we use to parameterize the control approximation. These data are generated by an online optimization algorithm, and they may be updated from one time step to the next one, according to the law $w^{k+1} = F(x^k, w^k)$, where the argument x^k takes account of the fact that the update shall of course depend on the current system state. To achieve this the computations performed are thus divided in two parts

1. Preparation: computation of $w^k = F(x^{k-1}, w^{k-1})$, and generation of the feedback approximation function $\tilde{u}(\cdot, w^k)$, during the transition of the system from state x^{k-1} to x^k.

2. Feedback Response: At time k, give the feedback approximation $u^k := \tilde{u}(x^k, w^k)$ to the system, which then evolves according to $x^{k+1} = f(x^k, u^k)$.

From a system theoretic point of view, instead of the ideal NMPC dynamics (4), we now have to investigate the combined system-optimizer dynamics

$$x^{k+1} = f(x^k, \tilde{u}(x^k, w^k)), \quad (8a)$$
$$w^{k+1} = F(x^k, w^k). \quad (8b)$$

The difficulty in the analysis of the closed-loop behavior of this system stems from the fact that the two subsystems mutually depend on each other.

The real-time iteration scheme investigated in this paper is one specific approach to online NMPC, where the data vector w^k is essentially a guess for the optimal solution trajectory of $P(x^k)$. The data update law $w^{k+1} = F(x^k, w^k)$ shall provide iteratively refined solution guesses, and is derived from a Newton type optimization scheme. The approximate feedback law $\tilde{u}(x^k, w^k)$ can be considered an (essential) by-product of this Newton type iteration scheme.

1.4 Organisation of the Paper

The principal aim of the paper is to prove a nominal stability result for the system-optimizer dynamics (8) due to the real-time iteration scheme. The investigation has to combine concepts from both, classical stability theory for NMPC as well as from convergence theory for Newton type optimization methods.

In Section 2 we introduce the real-time iteration scheme and its combined system-optimizer dynamics in x^k and w^k. Section 3 contains a detailed discussion of the convergence properties of Newton type methods for NMPC and a convergence result for ideal NMPC optimization with a shift initialization for each new optimization problem. In Section 4 we review a nominal stability result for ideal NMPC that is based on a decrease of the optimal value function $V(x^k)$ in each time step, and in Subsection 4.2 we give a bound on the errors due to the feedback approximation $\tilde{u}(x^k, w^k)$ in the real-time iteration scheme, with respect to the decrease of the value function. In Section 5 we analyze the contraction properties of the optimizer states w^k under the assumption that the system states x^k stay in the level set X_α. In Section 6 we finally combine the results of Section 4.2 and Section 5 to prove convergence of the real-time iteration NMPC scheme, and in Section 7 we conclude the paper with a short summary.

2 Real-Time Iteration Scheme

In order to characterize the solution of the optimization problem $P(x)$ we introduce the Lagrange multipliers $\lambda_0, \ldots, \lambda_N$ for the constraints (5c) and λ_{N+1} for (5d), and define the Lagrangian function $\mathcal{L}_x(\lambda_0, s_0, q_0, \ldots)$ of problem $P(x)$ as

$$\mathcal{L}_x(\cdot) \quad := \quad \textstyle\sum_{i=0}^{N} L(s_i, q_i) + {\lambda_0}^T (x - s_0) + \sum_{i=0}^{N-1} {\lambda_{i+1}}^T (f(s_i, q_i) - s_{i+1}) + {\lambda_{N+1}}^T f(s_N, q_N).$$

We assume in the following that $\mathcal{L}_x$ is twice continuously differentiable in its arguments over the considered regions. Summarizing all variables in a vector $w := (\lambda_0, s_0, q_0, \ldots, \lambda_N, s_N, q_N, \lambda_{N+1}) \in \mathbb{R}^n$, the necessary optimality conditions of first order for $P(x)$ are:

$$\nabla_w \mathcal{L}_x(w) = \begin{bmatrix} x - s_0 \\ \nabla_x L(s_0, q_0) + \frac{\partial f}{\partial x}(s_0, q_0)^T \lambda_1 - \lambda_0 \\ \nabla_u L(s_0, q_0) + \frac{\partial f}{\partial u}(s_0, q_0)^T \lambda_1 \\ \vdots \\ f(s_{N-1}, q_{N-1}) - s_N \\ \nabla_x L(s_N, q_N) + \frac{\partial f}{\partial x}(s_N, q_N)^T \lambda_{N+1} - \lambda_N \\ \nabla_u L(s_N, q_N) + \frac{\partial f}{\partial u}(s_N, q_N)^T \lambda_{N+1} \\ f(s_N, q_N) \end{bmatrix} = 0. \tag{9}$$

One possible solution method for this set of nonlinear equations is to use Newton type iterations, as outlined in the following.

2.1 Newton Type Iterations

Starting at some guess w for the optimal solution of $P(x)$, the Newton type methods investigated in this paper compute a corrected iterate $w' = w + \Delta w(x, w)$ towards the exact solution by

$$\Delta w(x, w) := -J(w)^{-1} \; \nabla_w \mathcal{L}_x(w), \tag{10}$$

where $J(w)$ is an approximation of the second derivative $\nabla_w^2 \mathcal{L}_x(w)$. Note that $\nabla_w^2 \mathcal{L}_x$ – often called *Karush-Kuhn-Tucker (KKT)* matrix – is independent of the initial value x, which enters the Lagrangian $\mathcal{L}_x(w)$ only linearly. The index argument x can therefore be omitted for the KKT matrix, i.e., we will write $\nabla_w^2 \mathcal{L}(w)$ in the sequel. Of course, its approximation $J(w)$ shall also be independent of x, and we assume in the following that $J(w)$ is continuous over the considered regions. Moreover, the steps $\Delta w(x, w) = (\Delta\lambda_0(x, w), s_0(x, w), \ldots)$ shall have the property that $s_0' = s_0 + \Delta s_0(x, w) = x$, i.e., that the linear initial value constraint (5b), $x - s_0 = 0$, is satisfied after one Newton type iteration. This is easily accomplished by noting that the first n_x rows of $\nabla_w^2 \mathcal{L}(w)$ are constant, cf. (11), and choosing them to also be the first n_x rows of $J(w)$.

We mention here that the Lagrangian function $\mathcal{L}_x$ of the optimal control problem is *partially separable* and its second derivative therefore has a block diagonal structure,

$$\nabla_w^2 \mathcal{L}(w) = \begin{bmatrix} & -\mathbb{I} & & & & & & \\ -\mathbb{I} & Q_0 & M_0 & A_0^T & & & & \\ & M_0^T & R_0 & B_0^T & & & & \\ & A_0 & B_0 & & & & & \\ & & & & \ddots & -\mathbb{I} & & \\ & & & & -\mathbb{I} & Q_N & M_N & A_N^T \\ & & & & & M_N^T & R_N & B_N^T \\ & & & & & A_N & B_N & \end{bmatrix}, \tag{11}$$

which should also be chosen to be the structure of $J(w)$, and which should be exploited in the actual implementation of the Newton type method.

2.2 Real-Time Iteration Algorithm with Shift

Let us assume that during the transition from one sampling instant to the next we only have time to perform one Newton type iteration. To allow fast convergence while the process evolves, the real-time iteration scheme is based on a suitable transition between subsequent problems. After an initial disturbance it subsequently delivers approximations u^k for the optimal feedback control that allow to steer the system close to the desired steady state, as will be shown in Section 6, under suitable conditions.

Furthermore, as shown in [Die02], the computations of the real-time iteration belonging to problem $P(x^k)$ can largely be prepared *without knowledge of the value of* x^k, so that the approximation u^k of the optimal feedback control is practically available at the time k. To underline the basic idea, suppose that the inverse $J(w)^{-1}$ is available in explicit form (which is in practice never computed). Therefore one can write (10) as

$$\begin{bmatrix} \Delta\lambda_0 \\ \Delta s_0 \\ \Delta q_0 \\ \vdots \end{bmatrix} = - \begin{bmatrix} \cdot & \cdots \\ -\mathbb{I}_{n_x} & \cdots \\ K(w) & \cdots \\ \vdots & \ddots \end{bmatrix} \quad \begin{bmatrix} x - s_0 \\ \vdots \end{bmatrix}. \tag{12}$$

The real-time iteration scheme with shift initialization proceeds now as follows:

1. Preparation: Based on the current guess $w^k = (\lambda_0^k, s_0^k, q_0^k, \lambda_1^k, s_1^k, q_1^k, \ldots, \lambda_N^k, s_N^k)$ compute all components of the vector $\nabla_w \mathcal{L}_{x^k}(w^k)$ apart from the first one, and compute the matrix $J(w^k)$. Prepare the linear algebra computation for the implicit representation of the inverse-vector product $J(w^k)^{-1}\nabla_w \mathcal{L}_{x^k}(w^k)$ as much as possible without knowledge of the value of x^k (a detailed description how this can be achieved is given in [DBS+02] or [Die02]). Essentially, this amounts to providing the matrix $K(w^k)$ as in (12).

2. Feedback Response: At the time k, when x^k measured, compute the feedback approximation $u^k = \tilde{u}(x^k, w^k) := q_0^k - K(w^k)(x^k - s_0^k)$ and apply the control u^k immediately to the real system.

3. Transition: Compute the next initial guess w^{k+1} by first adding the step vector Δw^k to w^k and then shifting all variables to account for the movement in time. That is, compute w^{k+1} as

$$w^{k+1} := S\left(w^k + \Delta w^k\right) = S\left(\ w^k - J(w^k)^{-1}\ \nabla_w \mathcal{L}_{x^k}(w^k)\ \right),$$

where S is a shifting matrix operating on

$$w = \left[\begin{array}{c} \lambda_0 \\ s_0 \\ q_0 \\ \hline \lambda_1 \\ s_1 \\ q_1 \\ \cdot \\ \cdot \\ \cdot \\ \lambda_N \\ s_N \\ q_N \\ \lambda_{N+1} \end{array}\right] \quad \text{such that} \quad Sw = \left[\begin{array}{c} \lambda_1 \\ s_1 \\ q_1 \\ \cdot \\ \cdot \\ \cdot \\ \lambda_N \\ s_N \\ q_N \\ \lambda_{N+1} \\ \hline 0 \\ 0 \\ 0 \end{array}\right].$$

Continue by setting $k = k + 1$ and going to 1.

In contrast to the ideal NMPC feedback closed loop (4), in the real-time iteration scheme we have to regard combined system-optimizer dynamics of the form (8), which are given by

$$x^{k+1} = f\left(x^k, q_0^k - K(w^k)(x^k - s_0^k)\right) \qquad = f\left(\ x^k, q_0^k + \Delta q_0(x^k, w^k)\ \right) \tag{13a}$$

$$w^{k+1} = S\left(\ w^k - J(w^k)^{-1}\ \nabla_w \mathcal{L}_{x^k}(w^k)\ \right) \qquad = S\left(\ w^k + \Delta w(x^k, w^k)\ \right). \tag{13b}$$

In the remainder of the paper we concentrate on investigating the nominal stability of these system-optimizer dynamics.

2.3 Connection to Existing Approaches

Several features of the algorithm have been presented by other researchers for real-time optimization in NMPC. In particular, a one-iteration scheme has been proposed by Li and Biegler in [LB89] for the sequential approach. For this scheme even a stability result is derived, that is, however, only applicable to stable systems. In the application of classical off-line optimization schemes to on-line control, the question of how to initialize subsequent problems has found some attention in the literature [BR91a, LEL92], and a shift strategy has been proposed, e.g., by de Oliveira and Biegler [OB95a] for the sequential approach.

3 Local Convergence of Newton Type Optimization

In this section we present results on the convergence properties of Newton type methods for optimization in NMPC that lay the basis for the discussion in all subsequent sections.

3.1 Local Convergence for a Single Optimization Problem

In a first step we review a local convergence result of Newton type optimization for the solution of one fixed optimization problem (i.e. no shift of w after each iteration). Thus we consider in this subsection a fixed $x \in X_\alpha$ and we will denote in the following by w_0 an (arbitrary) initial guess for the primal-dual variables of problem $P(x)$. A standard Newton type scheme proceeds by computing iterates $w_1, w_2, \ldots$ according to

$$w_{i+1} := w_i + \Delta w_i, \quad \Delta w_i := \Delta w(x, w_i) = -J(w_i)^{-1} \nabla_w \mathcal{L}_x(w_i).$$

The following standard result states conditions that ensure the convergence of the iterates (for fixed x) from the initial guess w_0 to a point that satisfies the first order necessary conditions:

Theorem 3.1 (Local Convergence of Newton Type Optimization)
Assume that $J(w)$ is invertible for all $w \in D$, where $D \subset \mathbb{R}^n$. Furthermore, assume that there exist constants $\kappa < 1$, $\omega < \infty$ such that for all $w', w \in D$, $\Delta w = w' - w$ and all $t \in [0,1]$

$$\left\| J(w')^{-1} \left(J(w + t\Delta w) - \nabla_w^2 \mathcal{L}(w + t\Delta w) \right) \Delta w \right\| \leq \kappa \|\Delta w\|, \tag{14a}$$

$$\left\| J(w')^{-1} \left(J(w + t\Delta w) - J(w) \right) \Delta w \right\| \leq \omega t \|\Delta w\|^2, \tag{14b}$$

that the the first step $\Delta w_0 := -J(w_0)^{-1} \nabla_w \mathcal{L}_x(w_0)$ is sufficiently small, such that

$$\delta_0 := \kappa + \frac{\omega}{2} \|\Delta w_0\| < 1, \tag{14c}$$

and that the ball $B_0 := \left\{ w \in \mathbb{R}^n | \; \|w - w_0\| \leq \frac{\|\Delta w_0\|}{1-\delta_0} \right\}$ is completely contained in D. Then the Newton type iterates $w_0, w_1, \ldots$ are well-defined, stay in the ball B_0, and converge towards a point $w^* \in B_0$ satisfying $\nabla_w \mathcal{L}_x(w^*) = 0$.

Remark: We would like to mention that the assumptions made are standard assumptions for the convergence of Newton type methods (see e.g. [Boc87]). One should note that in general it is rather difficult to check the conditions a priori, but that a posteriori estimates can be obtained when the Newton type iterations are carried out.

For the proof of the theorem we need the following lemma:

Lemma 3.2 (Contraction Rate)
Under the same assumptions as in Theorem 3.1 the Newton type iterates satisfy the contraction property

$$\|\Delta w_{i+1}\| \leq \left(\kappa + \frac{\omega}{2}\|\Delta w_i\|\right) \|\Delta w_i\| =: \delta_i \|\Delta w_i\|. \tag{15}$$

Proof of Lemma3.2: We prove the lemma using a standard arguments for convergence of Newton type methods (see e.g. [Boc87]):

$$\begin{aligned}
\|\Delta w_{i+1}\| &= \|J(w_{i+1})^{-1} \cdot \nabla_w \mathcal{L}_x(w_{i+1})\| \\
&= \|J(w_{i+1})^{-1} \cdot (\nabla_w \mathcal{L}_x(w_{i+1}) - \nabla_w \mathcal{L}_x(w_i) - J(w_i) \cdot \Delta w_i)\| \\
&= \|J(w_{i+1})^{-1} \cdot \textstyle\int_0^1 (\nabla_w^2 \mathcal{L}(w_i + t\Delta w_i) - J(w_i)) \cdot \Delta w_i \, dt\| \\
&= \|J(w_{i+1})^{-1} \cdot \textstyle\int_0^1 (\nabla_w^2 \mathcal{L}(w_i + t\Delta w_i) - J(w_i + t\Delta w_i))\Delta w_i \, dt \\
&\quad + J(w_{i+1})^{-1} \cdot \textstyle\int_0^1 (J(w_i + t\Delta w_i) - J(w_i))\Delta w_i \, dt\| \\
&\leq \textstyle\int_0^1 \|J(w_{i+1})^{-1} \, (\nabla_w^2 \mathcal{L}(w_i + t\Delta w_i) - J(w_i + t\Delta w_i))\Delta w_i\| \, dt \\
&\quad + \textstyle\int_0^1 \|J(w_{i+1})^{-1} \, (J(w_i + t\Delta w_i) - J(w_i))\Delta w_i\| \, dt \\
&\leq \kappa \|\Delta w_i\| + \textstyle\int_0^1 \omega t \|\Delta w_i\|^2 \, dt \\
&= \left(\kappa + \tfrac{\omega}{2}\|\Delta w_i\|\right) \|\Delta w_i\| = \delta_i \|\Delta w_i\|.
\end{aligned}$$

□

Proof of Theorem 3.1: Using Lemma 3.2 we first observe that $\delta_{i+1} \leq \delta_i$ and that

$$\|\Delta w_i\| \leq \delta_{i-1}\delta_{i-2}\ldots\delta_0 \|\Delta w_0\| \leq (\delta_0)^i \|\Delta w_0\|.$$

so that

$$\|w_i - w_{i+m}\| \leq \|\Delta w_i\| + \ldots + \|\Delta w_{i+m-1}\| \leq \frac{(\delta_0)^i \|\Delta w_0\|}{1 - \delta_0}$$

i.e., $w_0, w_1, w_2, \ldots$ is a Cauchy sequence and remains in the (compact) ball B_0, and thus converges towards a point $w^* \in B_0$. This point satisfies $\nabla_w \mathcal{L}_x(w^*) = 0$ due to continuity of $\nabla_w \mathcal{L}_x(\cdot)$ and boundedness of J on the compact ball B_0, as

$$\|\nabla_w \mathcal{L}(w^*)\| = \lim_{i \to \infty} \|\nabla_w \mathcal{L}(w_i)\| = \lim_{i \to \infty} \|J(w_i)\Delta w_i\| \leq \|J\|_{\max} \lim_{i \to \infty} \|\Delta w_i\| = 0.$$

□

3.2 Local Convergence for a Class of Optimization Problems

We will tailor in this subsection the results of the previous subsection to the NMPC problem. For this purpose we need to define two sets $D_C \subset D_{2C}$ which are defined in terms of a fixed $C > 0$

$$\begin{aligned}
D_C &:= \{w \in \mathbb{R}^n \,|\, \exists x \in X_\alpha, \quad \|w - w^*(x)\| \leq C\} \qquad (16) \\
D_{2C} &:= \{w \in \mathbb{R}^n \,|\, \exists x \in X_\alpha, \quad \|w - w^*(x)\| \leq 2C\}, \qquad (17)
\end{aligned}$$

where $w^*(x)$ is the primal-dual solution of problem $P(x)$, and where X_α is the maximum level set of V in X as introduced in 1.1. Given these sets we can now state the assumptions necessary for the following corollary.

Assumption 2 *Each solution $w^*(x)$ is unique in D_{2C}, i.e.,*

$$\forall x \in X_\alpha, \forall w \in D_{2C} \backslash \{w^*(x)\} : \nabla_w \mathcal{L}_x(w) \neq 0, \tag{18a}$$

and $J(w)$ is invertible on $\in D_{2C}$. Furthermore there exist constants $\omega < \infty$, $\kappa < 1$ such that for all $w', w \in D_{2C}$, $\Delta w = w' - w$ and all $t \in [0, 1]$

$$\left\| J(w')^{-1} \left(J(w + t\Delta w) - \nabla_w^2 \mathcal{L}(w + t\Delta w) \right) \Delta w \right\| \leq \kappa \|\Delta w\| \tag{18b}$$

$$\left\| J(w')^{-1} \left(J(w + t\Delta w) - J(w) \right) \Delta w \right\| \leq \omega t \|\Delta w\|^2. \tag{18c}$$

The following two scalars d and δ will be used throughout the paper.

Definition 3.1 *Given a fixed $C > 0$, that shall be chosen as large as possible such that Assumption 2 holds, we define the positive scalars*

$$d := \frac{C(1-\kappa)}{1 + \frac{\omega}{2} C} \quad \textit{and} \quad \delta := \kappa + \frac{\omega}{2} d. \tag{19}$$

Note that

$$\delta = \frac{\kappa + \frac{\omega}{2} C}{1 + \frac{\omega}{2} C} < 1. \tag{20}$$

Now we can state the following corollary giving conditions for the convergence of Newton type methods for NMPC:

Corollary 3.3 (Local Convergence of Newton type methods for NMPC problems)
Suppose Assumption 2. If for some $x \in X_\alpha$ and some $w_0 \in D_C$ it holds that $\|\Delta w(x, w_0)\| \leq d$, then the Newton type iterates w_i for the solution of $\nabla_w \mathcal{L}_x(w) = 0$, initialized with the initial guess w_0, converge towards the solution $w^*(x)$. Furthermore, the iterates remain in D_C.

Proof: We start by noting that $C = \frac{d}{1-\delta}$. The ball B_0 of Theorem 3.1 is contained in the ball $\{w' \in \mathbb{R}^n | \ \|w' - w_0\| \leq C\}$, which itself is contained in the set D_{2C}, as $w_0 \in D_C$. Therefore, there is a solution $w^* \in D_{2C}$ satisfying $\nabla_w \mathcal{L}_x(w^*) = 0$, which must be equal to $w^*(x)$ due to the uniqueness assumption (18a). Furthermore, the distance of iterate w_i from $w^*(x)$ is bounded by

$$\|w_i - w^*(x)\| \leq \frac{\|\Delta w_i\|}{1 - \delta_i} \leq \frac{d}{1-\delta} = C, \tag{21}$$

i.e., $w_i \in D_C$. □

In the remainder of the paper we will consider fixed values for α and C and assume that Assumption 2 is satisfied. Furthermore, we will often refer to the set Ξ defined as follows:

Definition 3.2 (Ξ)

$$\Xi := \{(x, w) \in \mathbb{R}^{n_x} \times \mathbb{R}^n \, | x \in X_\alpha, w \in D_C, \|\Delta w(x, w)\| \leq d\} \tag{22}$$

This set Ξ contains all pairs (x, w) for which Corollary 3.3 ensures numerical solvability. Note that Ξ is nonempty, as it contains at least the points $(x, w^*(x))$, $\forall x \in X_\alpha$, and their neighborhoods.

3.3 Local Convergence for Ideal NMPC with Shift Initialization

We are now interested what influence a shift initialization has on the Newton type solution of two consecutive ideal NMPC problems $P(x^{k+1})$ and $P(x^k)$. In other words, we want to investigate under which conditions a shifted version of the previous solution, $w^*(x^k)$, i.e., setting $w_0^{k+1} := Sw^*(x^k)$, leads to convergence of the Newton scheme at time k. Here $w^*(x^k)$ denotes the optimal solution at time k, while w_0^{k+1} denotes the initialization of the Newton type iteration $w_0^{k+1}, w_1^{k+1}, \ldots$, at time $k+1$, which is given by the iteration rule $w_{i+1}^{k+1} := w_i^{k+1} + \Delta w(x^{k+1}, w_i^{k+1})$ and satisfy $w_i^{k+1} \to w^*(x^{k+1})$ if the initial guess w_0^{k+1} is sufficiently close to the solution, i.e., if $(x^k, w_0^{k+1}) \in \Xi$.

Note that the shifted initialization has the advantage that the initial value constraint (5b) of the new problem is already satisfied. But is this initialization close enough to the exact solution $w^*(x^k)$ to guarantee local convergence?

The following theorem gives a partial answer to this question; roughly speaking, the shift provides a good initialization if the length N of the optimization horizon is chosen sufficiently big, so that the zero terminal constraint (5d) is not to strongly active, i.e., that the last multiplier λ_{N+1} is sufficiently small.

Theorem 3.4 (Numerical Solvability for Ideal NMPC with Shift)
Assume that for all $w^*(x) = (\lambda_0^*, s_0^*, \ldots, \lambda_{N+1}^*)$, $x \in X_\alpha$

$$\left\| J(Sw^*(x))^{-1} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \lambda_{N+1}^*(x) \\ 0 \\ 0 \end{bmatrix} \right\| \leq d. \tag{23}$$

Then $(x^{k+1}, Sw^*(x^k)) \in \Xi$, i.e., the shift initialization $w_0^{k+1} := Sw^*(x^k)$ for each new problem $P(x^{k+1})$ guarantees convergence of the Newton type scheme towards the new optimal solution $w^*(x^{k+1})$.

Proof: First note that $f(x, u(x)) = f(s_0^*(x), q_0^*(x))$ and $Sw^*(x) = (\lambda_1^*(x), s_1^*(x), \ldots, \lambda_{N+1}^*, 0, 0, 0)$. Thus it holds that

$$\begin{aligned}
\nabla_w \mathcal{L}_{f(x,u(x))}(Sw^*(x)) &= \begin{bmatrix} f(s_0^*(x), q_0^*(x)) - s_1^*(x) \\ \vdots \\ f(s_N^*(x), q_N^*(x)) - 0 \\ \nabla_x L(0,0) + \frac{\partial f}{\partial x}(0,0)^T 0 - \lambda_{N+1}^*(x) \\ \nabla_u L(0,0) + \frac{\partial f}{\partial u}(0,0)^T 0 \\ f(0,0) \end{bmatrix} \\
&= S\nabla_w \mathcal{L}_x(w^*(x)) + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -\lambda_{N+1}^*(x) \\ 0 \\ 0 \end{bmatrix} = 0 + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -\lambda_{N+1}^*(x) \\ 0 \\ 0 \end{bmatrix}.
\end{aligned}$$

Therefore, condition (23) is equivalent to

$$\|J(Sw^*(x))^{-1}\nabla_w \mathcal{L}_{f(x,u(x))}(Sw^*(x))\| \le d, \quad \forall x \in X_\alpha,$$

i.e.,

$$(f(x,u(x)), Sw^*(x)) \in \Xi, \quad \forall x \in X_\alpha, \tag{24}$$

and in particular

$$(f(x,u(x^k)), Sw^*(x^k)) = (x^{k+1}, Sw^*(x^k)) \in \Xi, \quad \forall k \in \mathbb{N}.$$

❑

A direct consequence of the theorem is that under certain conditions one cannot only guarantee closed loop stability, but also numerical solvability for all optimization problems $P(x^k)$, if the initial state x^0 is in X_α and if the first initial guess $w_0^0 \in D_C$ is such that $\|\Delta w(x^0, w_0^0)\| \le d$ (i.e. that $(x^0, w_0^0) \in \Xi$). However, this favorable result was obtained under the assumption that computation times are negligible, i.e., that the Newton type method can be iterated until convergence at every sampling time.

Remark: A major difference of the real-time iteration scheme as described in Section 2.2 to the ideal scheme considered in Theorem 3.4 is that the iterations $w_0^k, w_1^k, \ldots$ for problem $P(x^k)$, $w_i^k \to w^*(x^k)$, are terminated prematurely, namely after the first iteration. Instead of initializing each new problem $P(x^{k+1})$ by $w_0^{k+1} := S(\lim_{i\to\infty} w_i^k) = Sw^*(x^k)$, we initialize with $w_0^{k+1} := Sw_1^k = S(\, w_0^k + \Delta w(x^k, w_0^k)\,)$. For the real-time iterations, we simply drop the lower iteration index and set $w^k := w_0^k$.

4 Nominal Stability and Decrease of the Optimal Value Function

In this section we will first review a well known result for nominal stability of ideal NMPC, which is based on a guaranteed decrease of the value function. The line of proof allows us then to examine the influence of the "input disturbance" introduced by the feedback approximation of the real-time iteration scheme (compared to ideal NMPC). We will give a bound on the error of the feedback approximation with regard to the decrease of the value function.

4.1 Nominal Stability for Ideal NMPC

Let us first review the following result for nominal stability of ideal NMPC (cf. [MM90, ABQ$^+$99, DMS00]):

Theorem 4.1 (Nominal Stability for Ideal NMPC) *Let Assumption 1 hold, and assume that $x^0 \in X_\alpha$. Then the closed-loop dynamics $x^{k+1} = f(x^k, u(x^k))$, $k = 0, 1, \ldots$, generated by the ideal NMPC law* (3) *leads the system state towards the origin,* $\lim_{k\to\infty} x^k = 0$.

Proof: We give an outline of the well known proof here, since this allows us to see that under the formulated assumptions NMPC has some inherent robustness properties, which can be utilized for showing stability of the real-time iteration scheme. As standard in NMPC we use the optimal value function $V(x)$ as a Lyapunov function for the closed-loop system. First note that

$$V(x) - L(x, u(x)) = \sum_{i=1}^{N} L(s_i^*(x), q_i^*(x)).$$

Furthermore it is clear that the shifted state and control vector $(s_1^*(x), \ldots, s_N^*(x), 0)$ and $(q_1^*(x), \ldots, q_N^*(x), 0)$, is a feasible (but not optimal) solution for the next optimal control problem $P(f(x, u(x))$, with associated costs

$$\sum_{i=1}^{N} L(s_i^*(x), q_i^*(x)) + L(0, 0).$$

Since the optimal cost $V(f(x, u(x))$ can only be lower than this value, it follows that

$$V(f(x, u(x))) \leq V(x) - L(x, u(x)) \leq V(x) - m\|x\|^2, \tag{25}$$

and it is clear that $f(x, u(x)) \in X_\alpha$ if $x \in X_\alpha$, so that $V(f(x, u(x))$ is indeed well defined. Note also that

$$V(x^{k+1}) \leq V(x^k) - m\|x^k\|^2 \quad \forall x^k \in X_\alpha.$$

As X_α is assumed to be compact, the sequence $(x^k)_{k \in \mathbb{N}}$ has at least one accumulation point $x^* \in X_\alpha$. By continuity of V and $\|\cdot\|^2$ we obtain

$$V(x^*) \leq V(x^*) - m\|x^*\|^2$$

which can only be satisfied if $x^* = 0$. □

Remark: Recent results on the robustness of Lyapunov functions for discontinuous difference equations and results on the stability of NMPC under perturbations suggest that the ideal NMPC controller has some inherent robustness properties with respect to disturbances under the stated assumptions (in particular because V is continuous). The main observation is that the term $-L(x, u(x))$ in (25) provides some robustness with respect to disturbances that might lead to a lower decrease – but no increase – of the value function V from time step to time step [SRM97, KT02, Fin03, FIAF02]. Thus, considering the error of an approximate feedback compared to the ideal NMPC input $u(x^k)$ as a disturbance, it can be assumed that under certain conditions the closed loop should be stable. We will build on somewhat similar arguments in the proof of the main result of this paper in Section 6. To prepare this proof, we will first provide a bound on the error of the feedback approximation due to the real-time iteration scheme.

4.2 An Error Bound for the Feedback Approximation

In the real-time iteration scheme, instead of applying, at state x, the ideal NMPC control $u(x) := q_0^*(x)$ to the plant, we employ a feedback approximation $\tilde{u}(x, w) := q_0 + \Delta q_0(x, w)$ that depends not only on the system state x but also on the current optimizer parameter vector $w = (\lambda_0, s_0, q_0, \ldots)$. Here, $\Delta q_0(x, w)$

is the first control of the Newton type step vector $\Delta w(x,w) = (\Delta\lambda_0(x,w), \Delta s_0(x,w), \Delta q_0(x,w), \ldots)$. What matters is the error $\epsilon(x,w)$ with respect to the descent property (25)

$$V(f(x,\tilde{u}(x,w))) \le V(x) - L(x,\tilde{u}(x,w)) + \epsilon(x,w). \tag{26}$$

The decrease (and thus convergence to the origin) in the value function along the disturbed trajectory is ensured as long as $-L(x,\tilde{u}(x,w)) + \epsilon(x,w) < 0$. The following theorem establishes a bound on the error $\epsilon(x,w)$, which is quadratic in the Newton type step size $\Delta w(x,w)$. It will be used in the proof of stability for the real-time iteration scheme in Section 6.

Theorem 4.2 (Error Bound for Approximate Feedback)
Suppose Assumptions 1, 2 and 5 hold. Then there is a $\mu > 0$ *such that for each* $(x,w) \in \Xi$

$$V(f(x, q_0 + \Delta q_0(x,w))) \le V(x) - L(x, q_0 + \Delta q_0(x,w)) + \mu \|\Delta w(x,w)\|^2.$$

The theorem is proven in the appendix, where also a specific value for the constant μ is given, in Eq. (42). The purely technical Assumption 5 is also stated in the appendix.

As the theorem states that the error $\epsilon(x,w)$ is small if the Newton type step size $\Delta w(x,w)$ is small, we will in the following section investigate the behaviour of $\|\Delta w(x^k,w^k)\|$ during the real-time iterations.

5 Contractivity of the Real-Time Iterations

Before being able to prove stability of the real-time iteration schem in Section 6 we need to establish some convergence properties of the Newton type iterations in the real-time iteration scheme. For this purpose we recall that the system and optimizer states of the real-time iteration algorithm with shift obey the system-optimizer dynamics (13):

$$\begin{aligned} x^{k+1} &= f(x^k, q_0^k + \Delta q_0(x^k,w^k)), \\ w^{k+1} &= S(\, w^k + \Delta w(x^k,w^k)\,). \end{aligned}$$

To investigate the stability of these combined dynamics we will in this section establish a bound on the size of the steps $\Delta w^k := \Delta w(x^k,w^k)$, which is based on a stricter version of condition (23) in Theorem 3.4.

Assumption 3 *There exist constants* $\sigma > 0$, $\eta > 0$ *with* $\sigma < 1 - \delta$ *and*

$$\eta \le \sqrt{\frac{m}{\alpha}}(1-(\delta+\sigma))d, \qquad \eta \le \frac{1}{2}\frac{m(1-(\delta+\sigma))}{\sqrt{32(M+m)\mu}}, \tag{28}$$

such that for all $(x,w) \in \Xi$, $w' = w + \Delta w(x,w)$

$$\left\| J(Sw')^{-1} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \lambda'_{N+1} \\ 0 \\ 0 \end{bmatrix} \right\| \le \eta\|x\| + \sigma\|\Delta w(x,w)\|. \tag{29}$$

Remark: This assumption is critical for the subsequent reasoning. In particular the fact that η might be required to be quite small deserves some discussion. If $w = w^*(x)$, then $\Delta w(x, w) = 0$ and $w' = w^*(x)$, so the bound can essentially be seen as a bound on the last multiplier $\lambda^*_{N+1}(x)$ whose modulus can be interpreted the "shadow price" of the final state constraint $f(s_N, q_N) = 0$. For a sufficiently large size N of the optimization horizon we expect the multiplier $\lambda^*_{N+1}(x)$ to decrease, as the cost function itself drives the system to the steady state, and the constraint becomes less and less important. Therefore, we can argue that it is reasonable to assume that η can be made sufficiently small by enlarging the optimization horizon – of course, such an enlargement changes the dimensions of the problem and therefore also the matrix J and its inverse, but numerical experiments with large N have shown that – for controllable systems – the vectors $J(w)^{-1}(0, \ldots, 0, 1^T, 0, 0)^T$ only have significant nonzero elements at the end of the horizon and are decaying in backwards direction, i.e., their norms do practically not depend on the dimension N. Note that the conditioning of J is independent of N for controllable systems.

A second, more technical assumption is the following modification of Assumption 2, where the shifting matrix S is introduced

Assumption 4 *For all $w', w \in D_{2C}$, $\Delta w = w' - w$ and all $t \in [0, 1]$*

$$\left\| J(Sw')^{-1} S \left(J(w + t\Delta w) - \nabla^2_w \mathcal{L}(w + t\Delta w) \right) \Delta w \right\| \leq \kappa \|\Delta w\| \tag{30a}$$

$$\left\| J(Sw')^{-1} S \left(J(w + t\Delta w) - J(w) \right) \Delta w \right\| \leq \omega t \|\Delta w\|^2. \tag{30b}$$

As before it should be noted that checking Assumption 4 a priori is in general difficult, if not impossible. The obtained results should rather be seen as a theoretical underpinning of the real-time iteration scheme than as a constructive approach to pick suitable controller parameters for stability. Under the above two assumptions we can prove the following lemma.

Lemma 5.1 (Stepsize Contraction for Real-Time Iterations)
Suppose Assumptions 3 and 4 are satisfied. Furthermore, assume that $(x^k, w^k) \in \Xi$ and $x^{k+1} := f(x^k, q_0^k + \Delta q_0(x^k, w^k)) \in X_\alpha$. Then, using the shorthands $\Delta w^k := \Delta w(x^k, w^k)$ and $w^{k+1} := S(w^k + \Delta w^k)$, the following holds

$$\|\Delta w(x^{k+1}, w^{k+1})\| \leq \left(\kappa + \sigma + \frac{\omega}{2} \|\Delta w^k\| \right) \|\Delta w^k\| + \eta \|x^k\| \leq (\delta + \sigma) \|\Delta w^k\| + \eta \|x^k\|. \tag{31}$$

In particular, $\|\Delta w(x^{k+1}, w^{k+1})\| \leq d$, i.e., $(x^{k+1}, w^{k+1}) \in \Xi$.

Proof: First note that for any $w = (\lambda_0, s_0, q_0, \ldots, \lambda_{N+1}) \in \mathbb{R}^n$ and regardless of $x \in R^{n_x}$,

$$\nabla_w \mathcal{L}_{f(s_0, q_0)}(Sw) = S \nabla_w \mathcal{L}_x(w) + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -\lambda_{N+1} \\ 0 \\ 0 \end{bmatrix}.$$

Let us now introduce for a moment the shorthand $w' := w^k + \Delta w^k$ and observe that $w^{k+1} = Sw'$ and that $x^{k+1} = f(x^k, q_0^k + \Delta q_0^k) = f(s_0', q_0')$. Therefore, we can deduce similar as in Lemma 3.2 that

$$
\begin{aligned}
\|\Delta w(x^{k+1}, w^{k+1})\| &= \|\Delta w(f(s_0', q_0'), Sw')\| \\
&= \|J(Sw')^{-1} \cdot \nabla_w \mathcal{L}_{f(s_0', q_0')}(Sw')\| \\
&= \left\| J(Sw')^{-1} \cdot \left(S\nabla_w \mathcal{L}_{x^k}(w') + \begin{bmatrix} \vdots \\ 0 \\ -\lambda_{N+1}' \\ 0 \\ 0 \end{bmatrix} \right) \right\| \\
&\leq \|J(Sw')^{-1} \cdot S\nabla_w \mathcal{L}_x(w')\| \quad + \quad \eta\|x^k\| + \sigma\|\Delta w^k\| \\
&\leq \left(\kappa + \tfrac{\omega}{2}\|\Delta w^k\|\right)\|\Delta w^k\| \quad + \quad \eta\|x^k\| + \sigma\|\Delta w^k\| \\
&\leq \left(\kappa + \sigma + \tfrac{\omega}{2}\|\Delta w^k\|\right)\|\Delta w^k\| + \eta\|x^k\| \\
&\leq (\delta + \sigma)\|\Delta w^k\| + \eta\|x^k\|,
\end{aligned}
$$

where we have made use of Assumption 3 in the 4th transformation and of Assumption 4 in the 5th, as in the proof of Lemma 3.2. From $m\|x\|^2 \leq V(x) \leq \alpha$ it follows for all $x \in X_\alpha$ that $\|x\| \leq \sqrt{\frac{\alpha}{m}}$, and from $\|\Delta w^k\| \leq d$ and from the left inequality of (28), we can finally deduce that $\|\Delta w(x^{k+1}, Sw^{k+1})\| \leq (\delta + \sigma)\, d + \eta\sqrt{\frac{\alpha}{m}} \leq (\delta + \sigma)\, d + \sqrt{\frac{m}{\alpha}}(1 - (\delta + \sigma))d\sqrt{\frac{\alpha}{m}} = d$.

□

The lemma allows us to conclude the following contraction property for the real-time iterations (x^k, w^k) (as defined in Eqs. (13)), which we use in the following section.

Corollary 5.2 (Shrinking Stepsize for Real-Time Iterations)
Let in addition to Assumptions 1-4 assume that the real-time iterations start with an initialization $(x^0, w^0) \in \Xi$, and that for a given $\alpha_0 \leq \alpha$ and for some $k_0 > 0$ we have that $\forall k \leq k_0 : \; x^k \in X_{\alpha_0}$. Then $\forall k \leq k_0 : \; (x^k, w^k) \in \Xi$ and

$$
\|\Delta w^k\| \leq (\delta + \sigma)^k \|\Delta w^0\| + \frac{\rho}{2}\sqrt{\alpha_0} \quad \textit{with} \quad \rho := \frac{2\eta}{\sqrt{m}(1 - (\delta + \sigma))}. \tag{32}
$$

Proof: Inductively applying Lemma 5.1 to the iterates (x^{k+1}, w^{k+1}), we immediately obtain that $(x^k, w^k) \in \Xi$, for $k = 1, 2, \ldots, k_0$. Similarly one obtains inductively from the contraction inequality (31), $\|\Delta w^{k+1}\| \leq (\delta + \sigma)\|\Delta w^k\| + \eta\|x^k\|$, and the fact that $\|x^k\| \leq \sqrt{\frac{\alpha_0}{m}}$ that

$$
\|\Delta w^k\| \leq (\delta + \sigma)^k \|\Delta w^0\| + \eta\sqrt{\frac{\alpha_0}{m}} \sum_{i=0}^{k-1} (\delta + \sigma)^i \leq (\delta + \sigma)^k \|\Delta w^0\| + \frac{\eta\sqrt{\frac{\alpha_0}{m}}}{1 - (\delta + \sigma)}.
$$

□

We may furthermore ask how many iterations we need to reduce the stepsize such that it becomes smaller than a given level. However, considering Corollary 5.2 we must expect that they will not become smaller than the constant $\frac{\rho}{2}\sqrt{\alpha_0}$ in Eq. (32). But how many iterations do we need, for example, to push the stepsize under twice that level?

Corollary 5.3 (Iterations needed for Stepsize Reduction)
Let us in addition to Assumptions 1-4 assume that the real-time iterations start with an initialization $(x^0, w^0) \in \Xi$*, and that for a given* $\alpha_0 \leq \alpha$ *we have that* $\forall k \leq k_0 : \ x^k \in X_{\alpha_0}$ *for some*

$$k_0 \geq \log_{\delta+\sigma} \left(\frac{\rho\sqrt{\alpha_0}}{2\|\Delta w^0\|} \right). \tag{33}$$

Then

$$\|\Delta w^{k_0}\| \leq \rho\sqrt{\alpha_0}. \tag{34}$$

Proof: From (33) we conclude that

$$(\delta + \sigma)^{k_0} \|\Delta w^0\| \leq \frac{\rho}{2}\sqrt{\alpha_0}.$$

This together with (32) yields (34). ❑

6 Nominal Stability of the Real-Time Iteration Scheme

Equipped with the error bound from Section 4.2 and the contractivity of the real-time iterations from Section 5 we can finally prove nominal stability of the real-time iteration closed-loop scheme. However, since the error $\epsilon(x^k, w^k)$ in the decrease in the value function depends on the real-time stepsize Δw^k, we have to investigate two competing effects: on the one hand, the feedback errors may allow an *increase* in $V(x^k)$, instead of the desired *decrease* that was needed to prove nominal stability for ideal NMPC in Theorem 4.1. On the other hand, we know that the stepsizes Δw^k shrink during the iterations, and thus we also expect the errors to become smaller. Since an increase in the value function might imply that we leave the level set X_α, we will not be able to stabilize with the real-time iteration scheme the whole set X_α (at least not if Δw^0 is too large). Thus, we have to back of a little from the boundary of X_α to allow an increase in the value function without leaving X_α until Δw^k is small enough to guarantee a decrease of the value function. For this reason we will distinguish two phases:

- In the first phase we may have an increase of the value function $V(x^k)$, therefore we must allow for a safety back-off. However, the stepsizes $\|\Delta w^k\|$ can be shown to shrink.
- In the second phase, finally, the numerical errors are small enough to guarantee a decrease of both, $V(x^k)$ and $\|\Delta w^k\|$ and we can prove convergence of iterates (x^k, w^k) towards the origin $(0, 0)$.

6.1 Phase 1: Increase in Objective, but Decrease in Stepsize

Exploiting Corollary 5.3, let us define the number k_α of iterations that are at maximum needed for reduction of the stepsize under the value $\rho\sqrt{\alpha}$ if all iterates stay in the level set X_α.

Definition 6.1 (k_α **and** Ξ_{attr}**)** *We define* k_α *to be the smallest integer such that*

$$k_\alpha \geq \log_{\delta+\sigma} \left(\frac{\rho\sqrt{\alpha}}{2d} \right). \tag{35}$$

Furthermore, we define our safety back-off set as the set

$$\Xi_{\text{attr}} := \left\{ (x, w) \in \Xi \,\middle|\, V(x) \leq \alpha - k_\alpha \mu d^2 \right\}. \tag{36}$$

Figure 2 tries to clarify the appearing regions and the key ideas of the complete stability proof.

Theorem 6.1 (Increase of Objective, Decrease of Stepsize) *Assume that Assumptions 1-4 and 5 hold and that* $(x^0, w^0) \in \Xi_{\text{attr}}$. *Then for* $k = 0, \dots k_\alpha$ *it holds that* $(x^k, w^k) \in \Xi$. *Furthermore,*

$$\|\Delta w^{k_\alpha}\| \leq \rho\sqrt{\alpha}.$$

Proof: We make use of Corollary 5.2 and 5.3. To apply them, we first observe that $(x^0, w^0) \in \Xi$. It remains to be shown that $x^0, \dots, x^{k_\alpha} \in X_\alpha$. We do this inductively, and show: if for some $k \leq k_\alpha$ it holds that $(x^k, w^k) \in \Xi$ and $V(x^k) \leq \alpha + (k - k_\alpha)\mu d^2$ then also $(x^{k+1}, w^{k+1}) \in \Xi$ and $V(x^{k+1}) \leq \alpha + (k + 1 - k_\alpha)\mu d^2$. To show this we first note that $\|\Delta w^k\| \leq d$ as an immediate consequence of Corollary 5.2. Now from Theorem 4.2 we know that

$$V(x^{k+1}) \leq V(x^k) - L(x^k, u^k) + \mu d^2$$

from which we conclude

$$V(x^{k+1}) \leq V(x^k) + \mu d^2 \leq \alpha + (k - k_\alpha)\mu d^2 + \mu d^2 = \alpha + (k + 1 - k_\alpha)\mu d^2.$$

□

Remark: The restriction of the initial system state x^0 to the level set $\left\{ x \in X \,\middle|\, V(x) \leq \alpha - k_\alpha \mu d^2 \right\}$ is unnecessarily restrictive. On the one hand we neglected the decrease $-L(x^k, u^k)$ in each step; and on the other hand an initial stepsize $\|\Delta w(x^0, w^0)\|$ considerably smaller than d would allow the errors in the decrease condition be considerably smaller than μd^2. Note in particular that an initial iterate (x^0, w^0) where the optimizer is initialized so well that $\|\Delta w(x^0, w^0)\| \leq \rho\sqrt{\alpha}$ directly qualifies for Phase 2, if only $V(x^0) \leq \alpha$, without requiring any safety back-off at all. However, to keep the discussion as simple as possible, we chose to stick to our above definition of the set Ξ_{attr} of states attracted by the origin.

6.2 Phase 2: Convergence towards the Origin

We now show that the real-time iterations – once the errors have become small enough – not only remain in their level sets, but moreover, are attracted by even smaller level sets. For convenient formulation of the results of this subsection we first define two constant integers.

Definition 6.2 (k_1 and k_2) *Let us define the constants* k_1 *and* k_2 *to be the smallest integers that satisfy*

$$k_1 \geq \frac{6(M + m)}{m} \quad \text{and} \quad k_2 \geq \log_{\delta+\sigma}\left(\frac{1}{4}\right).$$

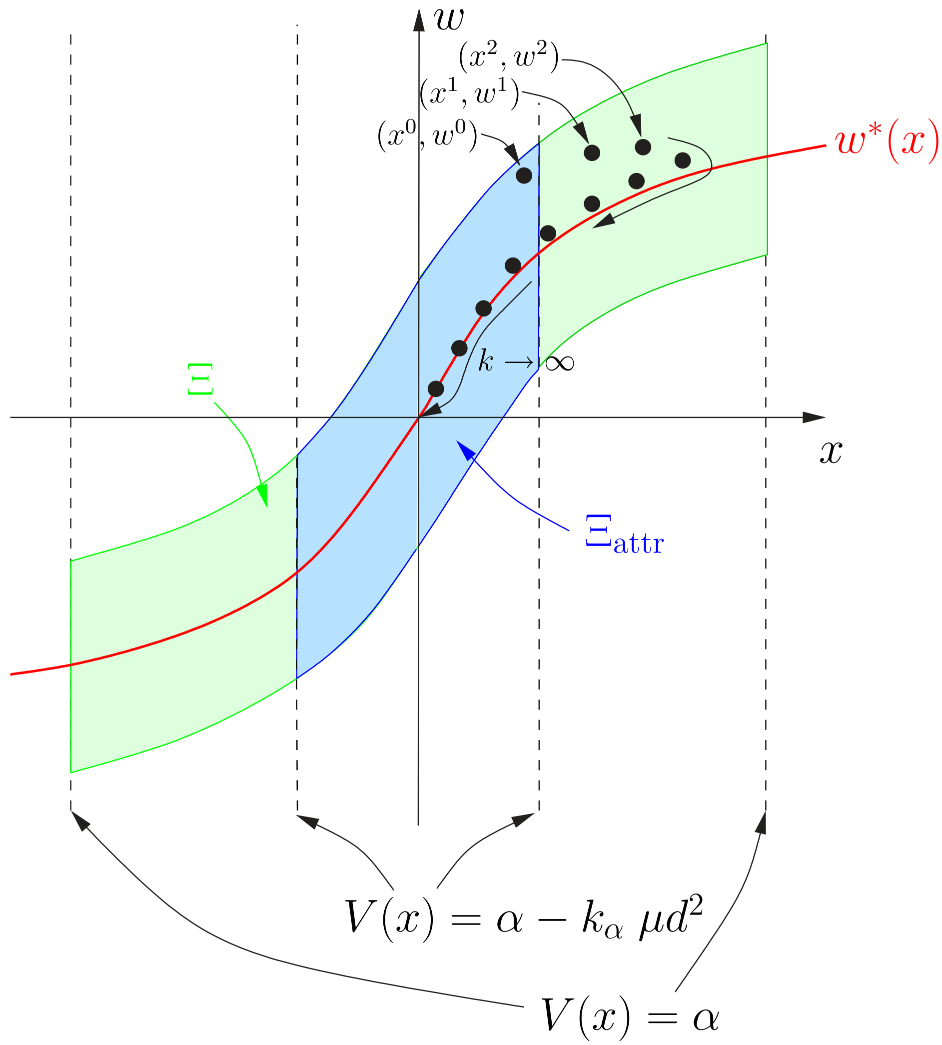

Figure 2: Illustration of the sets Ξ and Ξ_{attr} in the system-optimizeer space of variables (x, w), and visualization of the iterates during the two phases of the stability proof.

Theorem 6.2 (Objective and Stepsize Reduction)
Let in addition to Assumptions 1-5 assume that for an $\alpha_0 \le \alpha$ and a $k_0 \ge 0$ it holds that

$$V(x^{k_0}) \le \alpha_0 \quad \textit{and} \quad \|\Delta w^{k_0}\| \le \rho\sqrt{\alpha_0}.$$

Then all iterates $k \ge k_0$ are well-defined and also satisfy $V(x^k) \le \alpha_0$ and $\|\Delta w^k\| \le \rho\sqrt{\alpha_0}$. Moreover, for $k \ge k_0 + k_1 + k_2$

$$V(x^k) \le \frac{1}{4}\alpha_0 \quad \textit{and} \quad \|\Delta w^k\| \le \rho\sqrt{\frac{1}{4}\alpha_0}.$$

Proof: We prove the theorem in three steps: invariance of level sets, attractivity of a small level set for x^k and reduction of the Newton steps $\|\delta w^k\|$

Step 1: Well-definedness of all iterates and invariance of the level sets.
The proof is by induction. We assume that for some $k \ge k_0$ it holds that $V(x^k) \le \alpha_0$ and $\|\Delta w^k\| \le \rho\sqrt{\alpha_0}$. We will show that then the next real-time iterate is well-defined and remains in the level sets, i.e., $V(x^{k+1}) \le \alpha_0$ and $\|\Delta w^{k+1}\| \le \rho\sqrt{\alpha_0}$.

First note that by the definition of ρ in (32), by $\alpha_0 \le \alpha$, and by the left inequality of (28)

$$\|\Delta w^k\| \le \frac{2\eta}{\sqrt{m}(1-(\delta+\sigma))}\sqrt{\alpha} \le \frac{2\frac{1}{2}\sqrt{\frac{m}{\alpha}}(1-(\delta+\sigma))d}{\sqrt{m}(1-(\delta+\sigma))}\sqrt{\alpha} = d,$$

i.e., $(x^k, w^k) \in \Xi$ and the real-time iterate is well-defined. Now, due to Assumption 1 $\|x^k\| \le \sqrt{\frac{\alpha_0}{m}}$. By Lemma 5.1 we know that if $x^{k+1} \in X_\alpha$ then

$$\|\Delta w^{k+1}\| \le (\delta+\sigma)\|\Delta w^k\| + \eta\|x^k\| \le (\delta+\sigma)\rho\sqrt{\alpha_0} + \eta\sqrt{\frac{\alpha_0}{m}}$$

and therefore, using the definition of ρ in (32),

$$\|\Delta w^{k+1}\| \le \rho\sqrt{\alpha_0}\left((\delta+\sigma) + \frac{1}{2}(1-(\delta+\sigma))\right) = \rho\sqrt{\alpha_0}\frac{1+\delta+\sigma}{2} \le \rho\sqrt{\alpha_0}.$$

It remains to be shown that $x^{k+1} \in X_{\alpha_0} \subset X_\alpha$. To show this we first observe that due to the right inequality of (28) in Assumption 3 we have

$$\rho \le \sqrt{\frac{m}{8(M+m)\mu}}$$

and therefore

$$\epsilon(x^{k_0}, w^{k_0}) \le \mu\|\Delta w^{k_0}\|^2 \le \frac{m}{8(M+m)}\alpha_0 =: \epsilon_0. \tag{37}$$

By Theorem 4.2 we obtain $V(x^{k+1}) \le V(x^k) + \epsilon_0 - m\|x_k\|^2$. We now distinguish two cases:

a) $m\|x^k\|^2 \ge 2\epsilon_0$: we have $V(x^{k+1}) \le V(x^k) - \epsilon_0 \le \alpha_0 - \epsilon_0 \le \alpha_0$.

b) $m\|x^k\|^2 \le 2\epsilon_0$: because of $V(x^k) \le M\|x^k\|^2$ we have that $V(x^k) \le 2\frac{M}{m}\epsilon_0$ and therefore $V(x^{k+1}) \le 2\frac{M}{m}\epsilon_0 + \epsilon_0 = \frac{1}{4}\alpha_0 \le \alpha_0$ by the definition of ϵ_0 in (37).

This completes the first step of the proof.

Step 2: Attraction of the states x^k for $k \geq k_0 + k_1$ by the level set $X_{\frac{1}{4}\alpha_0}$.

We already showed that all iterates are well-defined and satisfy $V(x^k) \leq \alpha_0$ and $\|\Delta w^k\| \leq \rho\sqrt{\alpha_0}$, and furthermore, that $\epsilon(x^k, w^k) \leq \epsilon_0$.

To prove the stronger result that the states x^k are for $k \geq k_0 + k_1$ in the reduced level set $X_{\frac{1}{4}\alpha_0}$, we first show that once one iterate $x^{k'}$ is inside $X_{\frac{1}{4}\alpha_0}$, all following system states also remain inside. Again, we distinguish the two cases:

a) $m\|x^{k'}\|^2 \geq 2\epsilon_0$: we have $V(x^{k'+1}) \leq V(x^{k'}) - \epsilon_0 \leq V(x^{k'}) \leq \frac{1}{4}\alpha_0$

b) $m\|x^{k'}\|^2 \leq 2\epsilon_0$: as before, we have $V(x^{k+1}) \leq \frac{1}{4}\alpha_0$.

So let us see how many states x^k can at maximum remain outside $X_{\frac{1}{4}\alpha_0}$. First note that if $V(x^k) \geq \frac{1}{4}\alpha_0$ we also have $M\|x^k\|^2 \geq \frac{1}{4}\alpha_0 = 2\frac{M+m}{m}\epsilon_0 \geq 2\frac{M}{m}\epsilon_0$, i.e., $m\|x^k\|^2 \geq 2\epsilon_0$. Therefore, for each iterate that remains outside $X_{\frac{1}{4}\alpha_0}$, case a) holds, and $V(x^{k+1}) \leq V(x^k) - \epsilon_0$. We deduce that $V(x^{k_0+\Delta k}) \leq \alpha_0 - \Delta k \epsilon_0$, and therefore for $k \geq k_0 + \frac{6(M+m)}{m}$ that $V(x^k) \leq \alpha_0 - \frac{6(M+m)}{m}\epsilon_0 = \frac{1}{4}\alpha_0$ by definition (37).

Step 3: Reduction of the steps $\|\Delta w^k\|$ for $k \geq k_0 + k_1 + k_2$ under the level $\rho\sqrt{\frac{1}{4}\alpha_0}$.

We already know that all iterates $k \geq k_0 + k_1$ satisfy $V(x^k) \leq \frac{1}{4}\alpha_0$ and $\|\Delta w^k\| \leq \rho\sqrt{\alpha_0}$. We can now use Corollary 5.3 with $\|\Delta w^0\|$ replaced by $\rho\sqrt{\alpha_0}$, α_0 replaced by $\frac{1}{4}\alpha_0$, and k_0 replaced by $k - (k_0 + k_1)$, which yields the proposition:

$$\text{If} \quad k - (k_0 + k_1) \geq \log_{\delta+\sigma}\left(\frac{\rho\sqrt{\frac{1}{4}\alpha_0}}{2\rho\sqrt{\alpha_0}}\right) \quad \text{then} \quad \|\Delta w^k\| \leq \rho\sqrt{\frac{1}{2}\alpha_0}.$$

By definition of k_2 this implies $\|\Delta w^k\| \leq \rho\sqrt{\frac{1}{2}\alpha_0}$ for all $k \geq k_0 + k_1 + k_2$. □

Theorem 6.2 allows us to conclude that each $k_1 + k_2$ iterations, the level of the objective is reduced by a factor of $\frac{1}{4}$. This allows us to state the main result of this paper.

Theorem 6.3 (Nominal Stability of the Real-Time Iteration Scheme)
Let us suppose Assumptions 1-5 and assume that $(x^0, w^0) \in \Xi_{\text{attr}}$. Then all system-optimizer states are well-defined, i.e., satisfy $(x^k, w^k) \in \Xi$, and for all integers $p \geq 0$ and $k \geq k_\alpha + p(k_1 + k_2)$ it holds that $V(x^k) \leq \alpha\frac{1}{4^p}$ (respectively, $\|x^k\| \leq \sqrt{\frac{\alpha}{m}}\frac{1}{2^p}$) and $\|\Delta w^k\| \leq \rho\sqrt{\alpha}\frac{1}{2^p}$.

Proof: The theorem is an immediate consequence of Theorem 6.1 followed by an inductive application of Theorem 6.2. Furthermore, because $m\|x\|^2 \leq V(x)$, the inequality $V(x) \leq \frac{\alpha}{4^p}$ implies again $\|x\| \leq \sqrt{\frac{\alpha}{4^p m}}$. □

6.3 Discussion

From a practical point of view, the derived result can be interpreted as follows: whenever the system state is subject to a disturbance, but such that after the disturbance the combined system-optimizer state is in the region $\Xi_{\rm attr}$, the subsequent closed-loop response will lead the system towards the origin with a linear convergence rate, until another disturbance occurs. We would like to stress again, however, that the proof should not be seen as a construction rule for designing suitable real-time iteration schemes. Instead it gives a theoretical underpinning of the real-time iteration scheme.

Similar convergence results as for the real-time iteration scheme would also hold true for numerical schemes where more than one Newton type iteration is performed per sampling time, sacrificing, however, the instantaneous feedback of the real-time iteration scheme. In the limit of infinitely many iterations per optimization problem, the set $\Xi_{\rm attr}$ would approach the set Ξ and the whole region of attraction of the ideal NMPC controller would be recovered.

The result can in principle be expanded to other NMPC schemes without a zero terminal constraint. However, one should note that we assume that the value function is continuous. As is well known [MHER95, Fon00], NMPC can also stabilize systems that cannot be stabilized by feedback that is continuous in the state. This in general also implies a discontinuous value function. In this case the robustness properties utilized in Section 4.2 and used in Theorem 4.2 do not hold [KT02, GMTT03a, GMTT03b, SRM97] and further precautions must be taken.

7 Summary and Conclusions

We have presented a Newton type method for optimization in NMPC – the real-time iteration scheme with shift – and have proven nominal stability of the resulting system-optimizer dynamics. The scheme is characterized by a dovetailing of the dynamics of the system with those of the optimizer, resulting in an efficient online optimization algorithm which, however, shows intricate dynamics that do not allow to apply readily available standard stability results from NMPC.

The proof of nominal stability makes use of results from both, classical stability theory for NMPC as well as from convergence theory for Newton type optimization methods. Among several technical assumptions is one essential one (Assumption 3) that basically requires the disturbances in the optimization procedure – which are introduced by the movement of the optimization horizon – to be sufficiently small. We claim that this assumption can in practice always be satisfied by choosing a sufficiently long optimization horizon.

The proof of nominal stability gives a theoretical underpinning of the real-time iteration scheme that has already successfully been applied to several example systems, among them a real pilot-plant distillation column [DUF$^+$01, DFS$^+$03]. Experience has shown that in practice the real-time iteration scheme is able to bring the system-optimizer dynamics back into the region of attraction even after rather large disturbances (cf. [DBS03]).

Acknowledgements

Financial support by the *Deutsche Forschungsgemeinschaft (DFG)* within the priority program 469 "Online-Optimization of Large Scale Systems" is gratefully acknowledged, as well as support by the "Institute of Mathematics and its Applications" of the University of Minnesota (USA) which hosted the first author while parts of the paper have been developed.

References

[ABQ+99] F. Allgöwer, T.A. Badgwell, J.S. Qin, J.B. Rawlings, and S.J. Wright. Nonlinear predictive control and moving horizon estimation – An introductory overview. In P.M. Frank, editor, *Advances in Control, Highlights of ECC'99*, pages 391–449. Springer, 1999.

[BBB+01] T. Binder, L. Blank, H. G. Bock, R. Bulirsch, W. Dahmen, M. Diehl, T. Kronseder, W. Marquardt, J. P. Schlöder, and O. v. Stryk. Introduction to model based optimization of chemical processes on moving horizons. In M. Grötschel, S. O. Krumke, and J. Rambau, editors, *Online Optimization of Large Scale Systems: State of the Art*, pages 295–340. Springer, 2001. download at: http://www.zib.de/dfg-echtzeit/Publikationen/Preprints/Preprint-01-15.html.

[Bie00] L. T. Biegler. Efficient solution of dynamic optimization and NMPC problems. In F. Allgöwer and A. Zheng, editors, *Nonlinear Predictive Control*, volume 26 of *Progress in Systems Theory*, pages 219–244, Basel, 2000. Birkhäuser.

[Boc87] H. G. Bock. *Randwertproblemmethoden zur Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen*, volume 183 of *Bonner Mathematische Schriften*. University of Bonn, Bonn, 1987.

[BR91a] L. T. Biegler and J. B. Rawlings. Optimization approaches to nonlinear model predictive control. In Y. Arkun and W. H. Ray, editors, *Chemical Process Control – CPC IV*, pages 543–571, Austin, Texas, 1991. The CACHE Corp.

[BR91b] L.T. Biegler and J.B Rawlings. Optimization approaches to nonlinear model predictive control. In W.H. Ray and Y. Arkun, editors, *Proc. 4th International Conference on Chemical Process Control - CPC IV*, pages 543–571. AIChE, CACHE, 1991.

[BWB00] R. A. Bartlett, A. Wächter, and L. T. Biegler. Active set vs. inerior point strategies for model predicitve control. In *Proc. Amer. Contr. Conf.*, pages 4229–4233, Chicago, Il, 2000.

[CBO00] W. Chen, D.J. Ballance, and J. O'Reilly. Model predictive control of nonlinear systems: Computational delay and stability. *IEE Proceedings, Part D*, 147(4):387–394, 2000.

[DBS+02] M. Diehl, H. G. Bock, J. P. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer. Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. *J. Proc. Contr.*, 12(4):577–585, 2002.

[DBS03] M. Diehl, H. G. Bock, and J. P. Schlöder. Newton-type methods for the approximate solution of nonlinear programming problems in real-time. In G. Di Pillo and A. Murli, editors, *High Performance Algorithms and Software for Nonlinear Optimization*. Kluwer Academic Publishers B.V., 2003. in print.

[DFS+02] M. Diehl, R. Findeisen, S. Schwarzkopf, I. Uslu, F. Allgöwer, H. G. Bock, E. D. Gilles, and J. P. Schlöder. An efficient algorithm for nonlinear model predictive control of large-scale systems. Part I: Description of the method. *Automatisierungstechnik*, 12, 2002.

[DFS+03] M. Diehl, R. Findeisen, S. Schwarzkopf, I. Uslu, F. Allgöwer, H. G. Bock, E. D. Gilles, and J. P. Schlöder. An efficient algorithm for nonlinear model predictive control of large-scale systems. Part II: Application to a distillation column. *Automatisierungstechnik*, 1, 2003.

[Die02] M. Diehl. *Real-Time Optimization for Large Scale Nonlinear Processes*, volume 920 of *Fortschr.-Ber. VDI Reihe 8, Meß, Steuerungs- und Regelungstechnik*. VDI Verlag, Düsseldorf, 2002. Download also at: http://www.ub.uni-heidelberg.de/archiv/1659/.

[DMS00] G. De Nicolao, L. Magni, and R. Scattolini. Stability and robustness of nonlinear receding horizon control. In F. Allgöwer and A. Zheng, editors, *Nonlinear Predictive Control*, volume 26 of *Progress in Systems Theory*, pages 3–23, Basel, 2000. Birkhäuser.

[DUF+01] M. Diehl, I. Uslu, R. Findeisen, S. Schwarzkopf, F. Allgöwer, H. G. Bock, T. Bürner, E. D. Gilles, A. Kienle, J. P. Schlöder, and E. Stein. Real-time optimization for large scale processes: Nonlinear model predictive control of a high purity distillation column. In M. Grötschel, S. O. Krumke, and J. Rambau, editors, *Online Optimization of Large Scale Systems: State of the Art*, pages 363–384. Springer, 2001. download at: http://www.zib.de/dfg-echtzeit/Publikationen/Preprints/Preprint-01-16.html.

[FA03] R. Findeisen and F. Allgöwer. Computational delay in nonlinear model predictive control. Accepted for publication in *Proc. Int. Symp. Adv. Control of Chemical Processes, ADCHEM*, 2003.

[FDU+02] R. Findeisen, M. Diehl, I. Uslu, S. Schwarzkopf, F. Allgöwer, H.G. Bock, J.P. Schlöder, and Gilles. Computation and performance assesment of nonlinear model predictive control. In *Proc. 42th IEEE Conf. Decision Contr.*, Las Vegas, USA, 2002.

[FIAF02] R. Findeisen, L. Imsland, F. Allgöwer, and B.A. Foss. On the robustness of continuous time model predictive control. Workshop on Nonlinear Predictive Control, Oxford, UK, May 9th 2002.

[Fin03] R. Findeisen. Stability, computational efficiency, robustness, and output feedback in sampled-data nonlinear model predictive control. PhD thesis, University of Stuttgart, in preparation, 2003.

[Fon00] F.A. Fontes. A general framework to design stabilizing nonlinear model predictive controllers. *Syst. Contr. Lett.*, 42(2):127–143, 2000.

[GMTT03a] G. Grimm, M.J. Messina, A.R. Teel, and S. Tuna. Examples when model predictive control is nonrobust. submitted, 2003.

[GMTT03b] G. Grimm, M.J. Messina, A.R. Teel, and S. Tuna. Model predictive contro: for want of a local control lyapunov function, all is not lost. submitted, 2003.

[Hen98] M.A. Henson. Nonlinear model predictive control: Current status and future directions. *Comp. & Chem. Eng.*, 23:187–202, 1998.

[KT02] C.M. Kellett and A.R. Teel. On robustness of stability and lyapunov functions for discontinous difference equations. In *Proc. 42th IEEE Conf. Decision Contr.*, pages 4282–4287, Las Vegas, USA, 2002.

[LB89] W.C. Li and L.T. Biegler. Multistep, newton-type control strategies for constrained nonlinear processes. *Chem. Eng. Res. Des.*, 67:562–577, 1989.

[Lei99] D. B. Leineweber. *Efficient reduced SQP methods for the optimization of chemical processes described by large sparse DAE models*, volume 613 of *Fortschr.-Ber. VDI Reihe 3, Verfahrenstechnik*. VDI Verlag, Düsseldorf, 1999.

[LEL92] M. J. Liebman, T. F. Edgar, and L. S. Lasdon. Efficient data reconciliation and estimation for dynamic processes using nonlinear programming techniques. *Comp. & Chem. Eng.*, 16(10/11):963–986, 1992.

[MBF02] F. Martinsen, L.T. Biegler, and B.A Foss. Application of optimization algorithms to nonlinear mpc. In Proceedings of 15th IFAC World Congress, 2002.

[MHER95] E.S. Meadows, M.A. Henson, J.W. Eaton, and J.B. Rawlings. Receding horizon control and discontinuous state feedback stabilization. *Int. J. Contr.*, 62(5):1217–1229, 1995.

[MM90] D.Q. Mayne and H. Michalska. Receding horizon control of nonlinear systems. *IEEE Trans. Automat. Contr.*, 35(7):814–824, 1990.

[MRRS00] D.Q. Mayne, J.B. Rawlings, C.V. Rao, and P.O.M. Scokaert. Constrained model predictive control: stability and optimality. *Automatica*, 26(6):789–814, 2000.

[OB95a] N. M.C. de Oliveira and L. T. Biegler. Newton-type algorithms for nonlinear process control. algorithm and stability results. *Automatica*, 31(2):281–286, 1995.

[OB95b] N.M.C. de Oliveira and L.T. Biegler. An extension of Newton-type algorithms for nonlinear process control. *Automatica*, 31(2), 1995.

[Pyt99] R. Pytlak. *Numerical Methods for Optimal Control Problems with State Constraints*. Lecture Notes in Mathematics. Springer, Berlin, 1999.

[QB01] S.J. Qin and T.A. Badgwell. Review of nonlinear model predictive control applications. In B. Kouvaritakis and M. Cannon, editors, *Nonlinear model predictive control: theory and application*, pages 3–32, London, 2001. The Institute of Electrical Engineers.

[SRM97] P.O.M. Scokaert, J.B. Rawlings, and E.S. Meadows. Discrete-time stability with perturbations: Application to model predictive control. *Automatica*, 33(3):463–470, 1997.

[TR01] M. J. Tenny and J. B. Rawlings. Feasible real-time nonlinear model predictive control. In *6th International Conference on Chemical Process Control – CPC VI*, AIChE Symposium Series, 2001.

[TWR02] M.J. Tenny, S.J. Wright, and J.B. Rawlings. Nonlinear model predictive control via feasibility-pertubed sequetial quadratic programming. submitted, 2002.

[Wri96] S. J. Wright. Applying new optimization algorithms to model predictive control. In J.C. Kantor, C.E. Garcia, and B. Carnahan, editors, *Fifth International Conference on Chemical Process Control – CPC V*, pages 147–155. American Institute of Chemical Engineers, 1996.

Appendix: Proof of Theorem 4.2

In order to be able to prove Theorem 4.2, we will compare approximate solutions of the full problem $P(x)$ with those of a shrunk problem $\tilde{P}(f(x, \tilde{u}(x, w)))$, defined as follows.

Definition 7.1 **($\tilde{P}(x')$)**

$$\min_{\substack{s_1, \dots, s_N, \\ q_1, \dots, q_N}} \quad \sum_{i=1}^{N} L(s_i, q_i) \tag{38a}$$

subject to

$$x' - s_1 = 0, \tag{38b}$$

$$f(s_i, q_i) - s_{i+1} = 0, \quad i = 1, \dots, N-1, \tag{38c}$$

$$f(s_N, q_N) = 0. \tag{38d}$$

Let us also define the projection operator $\Pi : \mathbb{R}^n \to \mathbb{R}^{n-(2n_x+n_u)} = \mathbb{R}^{\tilde{n}}$

$$\Pi w = \Pi \begin{bmatrix} \lambda_0 \\ s_0 \\ q_0 \\ \hline \lambda_1 \\ s_1 \\ \vdots \\ \lambda_{N+1} \end{bmatrix} = \begin{bmatrix} \lambda_1 \\ s_1 \\ \vdots \\ \lambda_{N+1} \end{bmatrix} =: \tilde{w}$$

which simply removes the first components from $w \in \mathbb{R}^n$, to yield a vector $\tilde{w} \in \mathbb{R}^{\tilde{n}}$ in the primal-dual space of problem $\tilde{P}(\cdot)$, and we assume compatible norms in $\mathbb{R}^n$ and $\mathbb{R}^{\tilde{n}}$ in the sense that $\|\Pi w\| \leq \|w\|$ and $\|\Pi^T \tilde{w}\| = \|\tilde{w}\|$. Let us define the Lagrangian $\tilde{\mathcal{L}}_x(\tilde{w})$ of $\tilde{P}(x)$ in a straightforward way, and the corresponding second derivative approximation $\tilde{J}(\tilde{w})$, which can be shown to be $\tilde{J}(\tilde{w}) = \Pi J(\Pi^T \tilde{w}) \Pi^T$. We define the set

$$\tilde{X} := \{x' \in \mathbb{R}^{n_x} \,|\exists (x, w) \in \Xi : \; x' = f(x, q_0 + \Delta q_0(x, w))\}$$

and assume solvability of $\tilde{P}(x')$ for all $x' \in \tilde{X}$, and we define

$$\tilde{D}_{2C} := \left\{ \tilde{w} \in \mathbb{R}^{\tilde{n}} \,\middle|\, \exists x' \in \tilde{X} : \; \|\tilde{w} - \tilde{w}^*(x')\| \leq 2C \right\}$$

and make the analogon to Assumption 2 on $\tilde{D}_{2C}$, plus some additional technical assumptions.

Assumption 5 *We assume that the Lagrangian function $\tilde{\mathcal{L}}_x$ is twice continuously differentiable on $\tilde{D}_{2C}$ and that each solution $\tilde{w}^*(x)$ exists and is uniquely determined in $\tilde{D}_{2C}$, i.e.,*

$$\forall x \in \tilde{X}, \forall \tilde{w} \in \tilde{D}_{2C} \backslash \{\tilde{w}^*(x)\} : \nabla_{\tilde{w}} \tilde{\mathcal{L}}_x(\tilde{w}) \neq 0. \tag{39a}$$

We also assume that $\tilde{J}$ is continuous on $\tilde{D}_{2C}$ and that $\tilde{J}(\tilde{w})$ is invertible for all $\tilde{w} \in \tilde{D}_{2C}$, and that for all $\tilde{w}', \tilde{w} \in \tilde{D}_{2C}$, $\Delta\tilde{w} = \tilde{w}' - \tilde{w}$ and all $t \in [0,1]$ it holds that

$$\left\| \tilde{J}(\tilde{w}')^{-1} \left(\tilde{J}(\tilde{w} + t\Delta\tilde{w}) - \nabla^2_{\tilde{w}} \tilde{\mathcal{L}}(\tilde{w} + t\Delta\tilde{w}) \right) \Delta\tilde{w} \right\| \leq \kappa \|\Delta\tilde{w}\| \tag{39b}$$

and that

$$\left\| \tilde{J}(\tilde{w}')^{-1} \left(\tilde{J}(\tilde{w} + t\Delta\tilde{w}) - \tilde{J}(\tilde{w}) \right) \Delta\tilde{w} \right\| \leq \omega t \|\Delta\tilde{w}\|^2. \tag{39c}$$

Let us furthermore assume that for all $w', w \in D_{2C}$, $\Delta w = w' - w$ and all $t \in [0,1]$ it holds that

$$\left\| \tilde{J}(\Pi w')^{-1} \Pi \left(J(w + t\Delta w) - \nabla^2_w \mathcal{L}(w + t\Delta w) \right) \Delta w \right\| \leq \kappa \|\Delta w\| \tag{40a}$$

and that

$$\left\| \tilde{J}(\Pi w')^{-1} \Pi \left(J(w + t\Delta w) - J(w) \right) \Delta w \right\| \leq \omega t \|\Delta w\|^2. \tag{40b}$$

We also assume the following bound on the Hessian of the Lagrangian $\mathcal{L}.(w)$:

$$\|\nabla^2_w \mathcal{L}(w)\| \leq B, \quad \forall w \in D_{2C}. \tag{41}$$

By assumptions (39), we can guarantee numerical solvability of $\tilde{P}(x)$ by the Newton type scheme as in Corollary 3.3, if for some $\tilde{w}_0$ it holds that $\|\tilde{J}(\tilde{w}_0)^{-1} \nabla_{\tilde{w}} \tilde{\mathcal{L}}_x(\tilde{w}_0)\| \leq d$. Note that $\tilde{P}(x)$ needs never be solved in practice, but that this is only a hypothetical scheme which helps to establish the error bound.

Proof of Theorem 4.2 Now we are able to prove the theorem, with

$$\mu := 2B \left(\frac{\delta}{1-\delta} \right)^2. \tag{42}$$

We first define the shorthands $x' := f(x, q_0 + \Delta q_0(x,w))$ and $\Delta w := \Delta w(x,w)$. We will compare three vectors in $\mathbb{R}^n$:

- the solution $w^*(x)$ of $P(x)$,
- the first step $w' := w + \Delta w$ towards this solution, and
- an augmented version of the solution vector $\tilde{w}^*(x')$ of $\tilde{P}(x')$ defined as

  $$\tilde{w}^{*'} := \Pi^T \tilde{w}^*(x') + (\mathbb{I}_n - \Pi^T \Pi) w',$$

  so that, more intuitively, $\tilde{w}^{*'} = (\lambda_0', s_0', q_0', \tilde{\lambda}_1^*(x'), \tilde{s}_1^*(x'), \ldots)$.

We will show that all three vectors are in D_{2C}, and, to obtain a bound on the distance between $w^*(x)$ and $\tilde{w}^{*'}$ that

$$\|w^*(x) - w'\| \leq \frac{\delta \|\Delta w\|}{1-\delta}, \quad \text{and} \quad \|\tilde{w}^{*'} - w'\| \leq \frac{\delta \|\Delta w\|}{1-\delta}. \tag{43}$$

Clearly, the vector $w^*(x)$ is in D_{2C} and because $(x,w) \in \Xi$ the first step w' is also in D_{2C}. Furthermore, the left inequality for $w^*(x)$ was already proven in Corollary 3.3. For $\tilde{w}^{*'}$, we first note that $\|\tilde{w}^{*'} - w'\| = \|\Pi^T(\tilde{w}^*(x') - \Pi w')\| = \|\tilde{w}^*(x') - \Pi w'\|$. We consider hypothetical Newton type iterates $\tilde{w}_0, \tilde{w}_1, \ldots$ for solution of $\tilde{P}(x')$, started at the initial guess $\tilde{w}_0 := \Pi w'$. To show that these iterates are well-defined, let us first bound the size of the first step, $\Delta\tilde{w}_0 := \Delta\tilde{w}(x', \tilde{w}_0)$. Because

$$\begin{aligned} \nabla_{\tilde{w}}\tilde{\mathcal{L}}_{x'}(\tilde{w}_0) &= \nabla_{\tilde{w}}\tilde{\mathcal{L}}_{x'}(\Pi w') = \begin{bmatrix} x' - s'_1 \\ \vdots \end{bmatrix} = \begin{bmatrix} f(x, q_0 + \Delta q_0(x, w)) - s'_1 \\ \vdots \end{bmatrix} \\ &= \begin{bmatrix} f(s'_0, q'_0) - s'_1 \\ \vdots \end{bmatrix} = \Pi \nabla_w \mathcal{L}_x(w'), \end{aligned}$$

we have by adding $0 = \nabla_w \mathcal{L}_x(w) + J(w)\Delta w$ to the defining equation of $\Delta\tilde{w}_0$

$$\begin{aligned} -\Delta\tilde{w}_0 &= \tilde{J}(\tilde{w}_0)^{-1}\nabla_{\tilde{w}}\tilde{\mathcal{L}}_{x'}(\tilde{w}_0) \\ &= \tilde{J}(\tilde{w}_0)^{-1}\big(\nabla_{\tilde{w}}\tilde{\mathcal{L}}_{x'}(\tilde{w}_0) - \Pi(\nabla_w \mathcal{L}_x(w) + J(w)\Delta w)\big) \\ &= \tilde{J}(\tilde{w}_0)^{-1}\Pi\big(\nabla_w \mathcal{L}_x(w') - \nabla_w \mathcal{L}_x(w) - J(w)\Delta w\big). \end{aligned} \tag{44}$$

Therefore we can bound

$$\begin{aligned} \|\Delta\tilde{w}_0\| &= \|\tilde{J}(\tilde{w}_0)^{-1}\Pi\left(\nabla_w \mathcal{L}_x(w') - \nabla_w \mathcal{L}_x(w) - J(w)\Delta w\right)\| \\ &= \|\tilde{J}(\tilde{w}_0)^{-1}\Pi \textstyle\int_0^1 (\nabla_w^2 \mathcal{L}(w + t\Delta w) - J(w))\Delta w \, dt\| \\ &= \| \textstyle\int_0^1 \tilde{J}(\Pi w')^{-1}\Pi(\nabla_w^2 \mathcal{L}(w + t\Delta w) - J(w + t\Delta w))\Delta w \, dt \\ &\quad + \textstyle\int_0^1 \tilde{J}(\Pi w')^{-1}\Pi(J(w + t\Delta w) - J(w))\Delta w \, dt\| \\ &\leq \kappa\|\Delta w\| + \textstyle\int_0^1 \omega t \|\Delta w\|^2 \, dt = \left(\kappa + \frac{\omega}{2}\|\Delta w\|\right)\|\Delta w\| \leq \delta\|\Delta w\|, \end{aligned}$$

due to assumptions (40). After having established a bound on the first step $\Delta\tilde{w}_0$ of the hypothetical iterates, we conclude with assumptions (39) from the standard convergence result for Newton type iterates that the limit $\lim_{i\to\infty} \tilde{w}_i = \tilde{w}_*(x')$ satisfies

$$\|\tilde{w}_*(x') - \tilde{w}_0\| \leq \frac{\|\Delta\tilde{w}_0\|}{1-\delta} \leq \frac{\delta\|\Delta w\|}{1-\delta},$$

so that we have shown the right inequality of (43). With (43) we can now conclude that

$$\|\tilde{w}^{*'} - w^*(x)\| \leq 2\frac{\delta\|\Delta w\|}{1-\delta} \tag{45}$$

in particular that $\tilde{w}^{*'} \in D_{2C}$. We now compare the objective values of the two vectors $w^*(x)$ and $\tilde{w}^{*'}$. The objective contributions can be expressed in terms of the Lagrangian $\mathcal{L}_x(\cdot)$, because both, $w^*(x)$ and $\tilde{w}^{*'}$ are feasible points for $P(x)$:

$$V(x) = \sum_{i=0}^{N} L(s_i^*(x), q_i^*(x)) = \mathcal{L}_x(w^*(x))$$

and

$$L(x, q_0 + \Delta q_0(x, w)) + \tilde{V}(x') = L(s'_0, q'_0) + \sum_{i=1}^{N} L(\tilde{s}_i^*(x'), \tilde{q}_i^*(x')) = \mathcal{L}_x(w^{*'}).$$

Therefore, we can compare

$$\begin{aligned}
\|\mathcal{L}_x(\tilde{w}^{*'})-\mathcal{L}_x(w^*(x))\| &= \| \int_0^1 \nabla_w \mathcal{L}_x(w^*(x)+t_1(\tilde{w}^{*'}-w^*(x)))^T\,(\tilde{w}^{*'}-w^*(x))\,dt_1\| \\
&= \| \int_0^1\Big(\int_0^{t_1} \nabla_w^2 \mathcal{L}(w^*(x)+t_2(\tilde{w}^{*'}-w^*(x)))(\tilde{w}^{*'}-w^*(x))\,dt_2\Big)^T(\tilde{w}^{*'}-w^*(x))\,dt_1\| \\
&= \|(\tilde{w}^{*'}-w^*(x))^T\Big(\int_0^1\int_0^{t_1} \nabla_w^2 \mathcal{L}(w^{*0}+t_2(\tilde{w}^{*'}-w^*(x)))\,dt_2\,dt_1\Big)^T(\tilde{w}^{*'}-w^*(x))\| \\
&\leq \tfrac{B}{2}\|\tilde{w}^{*'}-w^*(x)\|^2,
\end{aligned}$$

where we have made use of the fact that $\nabla_w \mathcal{L}_x(w^*(x)) = 0$. Together with (45) we can now obtain the bound

$$L(x, q_0 + \Delta q_0(x, w)) + \tilde{V}(x') - V(x) \leq \frac{B}{2}\Big(2\frac{\delta\|\Delta w\|}{1-\delta}\Big)^2$$

and together with the property that $V(x') \leq \tilde{V}(x') = \sum_{i=1}^N L(\tilde{s}_i^*(x'), \tilde{q}_i^*(x'))$, as in the proof of Theorem 4.1, we immediately obtain the error bound of Theorem 4.2, with μ given by (42). ❑