Delft University of Technology

Low-complexity first-order constraint linearization methods for efficient nonlinear MPC

Torrisi, Giampaolo; Grammatico, Sergio; Frick, Damian; Robbiani, Tommaso; Smith, Roy S.; Morari, Manfred

DOI
10.1109/CDC.2017.8264304

Publication date
2017

Document Version
Final published version

Published in
Proceedings of the 2017 IEEE 56th Annual Conference on Decision and Control (CDC)

Citation (APA)
Torrisi, G., Grammatico, S., Frick, D., Robbiani, T., Smith, R. S., & Morari, M. (2017). Low-complexity first-order constraint linearization methods for efficient nonlinear MPC. In A. Astolfi et al (Ed.), *Proceedings of the 2017 IEEE 56th Annual Conference on Decision and Control (CDC)* (pp. 4376-4381). IEEE. https://doi.org/10.1109/CDC.2017.8264304

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

2017 IEEE 56th Annual Conference on Decision and Control (CDC)
December 12-15, 2017, Melbourne, Australia

Low-Complexity First-Order Constraint Linearization Methods for Efficient Nonlinear MPC

Giampaolo Torrisi, Sergio Grammatico, Damian Frick, Tommaso Robbiani, Roy S. Smith, Manfred Morari

Abstract— **In this paper, we analyze first-order methods to find a KKT point of the nonlinear optimization problems arising in Model Predictive Control (MPC). The methods are based on a projected gradient and constraint linearization approach, that is, every iteration is a gradient step, projected onto a linearization of the constraints around the current iterate.**

We introduce an approach that uses a simple ℓ_p merit function, which has the computational advantage of not requiring any estimate of the dual variables and keeping the penalty parameter bounded. We then prove global convergence of the proposed method to a KKT point of the nonlinear problem.

The first-order methods can be readily implemented in practice via the novel tool FalcOpt. The performance is then illustrated on numerical examples and compared with conventional methods.

I. Introduction

Several methods are available to solve nonlinear Model Predictive Control problems. The Sequential Quadratic Programming (SQP) method consists of iteratively solving Quadratic Programs (QPs) whose objective approximates the Lagrangian of the original problem around the current iterate and the constraints are obtained via linearization of the original nonlinear constraints. Under some technical assumptions [1], by using the solution to each QP to generate a new iterate, the algorithm converges to a Karush-Kuhn-Tucker (KKT) point of the original problem. The Gradient Descent method, on the other hand, considers only first-order information of the problem: the new iterate is obtained based on the local gradient of the objective function (gradient step) and is then projected onto the nonlinear constraint to guarantee feasibility. This projection step can be computationally expensive, unless the constraints are "simple" [2].

In line with previous work [3], [4], we consider a projected gradient and constraint linearization method, combining the properties of SQP and gradient descent. In particular, we take a gradient step and project it onto a linearization of the original nonlinear constraint, similarly to SQP. Thus, every intermediate iterate of the method need not be feasible for the original problem. By itself, the method is only locally convergent with a linear rate. Global convergence to a KKT point of the original problem is obtained by means of a merit function in the form of an augmented Lagrangian. This is an exact penalty function, i.e., for a sufficiently large value of a penalty parameter, the problem with augmented Lagrangian is equivalent to the original problem [5]. By showing that each iterate allows for a reduction of the merit function, we can prove convergence to a critical point. Note that the penalty parameter of the augmented Lagrangian may possibly become unbounded. This does not affect the convergence guarantee, but it may slow down the convergence.

In this work, the global convergence of the proposed projected gradient and constraint linearization method is instead proved by means of a simplified ℓ_p merit function. The ℓ_p merit function, with $p \geq 1$, has been long established as an exact penalty function and used in the SQP method [6], [7] or in convexification approaches [8]. This gives two main advantages compared to previous work. First, the dual variables need not be estimated, which reduces the overall number of variables in the problem and the computational burden for their evaluation. In addition, we are still able to (inexpensively) derive the dual optimum at the solution. Secondly, it is possible to prove that the penalty parameter in the ℓ_p merit function remains bounded.

II. The optimization algorithm

We consider an equality constrained nonlinear optimization problem (NLP)

$$\min_{z \in \mathbb{R}^n} J(z) \quad \text{s.t. } h(z) = 0 \tag{1}$$

where the functions $J : \mathbb{R}^n \to \mathbb{R}$ and $h : \mathbb{R}^n \to \mathbb{R}^m$ are twice continuously differentiable. For an approach that also includes inequality constraints, we refer to [9, Ch. 2 and 3].

We define the Lagrangian function of the NLP as

$$\mathcal{L}(z, \lambda, \nu) := J(z) + h(z)^\top \nu,$$

with Lagrange multiplier vector $\nu \in \mathbb{R}^m$.

We call $z^\star$ a critical point of (1) if it satisfies the first-order conditions with strict complementarity, i.e., there exists $\nu^\star \in \mathbb{R}^m$ such that

$$\nabla \mathcal{L}(z^\star, \lambda^\star, \nu^\star) = \nabla J(z^\star) + \nabla h(z^\star) \nu^\star = 0, \; h(z^\star) = 0. \tag{2}$$

We assume that the NLP in (1) has a finite number of critical points. In order to determine a critical point $z^\star$, iterative methods generate a sequence $\left(z^{(i)}\right)_{i=1}^{\infty}$. In this paper, we generate such sequences using first-order methods, that consist of taking projected gradient steps onto a linearization of the constraint around the current iterate $z^{(i)}$.

G. Torrisi, D. Frick, T. Robbiani and R.S. Smith are with the Automatic Control Laboratory, ETH Zurich, Switzerland. S. Grammatico is with the Delft Center for Systems and Control, TU Delft, The Netherlands. M. Morari is with the Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA. Email addresses: {torrisig, dafrick, tommasro, rsmith}@ethz.ch, s.grammatico@tudelft.nl, morari@seas.upenn.edu.

978-1-5090-2873-3/17/$31.00 ©2017 IEEE

Authorized licensed use limited to: TU Delft Library. Downloaded on February 11,2021 at 11:17:06 UTC from IEEE Xplore. Restrictions apply.

Given $z^{(i)}$, we define the next iterate $z^{(i+1)}$ as

$$z^{(i+1)} := z^{(i)} + t^{(i)} d_{\mathrm{z}}^{(i)}, \tag{3}$$

where the update $d_{\mathrm{z}}^{(i)}$ is determined through a gradient step that is projected onto the linearization of the constraints around $z^{(i)}$, and the variable $t^{(i)} \in (0, 1]$ is a penalty function step size. Formally,

$$d_{\mathrm{z}}^{(i)} := \Pi_{\mathcal{C}^{(i)}} \left(-\alpha^{(i)} \nabla J(z^{(i)}) \right), \tag{4}$$

with bounded gradient step size $\alpha^{(i)} \in \mathbb{R}_{>0}$, and $\Pi_{\mathcal{C}^{(i)}}(\cdot) : \mathbb{R}^n \to \mathcal{C}^{(i)} \subseteq \mathbb{R}^n$ being the Euclidean projection onto the set

$$\mathcal{C}^{(i)} := \left\{ d_{\mathrm{z}} \in \mathbb{R}^n \;\middle|\; h(z^{(i)}) + \nabla h(z^{(i)})^\top d_{\mathrm{z}} = 0 \right\}. \tag{5}$$

By definition of projection, it follows from (4) that:

$$\begin{aligned} d_{\mathrm{z}}^{(i)} = \underset{d_{\mathrm{z}} \in \mathbb{R}^n}{\arg\min} \quad & \frac{1}{2\alpha^{(i)}} \left\| d_{\mathrm{z}} + \alpha^{(i)} \nabla J(z^{(i)}) \right\|_2^2 \\ \text{s.t.} \quad & h(z^{(i)}) + \nabla h(z^{(i)})^\top d_{\mathrm{z}} = 0. \end{aligned} \tag{6}$$

For specific constraints h, the QP in (6) can be solved analytically. This will be used in Section III. By the KKT conditions for (6), under the assumptions below there exist dual multipliers $\nu_{\mathrm{G}}^{(i)} \in \mathbb{R}^m$ such that

$$\frac{1}{\alpha^{(i)}} d_{\mathrm{z}}^{(i)} + \nabla J(z^{(i)}) + \nabla h(z^{(i)}) \nu_{\mathrm{G}}^{(i)} = 0 \tag{7a}$$

$$h(z^{(i)}) + \nabla h(z^{(i)})^\top d_{\mathrm{z}}^{(i)} = 0. \tag{7b}$$

Throughout the paper, let us adopt the following basic regularity assumptions as in [1], [10].

Assumption 1: For all $i \in \mathbb{N}$, the matrix $\nabla h(z^{(i)})$ has full column rank.

Note that this implies that the dual variables $\nu_{\mathrm{G}}^{(i)}$ in (7) are bounded and unique.

Assumption 2: For all $i \in \mathbb{N}$, $z^{(i)}$, $z^{(i)} + d_{\mathrm{z}}^{(i)} \in \Omega \subset \mathbb{R}^n$ for some compact set Ω.

Assumption 3: The functions J, h, and their first and second derivatives are uniformly bounded in norm in Ω.

Under these assumptions, local convergence of the iteration in (4) when $t^{(i)} = 1$ for all i holds true.

Proposition 1 ([3]): Assume that $(z^\star, \nu^\star)$ is a critical point such that $\nabla^2 \mathcal{L}(z^\star, \nu^\star) \succ 0$, and let the initialization $z^{(0)}$ be close enough to $z^\star$. Then, there exist positive step sizes $(\alpha^{(i)})_i$ such that the sequence $\left(z^{(i)}\right)_i$ defined as in (3) with $t^{(i)} = 1$ converges to $z^\star$ with linear rate. □

Since the proposed method is first-order, we do not expect a convergence rate faster than linear. We will also utilize Assumptions 2.1, 2.2 and a relaxed version of Assumption 2.3 in Section II-B, where we establish global convergence.

Next, we establish global convergence to a critical point. In Section II-A, we review the available results [3]. In Section II-B, we introduce an ℓ_p merit function and provide the corresponding theoretical guarantees.

A. Global convergence via augmented Lagrangian function

To obtain global convergence to a critical point $z^\star$, a non-unit step size $t^{(i)}$ should be used. A common approach to determine such a step size is to introduce a penalty function that weights the optimality and feasibility of each iterate $z^{(i)}$. Note, in fact, that for general nonlinear constraints, each iterate need not be feasible for the original problem.

Along with the primal sequence $\left(z^{(i)}\right)_i$ we also consider the sequence of dual variables $\left(\nu^{(i)}\right)_i$, updated as

$$\begin{bmatrix} z^{(i+1)} \\ \nu^{(i+1)} \end{bmatrix} = \begin{bmatrix} z^{(i)} \\ \nu^{(i)} \end{bmatrix} + t^{(i)} \begin{bmatrix} d_{\mathrm{z}}^{(i)} \\ d_{\nu}^{(i)} \end{bmatrix}, \tag{8}$$

where

$$d_{\nu}^{(i)} := \nu_{\mathrm{G}}^{(i)} - \nu^{(i)}. \tag{9}$$

We determine the step size $t^{(i)}$ and the update in (8) via a line search on a merit function. First, we consider the augmented Lagrangian:

$$\mathcal{L}_{\text{aug}}(z, \nu, \rho) := \; J(z) + h(z)^\top \nu + \tfrac{\rho}{2} \left\| h(z) \right\|_2^2. \tag{10}$$

The penalty parameter $\rho > 0$ has a critical role in this algorithm: in fact, there exists $\rho^\star > 0$ such that the merit function in (10) is an exact penalty function for every $\rho > \rho^\star$ [5]. By defining the penalty functions $\phi(t) := \mathcal{L}_{\text{aug}}(z + \; t d_{\mathrm{z}}, \nu + t d_\nu, \rho)$, we choose the step size $t^{(i)} \in (0, 1]$ such that the following Armijo condition holds:

$$\phi(t^{(i)}) \leq \phi(0) + \sigma t^{(i)} \phi'(0), \tag{11}$$

where $\sigma \in (0, 1)$ and $\phi'(\cdot)$ is the derivative of $\phi(\cdot)$. We derive the value of t by safeguarded quadratic interpolation [11]. Should a tentative step size $\tilde{t}$ not satisfy (11), a new $t \in \left[\tau_1 \tilde{t}, \tau_2 \tilde{t}\right]$ is chosen, for some fixed $0 < \tau_1 < \tau_2 < 1$.

To satisfy the convergence conditions, at each iteration the penalty parameter $\rho^{(i)}$ is chosen such that:

$$\phi'(0) \leq -\frac{1}{2\alpha^{(i)}} \left\| d_{\mathrm{z}}^{(i)} \right\|_2^2, \tag{12}$$

which implies that the gradient step $d_{\mathrm{z}}^{(i)}$ is a descent direction for the augmented Lagrangian. The following Lemma gives an update rule for the penalty parameter $\rho^{(i)}$.

Lemma 2 ([3]): If $\rho^{(i)} \geq \hat{\rho}^{(i)} := 2 \left\| d_\nu^{(i)} \right\|_2 / \left\| h(z^{(i)}) \right\|_2$, then (12) holds for all $i \in \mathbb{N}$. □

Therefore, if we define the penalty parameter at the beginning of each iteration i as

$$\rho^{(i)} := \begin{cases} \rho^{(i-1)} & \text{if (12) holds} \\ \max\{\hat{\rho}^{(i)}, \, 2\rho^{(i-1)}\} & \text{otherwise,} \end{cases} \tag{13}$$

then the algorithm converges as stated next.

Proposition 3 ([3]): The primal and dual iterates in Algorithm 1 converge to the KKT triple associated to a critical point $z^\star$ of (1), i.e., $\lim_{i \to \infty} \left\| z^{(i)} - z^\star \right\|_2 = \lim_{i \to \infty} \left\| \nu^{(i)} - \nu^\star \right\|_2 = 0$. □

The resulting approach is summarized in Algorithm 1.

Authorized licensed use limited to: TU Delft Library. Downloaded on February 11,2021 at 11:17:06 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1 First-order augmented-Lagrangian algorithm

```
INITIALIZE $i \leftarrow 0$ and $z^{(0)} \in \mathbb{R}^n$
repeat
    COMPUTE $d_{\mathrm{z}}^{(i)}$ with step size $\alpha^{(i)}$ as in (4)
    DETERMINE $\nu_{\mathrm{G}}^{(i)}$ such that (7) holds
    if $d_{\mathrm{z}}^{(i)} = 0$ then
        SET $z^\star = z^{(i)}$, $\nu^\star = \nu_{\mathrm{G}}^{(i)}$ and STOP
    else
        if $i = 0$ then
            SET $\nu^{(0)} = \nu_{\mathrm{G}}^{(0)}$
        end if
        SET $d_\nu^{(i)} = \nu_{\mathrm{G}}^{(i)} - \nu^{(i)}$
    end if
    SET $\rho^{(i)}$ as in (13)
    DETERMINE the step size $t^{(i)}$ that satisfies (11), e.g. via line search
    UPDATE $z^{(i+1)}, \nu^{(i+1)}$ as in (8)
    $i \leftarrow i + 1$
until Convergence
return $z^\star$ and $\nu^\star$
```

B. Global convergence via ℓ_p merit function

In this section, we introduce an alternative approach that (i) does not require a sequence of dual variables and (ii) keeps the penalty parameter ρ bounded. Further, Assumption 3 is weakened as follows.

Assumption 4: The function J is uniformly bounded in Ω.

Note that the assumption is satisfied whenever J is continuous. Then, we make use of the following ℓ_p merit function:

$$\gamma_p(z, \rho) := J(z) + \rho \left\| h(z) \right\|_p , \tag{14}$$

with $p \geq 1$ (typically either 1, 2 or ∞), which is known to be an exact penalty function [7], [12].

By defining the function $\phi(t) := \gamma_p(z + t d_{\mathrm{z}}, \rho)$, we choose the penalty function step size $t^{(i)} \in (0, 1]$ that satisfies the following Armijo condition:

$$\phi(t^{(i)}) \leq \phi(0) + \sigma t^{(i)} D\phi(0), \tag{15}$$

where $\sigma \in (0, 1)$ and $D\phi(\cdot)$ is the directional derivative,

$$D\phi(0) = \lim_{t \to 0^+} \frac{\phi(t) - \phi(0)}{t}.$$

Analogously to (11), we determine the step size t via safeguarded quadratic interpolation, such that if a tentative step $\tilde{t}$ fails, then a new $t \in [\tau_1 \tilde{t}, \tau_2 \tilde{t}]$ is tested.

With the merit function in (14), the penalty parameter ρ can be chosen according to the following tuning rule, where we define q such that $\frac{1}{p} + \frac{1}{q} = 1$.

Lemma 4: If $\rho^{(i)} > \hat{\rho}^{(i)} := \left\| \nu_{\mathrm{G}}^{(i)} \right\|_q$ then

$$D\phi(0) \leq -\frac{1}{\alpha^{(i)}} \left\| d_{\mathrm{z}}^{(i)} \right\|_2^2 .$$

If $d_{\mathrm{z}}^{(i)} = 0$, then the algorithm has converged to a KKT point $(z^\star, \nu^\star)$. □

Algorithm 2 First-order algorithm with ℓ_p merit function

```
INITIALIZE $i \leftarrow 0$ and $z^{(0)} \in \mathbb{R}^n$
repeat
    COMPUTE $d_{\mathrm{z}}^{(i)}$ with step size $\alpha^{(i)}$ as in (4)
    DETERMINE $\nu_{\mathrm{G}}^{(i)}$ such that (7) holds
    if $d_{\mathrm{z}}^{(i)} = 0$ then
        SET $z^\star = z^{(i)}$, $\nu^\star = \nu_{\mathrm{G}}^{(i)}$ and STOP
    end if
    SET $\rho^{(i)} \geq 0$ as in (17)
    DETERMINE the step size $t^{(i)}$ that satisfies (15), e.g. via line search
    UPDATE $z^{(i+1)}$ as in (3)
    $i \leftarrow i + 1$
until Convergence
return $z^\star$ and $\nu^\star$
```

Proof: For ease of notation we omit the iteration index i. The directional derivative of (14) is [7, p. 115]:

$$D\phi(0) = \; = d_{\mathrm{z}}^\top \nabla J(z) - \rho \left\| h(z) \right\|_p .$$

By (7a), (7b) and the Cauchy-Schwarz inequality $a^\top b \leq \left\| a \right\|_p \left\| b \right\|_q$, we have:

$$\begin{aligned} D\phi(0) &= -\frac{1}{\alpha} d_{\mathrm{z}}^\top d_{\mathrm{z}} - d_{\mathrm{z}}^\top \nabla h(z) \nu_{\mathrm{G}} - \rho \left\| h(z) \right\|_p \\ &= -\frac{1}{\alpha} d_{\mathrm{z}}^\top d_{\mathrm{z}} + h(z)^\top \nu_{\mathrm{G}} - \rho \left\| h(z) \right\|_p \\ &\leq -\frac{1}{\alpha} d_{\mathrm{z}}^\top d_{\mathrm{z}} + \Big(\left\| \nu_{\mathrm{G}} \right\|_q - \rho \Big) \left\| h(z) \right\|_p , \end{aligned} \tag{16}$$

from which the first statement of the Lemma follows.

If $d_{\mathrm{z}} = 0$, from (7) and (2), then we have that the current z is a critical point for (1) with dual multiplier ν_{G}. ■

As summarized in Algorithm 2, we can update $\rho^{(i)}$ as

$$\rho^{(i)} := \begin{cases} \rho^{(i-1)} & \text{if } \rho^{(i-1)} \geq \hat{\rho}^{(i)} \\ \hat{\rho}^{(i)} + \varepsilon_1 & \text{otherwise,} \end{cases} \tag{17}$$

with $\varepsilon_1 > 0$. The next Lemma shows that, in a neighborhood of the optimum, the penalty parameter $\rho^{(i)}$ is bounded.

Lemma 5: For all $z^\star \in \mathbb{R}^n$, there exist $\rho^\star, \varepsilon^\star > 0$ such that $D\phi(0) \leq -\frac{1}{\alpha} \left\| d_{\mathrm{z}} \right\|_2^2$ for all $\rho \geq \rho^\star$ and $z \in z^\star + \varepsilon \mathbb{B}$, with $\mathbb{B}$ being the unit ball centered at the origin. □

Proof: Assumption 1 implies that the dual variable ν_G is bounded. By continuity, there exists an $\varepsilon(z^\star) > 0$ and $d(z^\star) > 0$ such that $\left\| \nu_{\mathrm{G}} \right\|_q \leq d(z^\star)$ for all $z \in z^\star + \varepsilon \mathbb{B}$. Thus, by (16), $\rho^\star = d(z^\star)$ and the procedure in (17) increases $\rho^{(i)}$ by at least ε_1 only a finite number of times. ■

Lemma 6: Let $\tilde{i}$ be such that $\rho^{(i)} = \rho^\star$ for all $i \geq \tilde{i}$. Then there exists $\bar{t} = \bar{t}(\rho^\star) > 0$ such that (15) is satisfied. □

Proof: For ease of notation we omit the iteration index i. By the continuity of the merit function in (14), we know that there exists a sufficiently small $t^{(i)} \leq 1$

Authorized licensed use limited to: TU Delft Library. Downloaded on February 11,2021 at 11:17:06 UTC from IEEE Xplore. Restrictions apply.

that satisfies the Armijo condition in (15). Suppose that the Armijo test failed for $\tilde{t}$, i.e.,

$$\phi\left(\tilde{t}\right) - \phi(0) > \sigma\, \tilde{t} D\phi(0),$$

and that $\tau_1\, \tilde{t} \leq t^{(i)}$. On the other hand, by expanding $\phi\left(\tilde{t}\right)$ to the second order, we have:

$$\phi\left(\tilde{t}\right) - \phi(0) \leq \tilde{t} D\phi(0) + \left(\tilde{t}\right)^2 b \left\|d_{\mathrm{z}}\right\|_2^2$$

with $b = b(\rho^{(i)}) > 0$.

It follows from the two latter inequalities that

$$(\sigma - 1)\, \tilde{t}\; D\phi(0) < \left(\tilde{t}\right)^2 b \left\|d_{\mathrm{z}}\right\|_2^2$$

which implies, since $\sigma < 1$, that $D\phi(0) > \frac{\tilde{t}\, b \|d_{\mathrm{z}}\|_2^2}{\sigma - 1}$. By Lemma 4, we have that $D\phi(0) \leq -\frac{1}{\alpha}\left\|d_{\mathrm{z}}\right\|_2^2$, hence, due to the last inequality, we have that $\tilde{t}\, b > \frac{1-\sigma}{\alpha} > 0$. Since $\rho^{(i)} = \rho^\star$ for sufficiently large $i \geq \tilde{i}$, $b = b(\rho^\star)$ must be bounded, hence $\tilde{t}$ is always strictly greater than zero. Finally, we can derive the lower bound $\bar{t} = \frac{\tau_1(1-\sigma)}{\alpha\, b}$ for $t^{(i)} \geq \tau_1 \tilde{t}$. ■

Theorem 7: The primal iterates in Algorithm 2 converge to the KKT point associated to a critical point $z^\star$ of (1), i.e., $\lim_{i\to\infty}\left\|z^{(i)} - z^\star\right\|_2 = 0$. Further, it holds that $\lim_{i\to\infty}\left\|\nu_{\mathrm{G}}^{(i)} - \nu^\star\right\|_2 = 0$. □

Proof: The proof consists of showing that $\lim_{i\to\infty}\left\|d_{\mathrm{z}}^{(i)}\right\|_2 = 0$. The statement of the theorem then follows by comparing (7) and (2). For the sake of contradiction, suppose that there exists an $\varepsilon > 0$ and $\tilde{i} \in \mathbb{N}$ such that $\left\|d_{\mathrm{z}}^{(i)}\right\|_2 > \varepsilon$ for all $i \geq \tilde{i}$. From Lemma 5, there exists a sufficiently large i such that $\rho^{(i)}$ is bounded. Without loss of generality, we consider that $\tilde{i} \in \mathbb{N}$ has such a property. Then, in every iteration $i \geq \tilde{i}$ the merit function decreases, due to Lemma 4, and in particular:

$$\begin{aligned}\phi(t^{(i)}) - \phi(0) &\leq \sigma t^{(i)} D\phi(0) \leq -\frac{1}{\alpha^{(i)}}\sigma\bar{t}\left\|d_{\mathrm{z}}^{(i)}\right\|_2^2\\ &\leq -\frac{1}{\alpha^{(i)}}\sigma\bar{t}\varepsilon^2 < 0,\end{aligned}$$

since $\alpha^{(i)}$ is designed to be bounded (e.g., $\alpha^{(i)} = 1\ \forall i$). This implies that the merit function decreases by a finite amount at every iteration, thus is unbounded from below. However, by Assumption 4, this leads to a contradiction, hence the proof follows. ■

Remark: The treatment in [6] (for an l_1 merit function) establishes a weaker result under less restrictive assumptions. In particular, under Assumptions 1 and 2, the existence of a convergent subsequence of the iterates is proved. In MPC problems, the objective function is typically positive semi-definite, hence Assumption 4 is satified. □

In Algorithm 1, full-size steps $t^{(i)} = 1$ are asymptotically allowed, thus recovering the linear convergence rate of Theorem 1 [3]. On the other hand, this does not necessarily hold for ℓ_p merit functions, where the Maratos effect can occur, i.e., only small steps $t^{(i)} < 1$ may be allowed close to the solution, thus requiring more iterates than expected [13]. In contrast to SQP methods, this is a small practical disadvantage of our method, since each iteration is computationally cheap (see Section III). An increased number of iterations is in practice preferable to typical SQP measures such as the second-order corrections or the watchdog technique [14].

III. Practical Implementation

A. Inequality constraints via slack variables

An inequality constrainted problem can be equivalently reformulated as the equality constrained problem in (1) via squared-slack variables y [3], [11]:

$$\min_{(z,y)\in\mathbb{R}^n\times\mathbb{R}^m} J(z) \quad \text{s.t. } g(z) + \frac{1}{2}\mathrm{diag}(y)\, y = 0.$$

Remarkably, in this case the projection in (4) admits a closed form solution. In fact, we can determine the dual variable ν_{G} as the solution of the dual problem:

$$\begin{aligned}\nu_{\mathrm{G}} = \left(\nabla g(z)^\top \nabla g(z) + \mathrm{diag}(y)^2\right)^{-1}\cdot&\\ \left(\frac{1}{\alpha}g(z) + \frac{1}{2\alpha}\mathrm{diag}(y)y - \nabla g(z)^\top \nabla J(z)\right)&. \quad (18)\end{aligned}$$

Then, the primal solution is given by

$$\begin{bmatrix} d_{\mathrm{z}} \\ d_{\mathrm{y}} \end{bmatrix} = \begin{bmatrix} -\alpha\nabla J(z) \\ 0 \end{bmatrix} - \alpha \begin{bmatrix} \nabla g(z) \\ \mathrm{diag}(y) \end{bmatrix} \nu_{\mathrm{G}} \qquad (19)$$

and the dual increments d_ν follow from (9). By the LICQ assumption on the active inequality constraints, the resulting equality constraints satisfy Assumption 1, thus the matrix $\nabla g(z)^\top \nabla g(z) + \mathrm{diag}(y)^2$ in (18) is invertible.

In general, inverting the matrix in (18) has a computational complexity that is cubic with respect to the matrix size m. In the next section, we focus on sparse MPC problems for which such inversion can be computed inexpensively.

B. Nonlinear Model Predictive Control problems

Let us apply the proposed Algorithms 1 and 2 to nonlinear MPC problems. In particular, we aim to derive the projection in (4) without numerically solving the associated QP in (6). The general form of the problem that we wish to solve is:

$$\begin{aligned}\min_{\{x_{k+1},u_k\}_{k=0}^{N-1}} \ & \tfrac{1}{2}\sum_{k=0}^{N-1}\left\{\left\|x_k - \bar{x}_k\right\|_Q^2 + \left\|u_k - \bar{u}_k\right\|_R^2\right\}\\ & + \tfrac{1}{2}\left\|x_N - \bar{x}_N\right\|_P^2\\ \text{s.t.} \quad & x_{k+1} = f\left(x_k, u_k\right)\\ & a_k \leq u_k \leq b_k\\ & n(u_k) \leq 0 \quad \forall k \in \{0, \ldots, N-1\},\\ & \tfrac{1}{2}\|x_{\tilde{k}} - \bar{x}_{\tilde{k}}\|_P^2 \leq c,\end{aligned} \qquad (20)$$

where the cost matrices satisfy $Q, R, P \succeq 0$, f is the discrete-time model of the dynamics, $a_k < b_k$ componentwise, $n : \mathbb{R}^{n_{\mathrm{u}}} \to \mathbb{R}^{n_{\mathrm{n}}}$ is a nonlinear constraint on the input, and the last constraint is a stability constraint with $c > 0$ and $\tilde{k} \in \{1, \ldots, N\}$ (e.g., terminal constraint if $\tilde{k} = N$, or a contractive constraint for time-varying $\tilde{k}$ [15], [16]).

Authorized licensed use limited to: TU Delft Library. Downloaded on February 11,2021 at 11:17:06 UTC from IEEE Xplore. Restrictions apply.

The algorithm step in (4) can be computed in closed form by appropriately recasting the problem. For ease of presentation, we do not consider the nonlinear constraint $n(u_k) \leq 0$. An analogous treatment applies otherwise by augmenting the system. We define the vectors $\mathbf{u} := [u_0; \dots; u_{N-1}]$ for the control input sequence, $\bar{\mathbf{u}}$ for the input reference and $\Delta\mathbf{u} := \mathbf{u} - \bar{\mathbf{u}}$. Analogously, the corresponding state evolution is $\mathbf{x} := [x_1; \dots; x_N]$, the state reference is $\bar{\mathbf{x}}$, and $\Delta\mathbf{x} := \mathbf{x} - \bar{\mathbf{x}}$. We recast the dynamics in a compact form as $\mathbf{x} = \psi(\mathbf{u})$, where for a fixed initial state x_0, the function $\psi : \mathbb{R}^{Nn_\text{u}} \to \mathbb{R}^{Nn_\text{x}}$ maps the sequence of inputs $\mathbf{u}$ to the predicted sequence of states $\mathbf{x}$ according to the nonlinear dynamics $x_{k+1} = f(x_k, u_k)$. The input bounds are given by $\mathbf{a} := [a_0; \dots; a_{N-1}]$ and $\mathbf{b} := [b_0; \dots; b_{N-1}]$ and we stack the state and input cost matrices $\mathcal{Q} = \text{blockdiag}\,(Q, \dots, Q, P)$ and $\mathcal{R} = \text{blockdiag}\,(R, \dots, R)$. Thus, by including the nonlinear dynamics within the objective and by adding the nonlinear slacks $\mathbf{y}_\text{a}, \mathbf{y}_\text{b} \in \mathbb{R}^{Nn_\text{u}}$ and $y_\text{c} \in \mathbb{R}$, the MPC problem in (20) is equivalent to

$$\begin{aligned} \min_{\mathbf{u},\mathbf{y}} \quad & \tfrac{1}{2}\left\|\psi\left(\mathbf{u}\right) - \bar{\mathbf{x}}\right\|_{\mathcal{Q}} + \tfrac{1}{2}\left\|\mathbf{u} - \bar{\mathbf{u}}\right\|_{\mathcal{R}} =: J(\mathbf{u}) \\ \text{s.t.} \quad & -\mathbf{u} + \boldsymbol{a} + \tfrac{1}{2}\text{diag}(\mathbf{y}_\text{a})\mathbf{y}_\text{a} = 0 \\ & \mathbf{u} - \boldsymbol{b} + \tfrac{1}{2}\text{diag}(\mathbf{y}_\text{b})\mathbf{y}_\text{b} = 0 \\ & \tfrac{1}{2}\psi_{\tilde{k}}(\mathbf{u})^\top P \psi_{\tilde{k}}(\mathbf{u}) - c + \tfrac{1}{2}y_\text{c}^2 = 0. \end{aligned} \tag{21}$$

The primal and dual variable updates in (4) are determined as explained in Section III-A. Note that the matrix inversion in (18) can be computed analytically offline. In fact, since the gradient of the constraint is $\nabla g(\mathbf{u}) = [\,-I \mid I \mid q\,]$, where the vector $q \in \mathbb{R}^{Nn_\text{u}}$ indicates the derivative of the last constraint, the matrix is inverted as follows:

$$\begin{aligned} & \left(\nabla g(\mathbf{u})^\top \nabla g(\mathbf{u}) + \text{diag}([\mathbf{y}_\text{a}; \mathbf{y}_\text{b}; y_\text{c}])^2\right)^{-1} \\ & = \begin{bmatrix} I + \text{diag}(\mathbf{y}_\text{a})^2 & -I & -q \\ -I & I + \text{diag}(\mathbf{y}_\text{b})^2 & q \\ -q^\top & q^\top & q^\top q + y_\text{c}^2 \end{bmatrix}^{-1} \\ & = \begin{bmatrix} D + B + r(Bq)(Bq)^\top & D - r(Bq)(Aq)^\top & rBq \\ D - r(Aq)(Bq)^\top & D + A + r(Aq)(Aq)^\top & -rAq \\ r(Bq)^\top & -r(Aq)^\top & r \end{bmatrix}, \end{aligned} \tag{22}$$

$$d_i := \frac{1}{y_{\text{a},i}^2 + y_{\text{b},i}^2 + y_{\text{a},i}^2 y_{\text{b},i}^2},\ r := \left(\sum_{i=1}^{Nn_\text{u}} y_{\text{a},i}^2 y_{\text{b},i}^2 d_i q_i^2 + y_\text{c}^2\right)^{-1},$$

and $D := \text{diag}(d_i)$, $A := \text{diag}(y_{\text{a},i}^2 d_i)$, $B := \text{diag}(y_{\text{b},i}^2 d_i)$. Because of the diagonal structure of A and B, the vectors Aq and Bq are cheap to compute, and this allows one to compute the matrix multiplication in (18) in only $\mathcal{O}\left(Nn_\text{u}\right)$ floating point operations (FLOPS). Moreover, since the contractive constraint in (20) has the same structure as the terminal cost in the objective function, the computation of q is inexpensive when performed together with the computation of $\nabla J(\mathbf{u})$.

The primal variable updates $\mathbf{d}_\text{u}$ and the slack updates $\mathbf{d}_{\text{y,a}}$, $\mathbf{d}_{\text{y,b}}$ and $d_{\text{y,c}}$ follow from (19). The computation of the gradient of the objective function can also be done efficiently by exploiting the causality of the nonlinear dynamics, f. From the definition of $J(\mathbf{u})$ in (21), we have

$$\nabla J(\mathbf{u}) = \nabla\psi(\mathbf{u})\mathcal{Q}\Delta\mathbf{x} + \mathcal{R}\Delta\mathbf{u}, \tag{23}$$

where the matrix $\nabla\psi(\mathbf{u})$ contains the standard linearization matrices of the nonlinear dynamics $F_k := \frac{\partial f}{\partial x}(\Delta x_k, \Delta u_k)$, $G_k := \frac{\partial f}{\partial u}(\Delta x_k, \Delta u_k)$ and it is block upper triangular, while $\mathcal{Q}$ and $\mathcal{R}$ are block diagonal. Thus:

$$\begin{aligned} \nabla\psi\,\mathcal{Q}\mathbf{x} &= \begin{bmatrix} G_0^\top & \cdots & & G_0^\top F_{N-1}^\top \cdots F_0^\top \\ & \ddots & & \vdots \\ & & G_{N-2}^\top & G_{N-2}^\top F_{N-1}^\top \\ & & & G_{N-1}^\top \end{bmatrix} \begin{bmatrix} Q\Delta x_1 \\ \vdots \\ Q\Delta x_{N-1} \\ P\Delta x_N \end{bmatrix} \\ &= \begin{bmatrix} \vdots \\ G_{N-3}^\top \left(Q\Delta x_{N-2} + F_{N-2}^\top \left(Q\Delta x_{N-1} + F_{N-1}^\top P\Delta x_N\right)\right) \\ G_{N-2}^\top \left(Q\Delta x_{N-1} + F_{N-1}^\top \left(P\Delta x_N\right)\right) \\ G_{N-1}^\top P\Delta x_N \end{bmatrix} \end{aligned}$$

which, for diagonal cost matrices Q, R and full P, can be computed in $\mathcal{O}\left(N(n_\text{x}^2 + n_\text{x}n_\text{u})\right)$ FLOPS (when computing the second last term, one need not recompute the term $P\Delta x_N$ and similarly for the terms above). Since the other steps of Algorithms 1 and 2 have lower complexity, including the computation of the merit function $\phi(t)$ and its derivative $\phi'(t)$ (or directional derivative $D\phi(t)$), this is the resulting complexity of the algorithm. Note that this complexity is competitive both with the Gradient Descent method for linear MPC [17], $\mathcal{O}\left((Nn_\text{u})^2\right)$, and the SQP method, whose complexity depends on the complexity of the QP solver. The Active Set Method would require $\mathcal{O}\left((Nn_\text{u})^2\right)$ FLOPS [18], while an Interior Point Method exploiting sparsity of the MPC requires $\mathcal{O}\left(N(n_\text{x}^3 + n_\text{x}^2 n_\text{u})\right)$ FLOPS [19].

IV. Computational results

We show the performance of the proposed algorithms in two examples: (1) contractive MPC of a centrifugal compressor; (2) MPC with a terminal constraint for an inverted pendulum. Algorithm 1 and 2 are implemented in the open-source tool FalcOpt [20], which allows one to automatically generate library-free and embeddable C code solving the nonlinear MPC problem. The generated code is specifically tailored for the considered application and heavily exploits sparsity in the problem data. The problem specifications can be given either via MATLAB or Simulink and interfaced to the C code via automatically generated MEX functions. The computational times shown here are relative to an off-the-shelf Windows computer with processor Intel Core i7-3740QM 2.70Ghz.

Contractive MPC for an industrial compressor: We consider the control of the compression system in [21, Section III-C], which is a contractive MPC with box constraints. The dimension of the problem is characterized by $N = 20$, $n_\text{x} = 5$ and $n_\text{u} = 2$. The computational times obtained by the code generated Algorithms 1 and 2 (with $p = 1$, $p = 2$ and $p = \infty$) are given in Table I for convergence tolerance $\varepsilon = 10^{-6}$ along with commercial solvers such as the SQP solver SNOPT [22] and the Interior-Point Method (IPM) solver FORCES Pro NL [19]. We also consider as an approximate solution the Real-Time Iteration (RTI), which

Authorized licensed use limited to: TU Delft Library. Downloaded on February 11,2021 at 11:17:06 UTC from IEEE Xplore. Restrictions apply.

TABLE I

COMPUTATIONAL TIMES FOR THE COMPRESSOR, TOL. $\varepsilon = 10^{-6}$

Method (Solver)	Avg. (ms)	Best (ms)	Worst (ms)
Algorithm 1 (FalcOpt)	**0.0892**	**0.0724**	**0.169**
Algorithm 2, $p = 1$ (FalcOpt)	0.101	0.0797	0.307
Algorithm 2, $p = 2$ (FalcOpt)	**0.0835**	**0.0677**	**0.167**
Algorithm 2, $p = \infty$ (FalcOpt)	0.147	0.111	0.418
SQP (SNOPT)	256	109	889
IPM (FORCES Pro NL)	1.06	0.616	2.70
RTI (FORCES Pro)	0.727	0.605	1.27

TABLE II

COMPUTATIONAL TIMES FOR THE INV. PENDULUM, TOL. $\varepsilon = 10^{-6}$

Method (Solver)	Avg. (ms)	Best (ms)	Worst (ms)
Algorithm 1 (FalcOpt)	3.05	0.411	4.74
Algorithm 2, $p = 1$ (FalcOpt)	**0.768**	**0.228**	**1.20**
Algorithm 2, $p = 2$ (FalcOpt)	1.189	0.176	2.10
Algorithm 2, $p = \infty$ (FalcOpt)	1.36	0.233	1.72
SQP (SNOPT)	9.66*	4.00	370*
IPM (FORCES Pro NL)	**0.127**	**0.0779**	**0.451**
RTI (FORCES Pro)	**0.160**	**0.107**	**0.244**

* The first MPC optimization in SNOPT exceeds the maximum number of major iterations and is not considered here.

yields a closed-loop cost comparable with the full-nonlinear solutions [23]. The proposed algorithms outperform both the exact and approximate methods in the considered example (best times are highlighted in bold).

Inverted pendulum with a terminal constraint: In our numerical experience, we observe that the computational advantage of ℓ_p merit functions is application dependent. In Table II, we show the numerical results obtained for the control of the inverted pendulum ($N = 8$, $n_{\mathrm{x}} = 4$ and $n_{\mathrm{u}} = 1$) with a terminal constraint, presented in [3, Section 6]. In this case, the dynamics are highly nonlinear and contain sinusoids, which are expensive to compute. As a result, Algorithms 1 and 2 require at least 10 times more iterations than IPM to converge. Using ℓ_p merit functions (Algorithm 2) yield to a substantial improvement over the performance obtained with Algorithm 1. In fact, Algorithm 1 requires on average 2564 iterations, compared to 582, 978 and 721 for Algorithm 2 respectively with $p = 1$, 2, ∞.

V. CONCLUSION

We have analyzed a projected gradient and constraint linearization method, which consists of taking a gradient step and project it onto a local linearization of the constraints. We have shown global convergence to a critical point of the nonlinear problem by means of a simplified ℓ_p merit function, which has some beneficial theoretical properties, such as keeping the penalty parameter bounded.

In our numerical experience, the most reasonable choice is $p = 1$. When the dynamics are expensive to evaluate, the proposed algorithms are slower than interior-point methods.

REFERENCES

[1] P. T. Boggs and J. W. Tolle, "Sequential quadratic programming," *Acta Numerica*, vol. 4, pp. 1–51, 1995.

[2] J. B. Rosen, "The gradient projection method for nonlinear programming. Part II. Nonlinear constraints," *Journal of the Society for Industrial and Applied Mathematics*, vol. 9, no. 4, pp. 514–532, 1961.

[3] G. Torrisi, S. Grammatico, R. S. Smith, and M. Morari, "A projected gradient and constraint linearization method for nonlinear model predictive control." [Online]. Available: http://arxiv.org/abs/1610.06834

[4] ——, "A variant to sequential quadratic programming for nonlinear model predictive control," *Proc. of the IEEE Conference on Decision and Control*, pp. 2814–2819, 2016.

[5] M. H. Wright, "Numerical methods for nonlinearly constrained optimization," Ph.D. dissertation, Stanford University, 1976.

[6] D. Q. Mayne and N. Maratos, "A first order, exact penalty function algorithm for equality constrained optimization problems," *Mathematical Programming*, vol. 16, no. 1, pp. 303–324, 1979.

[7] L. T. Biegler, *Nonlinear programming: concepts, algorithms, and applications to chemical processes*. SIAM, 2010.

[8] Y. Mao, M. Szmuk, and B. Açıkmeşe, "Successive convexification of non-convex optimal control problems and its convergence properties," pp. 3636–3641, 2016.

[9] G. Torrisi, "Low-complexity numerical methods for nonlinear model predictive control," Ph.D. dissertation, ETH Zurich, Switzerland, 2017.

[10] P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright, "Some theoretical properties of an augmented Lagrangian merit function," in *Advances in Optimization and Parallel Computing*. North Holland, 1992, pp. 101–128.

[11] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.

[12] S. P. Han, "A globally convergent method for nonlinear programming," *Journal of optimization theory and applications*, vol. 22, no. 3, pp. 297–309, 1977.

[13] N. Maratos, "Exact penalty function algorithms for finite dimensional and control optimization problems," Ph.D. dissertation, Imperial College London (University of London), 1978.

[14] J. Nocedal and S. Wright, *Numerical Optimization*. Springer, 1999.

[15] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789 – 814, 2000.

[16] S. de Oliveira Kothare and M. Morari, "Contractive model predictive control for constrained nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 45, no. 6, pp. 1053–1071, 2000.

[17] S. Richter, C. N. Jones, and M. Morari, "Real-time input-constrained MPC using fast gradient methods," *Proc. of the IEEE Conference on Decision and Control and Chinese Control Conference*, pp. 7387–7393, 2009.

[18] H. J. Ferreau, H. G. Bock, and M. Diehl, "An online active set strategy to overcome the limitations of explicit MPC," *International Journal of Robust and Nonlinear Control*, vol. 18, no. 8, pp. 816–830, 2008.

[19] A. Domahidi and J. Jerez, "FORCES Professional," embotech GmbH (http://embotech.com/FORCES-Pro), July 2014.

[20] G. Torrisi, D. Frick, T. Robbiani, S. Grammatico, R. S. Smith, and M. Morari, "FalcOpt: First-order Algorithm via Linearization of Constraints for OPTimization," https://github.com/torrisig/FalcOpt, May 2017.

[21] G. Torrisi, S. Grammatico, A. Cortinovis, M. Mercangöz, M. Morari, and R. S. Smith, "Model predictive approaches for active surge control in centrifugal compressor," *IEEE Transactions on Control System Technology* (in press), 2017. [Online]. Available: https://doi.org/10.1109/TCST.2016.2636027

[22] P. E. Gill, W. Murray, and M. A. Saunders, "SNOPT: An SQP algorithm for large-scale constrained optimization," *SIAM review*, vol. 47, no. 1, pp. 99–131, 2005.

[23] M. Diehl, H. Bock, J. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer, "Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations," *Journal of Process Control*, vol. 12, no. 4, pp. 577–585, 2002.

Authorized licensed use limited to: TU Delft Library. Downloaded on February 11,2021 at 11:17:06 UTC from IEEE Xplore. Restrictions apply.