Adjoint-based SQP Method with Block-wise quasi-Newton Jacobian Updates for Nonlinear Optimal Control

Pedro Hespanhol[a,b] and Rien Quirynen[a]

[a]Control and Dynamical Systems, Mitsubishi Electric Research Laboratories, Cambridge, MA, 02139, USA. quirynen@merl.com
[b]Department of Industrial Engineering and Operations Research, University of California, Berkeley, CA 94720, USA.

ABSTRACT
Nonlinear model predictive control (NMPC) generally requires the solution of a non-convex optimization problem at each sampling instant under strict timing constraints, based on a set of differential equations that can often be stiff and/or that may include implicit algebraic equations. This paper provides a local convergence analysis for the recently proposed adjoint-based sequential quadratic programming (SQP) algorithm that is based on a block-structured variant of the two-sided rank-one (TR1) quasi-Newton update formula to efficiently compute Jacobian matrix approximations in a sparsity preserving fashion. A particularly efficient algorithm implementation is proposed in case an implicit integration scheme is used for discretization of the optimal control problem, in which matrix factorization and matrix-matrix operations can be avoided entirely. The convergence analysis results as well as the computational performance of the proposed optimization algorithm are illustrated for two simulation case studies of nonlinear MPC.

1. Introduction

Optimization based control and estimation techniques have attracted an increasing attention over the past decades. They allow a model-based design framework, in which the system dynamics, performance metrics and constraints can directly be taken into account. Receding horizon techniques such as model predictive control (MPC) and moving horizon estimation (MHE) have been studied extensively because of their desirable properties [30] and these optimization-based techniques have already been applied in a wide range of applications [21]. One of the main practical challenges in implementing such an optimization-based predictive control or estimation scheme, lies in the ability to solve the corresponding nonlinear and generally non-convex optimal control problem (OCP) under strict timing constraints and typically on embedded hardware with limited computational capabilities and available memory.

Let us consider the following continuous-time formulation of the optimal control problem that needs to be solved at each sampling instant

$$\min_{x(\cdot),u(\cdot)} \quad \int_0^T \ell(x(t), u(t)) \, \mathrm{d}t \tag{1a}$$

$$\text{s.t.} \quad x_0 - \hat{x}_0 = 0, \tag{1b}$$

$$0 = f(\dot{x}(t), x(t), u(t)), \quad \forall t \in [0, T], \tag{1c}$$

$$p(x(t), u(t)) \le 0, \quad \forall t \in [0, T], \tag{1d}$$

where T denotes the control horizon length, $x(t) \in \mathbb{R}^{n_x}$ denotes the differential states and $u(t) \in \mathbb{R}^{n_u}$ are the control inputs. The function $\ell(\cdot)$ defines the stage cost and the nonlinear system dynamics are formulated as an implicit system of ordinary differential equations (ODE) in (1c), which could additionally be extended with implicit algebraic equations. A common assumption is that the resulting system of differential-algebraic equations (DAE) is of index 1 [7]. The optimization problem is parametric, since it depends on the state estimate $\hat{x}_0$ at the current sampling instant, through the initial value condition in (1b). The path constraints are defined by the function $p(\cdot)$ in Eq. (1d) and, for simplicity of notation, they are further assumed to be affine. Note that a similar problem as in (1) needs to be solved for optimization-based state and parameter estimation, without the given initial state value.

In direct optimal control methods, one forms a discrete-time approximation of the continuous-time OCP in (1) based on an appropriate parameterization of the state and control trajectories over the time horizon $t \in [0, T]$, resulting in a tractable nonlinear program (NLP) that needs to be solved. Popular examples of this approach include the direct multiple shooting method [8] and direct collocation [3, 5]. Note that these techniques often need to rely on implicit integration methods in order to deal with stiff and/or implicit systems of differential or differential-algebraic equations [36]. The resulting constrained optimization problem can be handled by standard Newton-type algorithms such as interior point methods [38] and sequential quadratic programming (SQP) [10] techniques for nonlinear optimization [31].

Quasi-Newton optimization methods are generally popular for solving such a constrained NLP. They result in computationally efficient Newton-type methods that solve the first order necessary conditions of optimality, i.e., the Karush-Kuhn-Tucker (KKT) conditions, without evaluating the complete Hessian of the Lagrangian and/or even without evaluating the Jacobian of the constraints [31]. Instead, quasi-Newton methods are based on low-rank update formulas for the Hessian and Jacobian matrix approximations [14]. Popular examples of this approach include the Broyden-Fletcher-Goldfarb-Shanno (BFGS) [11] and the symmetric rank-one (SR1) update formula [13] for approximating the Hessian of the Lagrangian. Similarly, quasi-Newton methods can be used for approximating Jacobian matrices, e.g., of the constraint functions, such as the good and bad Broyden methods [12] as well as the more recently proposed two-sided rank-one (TR1) update formula [25].

For the purpose of real-time predictive control and estimation, continuation-based online algorithms have been proposed that aim at further reducing the computational effort by exploiting the fact that a sequence of closely related parametric optimization problems is solved [7, 17]. One popular technique consists of the real-time iteration (RTI) algorithm that performs a single SQP iteration per time step, in combination with a sufficiently high sampling rate and a prediction-based warm starting in order to allow for closed-loop stability of the system [16]. The RTI algorithm can be

implemented efficiently based on (fixed-step) integration schemes with tailored sensitivity propagation for discretization and linearization of the system dynamics [36] in combination with structure-exploiting quadratic programming solvers [21]. In addition, a lifted algorithm implementation has been proposed in [37] to directly embed the iterative procedure of implicit integration schemes, e.g., collocation methods, within a Newton-type optimization framework for optimal control.

Unlike standard inequality constrained optimization, nonlinear optimal control problems typically result in a particular sparsity structure in the Hessian of the Lagrangian and in the Jacobian matrix for the equality constraints. In direct optimal control methods, the objective function is typically separable resulting in a block-diagonal Hessian matrix. This property has been exploited in partitioned quasi-Newton methods that approximate and update each of the Hessian block matrices separately, as proposed and studied in [23, 24, 29]. On the other hand, the Jacobian matrix corresponding to the discretized system dynamics has a block bidiagonal sparsity structure, because of the stage wise coupling of the optimization variables at subsequent time steps of the control horizon. For this purpose, the present article analyzes a novel tailored quasi-Newton method for optimal control using a partitioned or block-structured TR1-based Jacobian update formula. This adjoint-based SQP method for nonlinear optimal control, based on a Gauss-Newton Hessian approximation in combination with inexact Jacobian matrices, was proposed recently in [28].

1.1. Contributions and outline

This paper provides a complete presentation of the block-TR1 based SQP method for nonlinear optimal control, including a detailed discussion of the lifted collocation type implementation, extending earlier work of the same authors in [28]. Unlike the latter publication, a convergence analysis of this novel quasi-Newton type optimization algorithm is provided. More specifically, we prove convergence of the block-structured quasi-Newton Jacobian approximations to the exact Jacobian matrix within the null space of the active inequality constraints. Based on this result, under mild conditions, convergence of the overall inexact SQP method can be guaranteed. Locally linear or superlinear convergence rates can be shown, respectively, when using a Gauss-Newton or quasi-Newton based Hessian approximation scheme. In addition, it is shown how this convergence analysis extends to our lifted collocation implementation that avoids any matrix factorization or matrix-matrix operations. These convergence analysis results as well as the computational performance of the optimization algorithms are illustrated numerically for two simulation case studies of nonlinear MPC.

The paper is organized as follows. Section 2 briefly introduces the direct multiple shooting based OCP problem formulation as well as the proposed adjoint-based inexact SQP method that is based on block-wise TR1 Jacobian updates. Section 3 presents the detailed convergence analysis for the optimization method and contains the main theoretical results of the present paper. A numerically efficient implementation of the block-TR1 update formula in combination with a lifted Newton-type method for direct optimal control with implicit integration schemes such as, e.g., collocation methods, is then proposed and analyzed in Section 4. Finally, Section 5 presents numerical results of the NMPC case studies and Section 6 concludes the paper.

2. Block-wise TR1 based Sequential Quadratic Programming

A popular approach for direct optimal control is based on direct multiple shooting [8] that performs a time discretization, based on a numerical integration scheme [26] to solve the following initial value problem

$$0 = f(\dot{x}(\tau), x(\tau), u(\tau)), \quad \tau \in [t_i, t_{i+1}], \quad x(t_i) = x_i, \tag{2}$$

on each of N shooting intervals that are defined by a grid of consecutive time points t_i for $i = 0, \ldots, N$. For the sake of simplicity, we consider here an equidistant grid over the control horizon, i.e., $t_{i+1} - t_i = \frac{T}{N}$, and a piecewise constant control parametrization $u(\tau) = u_i$ for $\tau \in [t_i, t_{i+1})$ in (2). An explicit fixed-step integration scheme defines the discrete-time system dynamics $x_{i+1} = F_i(x_i, u_i)$ for the shooting interval $[t_i, t_{i+1}]$. For example, this can correspond to the popular Runge-Kutta method of order 4 (RK4) as defined in [26]. Based on the explicit discretization scheme, the resulting block-structured optimal control problem reads as

$$\min_{X, U} \quad \sum_{i=0}^{N-1} l_i(x_i, u_i) + l_N(x_N) \tag{3a}$$

$$\text{s.t.} \quad \hat{x}_0 = x_0, \tag{3b}$$

$$F_i(x_i, u_i) = x_{i+1}, \quad i = 0, \ldots, N-1, \tag{3c}$$

$$P_i\, w_i \leq p_i, \quad i = 0, \ldots, N, \tag{3d}$$

where the affine path constraints (3d) have been imposed on each of the shooting nodes and the compact notation $w_i := (x_i, u_i)$ for $i = 0, \ldots, N-1$ and $w_N := x_N$ is defined. Note that the optimization variables for the problem in (3) are directly the state $X = [x_0^\top, \ldots, x_N^\top]^\top$ and control trajectory $U = [u_0^\top, \ldots, u_{N-1}^\top]^\top$.

2.1. SQP algorithm with inexact Jacobians

For a local minimum w^* of the NLP in (3), for which the linear independence constraint qualification (LICQ) holds, there must exist a unique set of multiplier values λ^*, μ^* such that the following Karush-Kuhn-Tucker (KKT) conditions are satisfied

$$\nabla_w \mathcal{L}(w^*, \lambda^*) + P^T \mu^* = 0 \tag{4a}$$

$$F(w^*) = X^* \tag{4b}$$

$$P\, w^* \leq p \tag{4c}$$

$$\mu^* \geq 0 \tag{4d}$$

$$\mu_j^*(Pw^* - p)_j = 0, \quad j = 1, \ldots, n_\mathrm{p}, \tag{4e}$$

where $F(\cdot)$ and P are appropriate block-wise concatenations of the equality and inequality constraints, respectively, in (3c) and (3d) and n_p denotes the total number of inequality constraints. Here, we also lumped the initial condition constraint as part of the matrix P since we can represent a linear equality constraint as two linear inequality constraints. Lastly, $\mathcal{L}(w, \lambda)$ denotes the 'truncated Lagrangian', omitting inequality

constraints, and is therefore given by

$$\mathcal{L}(w, \lambda) = \sum_{i=0}^{N-1} \left(l_i(x_i, u_i) + \lambda_i^\top \left(F_i(w_i) - x_{i+1} \right) \right) + l_N(x_N). \tag{5}$$

Given the set of indices $\mathcal{A}$ for the inequality constraints that are active at the local minimum, the KKT system reduces to a nonlinear system of equations that can be solved directly by a Newton-type method. In particular, we are interested in a quasi-Newton algorithm where we will approximate $\nabla^2_{ww}\mathcal{L}(w^k, \lambda^k)$ by a matrix H^k and $\frac{\partial F}{\partial w}(w^k)$ by a matrix A^k. Namely, we solve the following linearized system

$$\begin{bmatrix} H^k & A^{k\top} - E^\top & P_A^\top \\ A^k - E & & \\ P_A & & \end{bmatrix} \begin{bmatrix} \Delta w^k \\ \Delta \lambda^k \\ \Delta \mu_A^k \end{bmatrix} = - \begin{bmatrix} g(w^k, \lambda^k) \\ F(w^k) - X^k \\ P_A\, w^k - p_A \end{bmatrix}, \tag{6}$$

where $g(w^k, \lambda^k) = \nabla_w \mathcal{L}(w^k, \lambda^k) + \mu_A^{k\top} (P_A\, w^k - p_A)$ at each Newton-type iteration k. Note that the matrix P_A is defined as the part of P that corresponds to the inequality constraints (4c) in the active set $\mathcal{A}$, and E denotes the constant matrix corresponding to the right-hand side of the equality constraints in (4b).

In order to efficiently solve the inequality constrained OCP in (3), let us consider the adjoint-based SQP algorithm with Gauss-Newton type Hessian approximation and inexact Jacobian information as introduced originally in [7, 39] for fast nonlinear MPC. Each SQP iteration solves a convex QP subproblem

$$\min_{\Delta W} \quad \sum_{i=0}^{N} \frac{1}{2} \Delta w_i^\top H_i^k \, \Delta w_i \, + \, h_i^{k\top} \Delta w_i \tag{7a}$$

$$\text{s.t.} \quad \Delta x_0 \, = \, \hat{x}_0 - x_0^k, \tag{7b}$$

$$a_i^k + A_i^k \, \Delta w_i \, = \, \Delta x_{i+1}, \qquad i = 0, \dots, N-1, \tag{7c}$$

$$P_i \, \Delta w_i \le p_i^k, \qquad\qquad i = 0, \dots, N, \tag{7d}$$

where notation $\Delta W = [\Delta w_0^\top, \dots, \Delta w_N^\top]^\top$ is used to denote the deviation variables $\Delta w_i := w_i - w_i^k$, given the current solution guess X^k, U^k for the state and control trajectories at iteration k of the adjoint-based SQP method. The function $p(\cdot)$ that defines the path constraint (1d) was assumed to be affine and $p_i^k := p_i - P_i\, w_i^k$. Note that tracking formulations for nonlinear MPC typically include a stage cost that is defined by a (nonlinear) least squares term $l_i(x_i, u_i) = \frac{1}{2}\|R(x_i, u_i)\|_2^2$ for $i = 0, \dots, N$. The generalized Gauss-Newton (GGN) method from [6] uses the block-structured Hessian approximation $H_i^k := \nabla R(w_i^k) \nabla R(w_i^k)^\top \approx \nabla^2_{w_i w_i} \mathcal{L}(\cdot)$.

The matrix $A_i^k \approx \frac{\partial F_i}{\partial w_i}(w_i^k)$ denotes the Jacobian approximation and $a_i^k := F_i(w_i^k) - x_{i+1}^k$ for the discrete-time system dynamics in Eq. (7c). For real-time NMPC, such a Jacobian approximation can be obtained by reusing information from a previous NLP solution [7, 39]. The gradient term in the objective (7a) reads as

$$h_i^k := \nabla_{w_i} l(w_i^k) + \left(\frac{\partial F_i}{\partial w_i}(w_i^k) - A_i^k \right)^\top \lambda_i^k, \tag{8}$$

for $i = 0, \dots, N-1$, in which λ_i^k denotes the current value of the Lagrange multi-

pliers for the nonlinear continuity constraints in (3c). Note that the linearized KKT conditions in (6) correspond to the KKT optimality conditions for the QP in (7), for a fixed active set $\mathcal{A}$. In addition, each QP subproblem is convex because $H^k \succeq 0$, e.g., for the Gauss-Newton Hessian approximation. A full-step inexact SQP method will sequentially solve each QP subproblem (7) and perform the following updates:

$$w^{k+1} = w^k + \Delta w^k \ \text{ and } \ \lambda^{k+1} = \lambda^k + \Delta \lambda^k = \lambda_{QP}^{k+1}, \tag{9}$$

where λ_{QP}^{k+1} denote the Lagrange multiplier values for Eq. (7c) at the QP solution. We do not need to perform explicit updates for the Lagrange multipliers associated with the inequality constraints, because they are assumed to be affine, hence not impacting any computation on the QP formulation in (7).

2.2. Dynamic block-wise TR1 Jacobian updates

At each SQP iteration, we perform the block-wise two-sided rank-one (TR1) Jacobian update, as proposed recently in [28]. Following the work in [25], given current Jacobian approximations A_i^k for $i = 0, \ldots, N-1$, we would like that each updated approximation matrix A_i^{k+1} satisfies the following two secant conditions

$$\begin{aligned} \text{Adjoint Condition (AC):} \quad & \sigma_i^{k\top} A_i^{k+1} = \gamma_i^{k\top} \\ \text{Forward Condition (FC):} \quad & A_i^{k+1} s_i^k = y_i^k, \end{aligned} \tag{10}$$

where we define the adjoint vector $\gamma_i^k = \frac{\partial F_i}{\partial w_i}(w_i^{k+1})^\top \sigma_i^k$, given $\sigma_i^{k\top} = (\lambda_i^{k+1} - \lambda_i^k)^\top$, and the difference in function evaluations $y_i^k = F(w_i^{k+1}) - F(w_i^k)$. Note that λ_i^{k+1} and λ_i^k, respectively, denote the new and old Lagrange multipliers for the linearized equality constraints in Eq. (7c). Similarly, $w_i^k := (x_i^k, u_i^k)$ and $w_i^{k+1} := w_i^k + \Delta w_i^k$ denote, respectively, the old and new primal variables, such that $s_i^k := w_i^{k+1} - w_i^k$. Note that the gradient $\gamma_i^k = \frac{\partial F_i}{\partial w_i}(w_i^{k+1})^\top \sigma_i^k$ can be computed efficiently using the backward or adjoint mode of algorithmic differentiation (AD), e.g., see [22].

The proposed block-wise TR1 update formula then reads as follows

$$A_i^{k+1} = A_i^k + \alpha_i^k \left(y_i^k - A_i^k s_i^k \right) \left(\gamma_i^{k\top} - \sigma_i^{k\top} A_i^k \right), \tag{11}$$

for $i = 0, \ldots, N-1$ and where α_i^k is a scalar that will be defined further. Aside from the case where the function $F(\cdot)$ is affine, the two conditions in Eq. (10) are not consistent with each other and they can therefore generally not both be satisfied by the updated matrix A_i^{k+1} at each iteration. Thus, similar to the standard TR1 update in [25], the block-wise update will only be able to satisfy one or the other. In the adjoint variant of the update, the scaling value is defined as

$$\alpha_{A,i}^k = \frac{1}{\sigma_i^{k\top}(y_i^k - A_i^k s_i^k)}, \tag{12}$$

such that the adjoint condition in (10) is satisfied exactly and the forward condition

holds up to some accuracy. Similarly, this value reads as follows for the forward variant

$$\alpha_{\mathrm{F},i}^k = \frac{1}{(\gamma_i^{k\top} - \sigma_i^{k\top} A_i^k)\, s_i^k},\tag{13}$$

where the forward condition is satisfied exactly. It is interesting to note that, since we apply the block-wise TR1 update from (11) for each shooting interval $i = 0, \ldots, N-1$, the resulting update for the complete constraint Jacobian matrix of the QP in (7) corresponds to a rank-N update.

As in [25], we impose a skipping condition in order to avoid a potential blow-up of the block-wise TR1 update when the denominator of the scaling factor becomes small or even zero. For our purposes, the skipping condition itself depends on the type of formula that is used. We update the block matrix A_i^k only if the following holds

$$\left| (\gamma_i^{k\top} - \sigma_i^{k\top} A_i^k) s_i^k \right| \geq c_1 \left\| \sigma_i^k \right\| \left\| y_i^k - A_i^k s_i^k \right\|,\tag{14}$$

with $c_1 \in (0,1)$ if $\alpha_i^k = \alpha_{\mathrm{F},i}^k$ in the forward TR1 update, and

$$\left| \sigma_i^{k\top} (y_i^k - A_i^k s_i^k) \right| \geq c_1 \left\| s_i^k \right\| \left\| \gamma_i^k - A_i^{k\top} \sigma_i^k \right\|,\tag{15}$$

with $c_1 \in (0,1)$ if $\alpha_i^k = \alpha_{\mathrm{A},i}^k$ in the adjoint TR1 update. In addition, in order to consistently choose either the forward or adjoint Jacobian update formula, we propose a more dynamic variant of the algorithm that picks either $\alpha_{\mathrm{F},i}^k$ or $\alpha_{\mathrm{A},i}^k$ for each block matrix at any given iteration. It may not be clear what is the best approach to select which type of update is to be executed for each block matrix at a given iteration. However, in the next section, we prove the local convergence properties of the algorithm under any arbitrary sequence of updates that satisfy the skipping conditions in (14) and (15) for each block i at every iteration k.

Algorithm 1 One iteration of SQP method with block-wise TR1 Jacobian updates.

Input: $w_i^k = (x_i^k, u_i^k)$, λ_i^k and A_i^k for $i = 0, \ldots, N-1$.

    `Problem linearization and QP preparation`

1: Formulate the QP in (7) with Jacobian matrices A_i^k, Gauss-Newton Hessian approximations H_i^k and vectors a_i^k, p_i^k and h_i^k in (8) for $i = 0, \ldots, N-1$.

    `Computation of Newton-type step direction`

2: Solve the QP subproblem in Eq. (7) to update optimization variables:
    $w_i^{k+1} \leftarrow w_i^k + \Delta w_i^k$ and $\lambda_i^{k+1} \leftarrow \lambda_i^k + \Delta \lambda_i^k$.   ▷ full step

    `Block-wise TR1 Jacobian updates`

3: **for** $i = 0, \ldots, N-1$ **do in parallel**

4:     Choose $\alpha_i^k = \alpha_{\mathrm{F},i}^k$ or $\alpha_i^k = \alpha_{\mathrm{A},i}^k$ via some decision rule.

5:     $A_i^{k+1} \leftarrow A_i^k + \alpha_i^k \left(y_i^k - A_i^k s_i^k \right) \left(\gamma_i^{k\top} - \sigma_i^{k\top} A_i^k \right)$.

6: **end for**

Output: $w_i^{k+1} = (x_i^{k+1}, u_i^{k+1})$, λ_i^{k+1} and A_i^{k+1} for $i = 0, \ldots, N-1$.

The complete adjoint-based SQP method that uses parallelizable block-wise TR1 Jacobian updates is summarized in Algorithm 1. Note that, for simplicity, the SQP algorithm is presented as a full-step method without any globalization or step-length

selection strategies to ensure convergence to a local minimum [31]. This is also further motivated by the use of online algorithms for real-time nonlinear MPC as discussed in [17].

3. Convergence Results for Block-wise TR1-based SQP Method

For the convergence analysis of sequential quadratic programming, it is standard to rely on a result that the active set, i.e., the set of active inequality constraints in the QP subproblems is stable in a neighbourhood around a local minimizer of the nonlinear program [31]. This allows us to study the local convergence properties of the block-TR1 based SQP method under the assumption that the active set has already been fixed, resulting, locally, in an equality constrained problem.

3.1. Stability of the active set and local convergence

Let us start by briefly repeating the result from [20] on the stability of the active set in the QP subproblems near the NLP solution and the corresponding conditions on local convergence properties for an adjoint-based SQP method with inexact Jacobians.

Theorem 3.1. *(Stability of active set and local convergence) Let the NLP solution vectors w^*, λ^* be given and assume that:*

(i) *at w^* LICQ holds, and there exist Lagrange multiplier values μ^* such that (w^*, λ^*, μ^*) satisfies the KKT conditions in (4).*

(ii) *at w^* strict complementarity holds, i.e., the multipliers μ_A^* of the active inequalities $P_A w^* = p_A$ satisfy $\mu_A^* > 0$, where P_A is a matrix consisting of all rows of P that correspond to the active inequalities at the NLP solution.*

(iii) *there are two sequences of uniformly bounded matrices (A^k, H^k), each H^k positive semidefinite on the null space of A^k, such that the sequence of matrices*

$$
J^k := \begin{bmatrix} N^\top H^k & N^\top A^{k^\top} \\ A^k \\ P_A \end{bmatrix} \approx \frac{\partial \mathcal{F}}{\partial y}(y^k), \quad \text{where} \quad \mathcal{F}(y) := \begin{bmatrix} N^\top \nabla_w \mathcal{L}(w, \lambda) \\ F(w) - X \\ P_A w - p_A \end{bmatrix},
$$

is uniformly bounded and invertible with a uniformly bounded inverse. Here, N is a null space matrix with appropriate dimensions with orthonormal column vectors such that $N^\top N = \mathbb{1}$ and $P_A N = 0$.

(iv) *there is a sequence of iterates $y^k := (w^k, \lambda^k)$ generated according to*

$$
w^{k+1} = w^k + \Delta w^k \quad \text{and} \quad \lambda^{k+1} = \lambda^k + \Delta \lambda^k = \lambda_{QP}^{k+1},
$$

where Δw^k is the primal solution of the QP subproblem in (7) and λ_{QP}^{k+1} denote the Lagrange multipliers corresponding to the equality constraints (7c). Each iteration can be written in compact form as $y^{k+1} = y^k - J^{k^{-1}} \mathcal{F}(y^k)$.

(v) *there exists $\kappa < 1$ such that, for all $k \in \mathbb{N}$, it can be guaranteed that*

$$
\left\| J^{k+1^{-1}} \left(J^k - \frac{\partial \mathcal{F}}{\partial y}(y^k + t \Delta y^k) \right) \Delta y^k \right\| \leq \kappa \|\Delta y^k\|, \quad \forall t \in [0, 1]. \tag{16}
$$

Then, there exists a neighbourhood $\bar{\mathcal{N}}$ of (w^*, λ^*) such that for all initial guesses $(w^0, \lambda^0) \in \bar{\mathcal{N}}$ the sequence (w^k, λ^k) converges q-linearly towards (w^*, λ^*) with rate κ, and the solution of each QP (7) has the same active set as w^*.

In addition to the latter result that guarantees a q-linear local convergence rate in a neighbourhood of the NLP solution, the following theorem states a condition under which q-superlinear local convergence can be obtained instead.

Theorem 3.2. *(Superlinear convergence) If the equality*

$$\lim_{k \to \infty} \begin{bmatrix} N^\top H^k N & N^\top A^{k\top} \\ A^k N & \mathbb{0} \end{bmatrix} = \begin{bmatrix} N^\top \nabla_{ww}^2 \mathcal{L}(w^\star, \lambda^\star) N & N^\top \frac{\partial F}{\partial w}(w^\star)^\top \\ \frac{\partial F}{\partial w}(w^\star) N & \mathbb{0} \end{bmatrix}, \qquad (17)$$

holds in addition to the assumptions of Theorem 3.1, then the local convergence rate is q-superlinear instead.

The proofs for both Theorem 3.1 and 3.2 can be found in [20] for an adjoint-based SQP method with inexact Jacobians that matches our problem formulation.

3.2. Convergence of the block-wise TR1 Jacobian updates

Theorem 3.1 holds for a general class of constraint Jacobian and Hessian approximation matrices (A^k, H^k). Therefore, we have to show that our block-wise TR1 updates produce a sequence of block-structured matrices that converge to the exact Jacobian, which is itself block-structured, projected onto the null space of the active inequality constraint matrix P_A. Namely, defining a null space matrix N as in Theorem 3.1, we need to prove that the following holds

$$\lim_{k \to \infty} \left\| \left(A_i^k - \frac{\partial F_i}{\partial w}(w_i^*) \right) N_i \right\| = 0, \quad \forall i = 0, \ldots, N-1, \qquad (18)$$

where N_i is the projection of the null space matrix N in the variable space corresponding to block i. The only non-zero entries that are inexact in the Jacobian approximation matrix A^k are those corresponding to the block-TR1 matrices A_i^k, $i = 0, \ldots, N-1$.

Assumption 3.3. *Let us make the following assumptions:*

(AS1) *The Lagrangian function is twice continuously differentiable.*

(AS2) *The function $\nabla_w F(w)$ is Lipschitz continuous, i.e., there exists a constant c_3 such that $\|\nabla_w F(w_1) - \nabla_w F(w_2)\| \leq c_3 \|w_1 - w_2\|$, for any w_1, w_2.*

(AS3) *Let $\{(w^k, \lambda^k)\}$ be a sequence of iterates generated by our block-TR1 based SQP method in Algorithm 1, with a corresponding sequence of update parameters $\{\alpha_i^k\}$, while satisfying the skipping criterion in eqs. (14)-(15).*

(AS4) *The SQP iterates $\{(w^k, \lambda^k)\}$ converge to a limit point (w^*, λ^*).*

(AS5) *There is k_0 such that the active set is stable for all iterates $k \geq k_0$.*

(AS6) *For each block i, the sequence of projections of $\{s^k\}$ on the subspace associated with block i, namely $\{s_i^k\}$ is uniformly linearly independent in the projected null space N_i, i.e., there exist $c_4 > 0$ and $l \geq q_i$ such that, for each $k_i \geq k_0$, there exist q_i distinct indices k_i^j with $k_i \leq k_i^1 < \ldots < k_i^{q_i} \leq k_i + l$, $s_{N,i}^{k_i^j} \in \mathbb{R}^{q_i}$, $s_i^{k_i^j} = N_i s_{N,i}^{k_i^j}$,*

$j = 1, \ldots, q_i$ and the minimum singular value $\sigma_{min}(S_{N_i}^{k_i})$ of the matrix

$$S_{N_i}^{k_i} = \begin{bmatrix} \dfrac{s_{N,i}^{k_i^1}}{\|s_{N,i}^{k_i^1}\|} & \cdots & \dfrac{s_{N,i}^{k_i^{q_i}}}{\|s_{N,i}^{k_i^{q_i}}\|} \end{bmatrix} \tag{19}$$

is bounded below by c_4, i.e., $\sigma_{min}(S_{N_i}^{k_i}) \geq c_4$.

Note that the assumptions $(AS1)$-$(AS5)$ are relatively mild and quite standard in Newton-type convergence analysis of SQP methods [20]. Especially, condition $(AS5)$ holds due to the local stability result in Theorem 3.1 for the active set near the NLP solution. Even though $(AS6)$ seems relatively strong, a very similar assumption is made in existing convergence results for quasi-Newton type matrix update schemes [13, 20]. Here, we only require uniform linear independence inside each block i. We proceed now to prove the convergence of the quasi-Newton block-structured constraint Jacobian approximation matrices, using ideas from [13] and [20]. We start by first showing an intermediate result in the following lemma.

Lemma 3.4. *Given $(AS1)$-$(AS3)$ in Assumption 3.3, then the following holds for each Jacobian block matrix approximation*

$$\left\| y_i^k - A_i^l s_i^k \right\| \leq \frac{c_3}{c_1} \left(\frac{2}{c_1^2} + 1 \right)^{l-k} \eta_i^{l,k} \|s_i^k\|, \quad \forall l \geq k+1, \tag{20a}$$

$$\left\| \gamma_i^k - A_i^{l^\top} \sigma_i^k \right\| \leq \frac{c_3}{c_1} \left(\frac{2}{c_1^2} + 1 \right)^{l-k} \eta_i^{l,k} \|\sigma_i^k\|, \quad \forall l \geq k+1, \tag{20b}$$

where $i = 0, \ldots, N-1$ and $\eta_i^{l,k} = \max\{\|w_i^p - w_i^s\| \mid k \leq s \leq p \leq l\}$ is defined.

Proof. Our proof follows closely the proof of Lemma 4.1 in [20] but extended to our block-structured method and generalized to include both the forward and adjoint TR1 Jacobian update formulas.

Step 1: Eq. (20a) based on forward Jacobian update
We start by showing the result of Eq. (20a) when $\alpha_i = \alpha_{F,i}$. The proof is by induction for each block $i = 0, \ldots, N-1$. For $l = k+1$, we know that $y_i^k - A_i^l s_i^k = 0$ based on the forward update. Assume that the result in (20a) holds for all $j = k+1, \ldots, l$. Then, we have the following

$$\|y_i^k - A_i^{l+1} s_i^k\| = \left\| y_i^k - A_i^l s_i^k - \alpha_{F,i}^l \rho_i^l \tau_i^{l^\top} s_i^k \right\|$$
$$\leq \|y_i^k - A_i^l s_i^k\| + \left| \frac{(\tau_i^l, s_i^k)}{(\tau_i^l, s_i^l)} \right| \|\rho_i^l\|, \tag{21}$$

where $\tau_i^l = \gamma_i^l - A_i^{l^\top} \sigma_i^l$ and $\rho_i^l = y_i^l - A_i^l s_i^l$ such that $\alpha_{F,i}^l = \frac{1}{(\tau_i^l, s_i^l)}$. Then, using the result in Eq. (20a), we can write

$$\left| (\tau_i^l, s_i^k) \right| = \left| (\gamma_i^l - A_i^{l^\top} \sigma_i^l, s_i^k) \right| \leq \left| (\gamma_i^l, s_i^k) - (\sigma_i^l, y_i^k) \right| + \left| (\sigma_i^l, y_i^k) - (\sigma_i^l, A_i^l s_i^k) \right|$$
$$\leq \left| (\gamma_i^l, s_i^k) - (\sigma_i^l, y_i^k) \right| + \frac{c_3}{c_1} \left(\frac{2}{c_1^2} + 1 \right)^{l-k} \eta_i^{l,k} \left\| \sigma_i^l \right\| \left\| s_i^k \right\|. \tag{22}$$

10

Using the mean-value theorem, it follows that

$$
\left|(\gamma_i^l, s_i^k) - (\sigma_i^l, y_i^k)\right| = \left|\sigma_i^{l\top}\left(\frac{\partial F_i}{\partial w}(w_i^l + s_i^l) - \int_0^1 \frac{\partial F_i}{\partial w}(w_i^k + t\,s_i^k)\mathrm{d}t\right) s_i^k\right|
$$
$$
\leq c_3\,\eta_i^{l+1,k}\left\|\sigma_i^l\right\|\left\|s_i^k\right\|,
\tag{23}
$$

based on the Lipschitz continuity in $(AS2)$. From the skipping condition in (14), we know that $\left|(\tau_i^l, s_i^l)\right| \geq c_1\left\|\sigma_i^l\right\|\left\|\rho_i^l\right\|$. In addition, given that $\eta_i^{l,k} \leq \eta_i^{l+1,k}$, we obtain

$$
\left\|y_i^k - A_i^{l+1}s_i^k\right\| \leq \frac{c_3}{c_1}\left(\frac{2}{c_1^2}+1\right)^{l-k}\eta_i^{l,k}\left\|s_i^k\right\|
$$
$$
+ \left(c_3\eta_i^{l+1,k} + \frac{c_3}{c_1}\left(\frac{2}{c_1^2}+1\right)^{l-k}\eta_i^{l,k}\right)\frac{\left\|\sigma_i^l\right\|\left\|s_i^k\right\|}{\left|(\tau_i^l, s_i^l)\right|}\left\|\rho_i^l\right\|
\tag{24}
$$
$$
\leq \frac{c_3}{c_1}\left(\frac{2}{c_1^2}+1\right)^{l+1-k}\eta_i^{l+1,k}\left\|s_i^k\right\|.
$$

Step 2: Eq. (20a) based on adjoint Jacobian update
Let us continue this proof by induction for Eq. (20a), based on the adjoint Jacobian update formula. First, we derive the following error bound for the adjoint Jacobian update in case $l = k + 1$ in Eq. (20a)

$$
\left\|y_i^k - A_i^{k+1}s_i^k\right\| = \left\|y_i^k - A_i^k s_i^k - \frac{1}{(\sigma_i^k, \rho_i^k)}\rho_i^k\tau_i^{k\top}s_i^k\right\| = \left\|\rho_i^k - \frac{1}{(\sigma_i^k, \rho_i^k)}\rho_i^k\tau_i^{k\top}s_i^k\right\|
$$
$$
= \left|1 - \frac{(\tau_i^k, s_i^k)}{(\sigma_i^k, \rho_i^k)}\right|\left\|\rho_i^k\right\| = \left|\frac{(\sigma_i^k, y_i^k - A_i^k s_i^k) - (\gamma_i^k - A_i^{k\top}\sigma_i^k, s_i^k)}{(\sigma_i^k, \rho_i^k)}\right|\left\|\rho_i^k\right\|
$$
$$
= \frac{\left|(\sigma_i^k, y_i^k) - (\gamma_i^k, s_i^k)\right|}{\left|(\sigma_i^k, \rho_i^k)\right|}\left\|\rho_i^k\right\|.
$$
$$
\tag{25}
$$

From the skipping conditions in (14)-(15), we know that $\left|(\sigma_i^k, \rho_i^k)\right| \geq c_1\left\|s_i^k\right\|\left\|\tau_i^k\right\|$ and $\left\|\rho_i^k\right\| \leq \frac{\left|(\tau_i^k, s_i^k)\right|}{c_1\left\|\sigma_i^k\right\|} \leq \frac{\left\|\tau_i^k\right\|\left\|s_i^k\right\|}{c_1\left\|\sigma_i^k\right\|}$ holds. We can use these lower and upper bounds to rewrite the latter expression as

$$
\left\|y_i^k - A_i^{k+1}s_i^k\right\| = \frac{\left|(\sigma_i^k, y_i^k) - (\gamma_i^k, s_i^k)\right|}{\left|(\sigma_i^k, \rho_i^k)\right|}\left\|\rho_i^k\right\| \leq \frac{\left|(\sigma_i^k, y_i^k) - (\gamma_i^k, s_i^k)\right|}{c_1\left\|s_i^k\right\|\left\|\tau_i^k\right\|}\left\|\rho_i^k\right\|
$$
$$
\leq \frac{\left|(\sigma_i^k, y_i^k) - (\gamma_i^k, s_i^k)\right|}{c_1^2\left\|\sigma_i^k\right\|}
\tag{26}
$$
$$
\leq \frac{c_3}{c_1^2}\left\|s_i^k\right\|^2,
$$

11

where we additionally used the result

$$\left|(\gamma_i^k, s_i^k) - (\sigma_i^k, y_i^k)\right| = \left|\sigma_i^{k\top}\left(\frac{\partial F_i}{\partial w}(w_i^{k+1}) - \int_0^1 \frac{\partial F_i}{\partial w}(w_i^k + t\, s_i^k)\mathrm{d}t\right) s_i^k\right| \tag{27}$$
$$\leq c_3 \left\|\sigma_i^k\right\| \left\|s_i^k\right\|^2.$$

Note that $\eta_i^{k+1,k} = \left\|s_i^k\right\|$ such that Eq. (20a) holds in case $l = k+1$. Assume that the result in (20a) holds for all $j = k+1, \dots, l$. Then, we have the following

$$\|y_i^k - A_i^{l+1} s_i^k\| = \left\|y_i^k - A_i^l s_i^k - \alpha_{\mathrm{A},i}^l \rho_i^l \tau_i^{l\top} s_i^k\right\|$$
$$\leq \|y_i^k - A_i^l s_i^k\| + \left|\frac{(\tau_i^l, s_i^k)}{(\sigma_i^l, \rho_i^l)}\right| \|\rho_i^l\|, \tag{28}$$

for the adjoint Jacobian update formula in which $\alpha_{\mathrm{A},i} = \frac{1}{(\sigma_i^l, \rho_i^l)}$. From the skipping conditions in (14)-(15), we know that $\left|(\sigma_i^l, \rho_i^l)\right| \geq c_1 \left\|s_i^l\right\| \left\|\tau_i^l\right\|$ and $\left\|\rho_i^l\right\| \leq \frac{|(\tau_i^l, s_i^l)|}{c_1 \|\sigma_i^l\|} \leq \frac{\|\tau_i^l\| \|s_i^l\|}{c_1 \|\sigma_i^l\|}$ holds such that $\frac{\|\sigma_i^l\| \|\rho_i^l\|}{|(\sigma_i^l, \rho_i^l)|} \leq \frac{1}{c_1} \frac{\|\sigma_i^l\| \|\rho_i^l\|}{\|s_i^l\| \|\tau_i^l\|} \leq \frac{1}{c_1^2}$. In addition, given eqs. (22) and (23) and given that $\eta_i^{l,k} \leq \eta_i^{l+1,k}$, we obtain

$$\left\|y_i^k - A_i^{l+1} s_i^k\right\| \leq \frac{c_3}{c_1}\left(\frac{2}{c_1^2} + 1\right)^{l-k} \eta_i^{l,k} \left\|s_i^k\right\|$$
$$+ \left(c_3 \eta_i^{l+1,k} + \frac{c_3}{c_1}\left(\frac{2}{c_1^2} + 1\right)^{l-k} \eta_i^{l,k}\right) \frac{\|\sigma_i^l\| \|s_i^k\|}{|(\sigma_i^l, \rho_i^l)|} \left\|\rho_i^l\right\|$$
$$\leq \frac{c_3}{c_1}\left(\frac{1}{c_1} + \left(\frac{1}{c_1^2} + 1\right)\left(\frac{2}{c_1^2} + 1\right)^{l-k}\right) \eta_i^{l+1,k} \left\|s_i^k\right\| \tag{29}$$
$$\leq \frac{c_3}{c_1}\left(\frac{2}{c_1^2} + 1\right)^{l+1-k} \eta_i^{l+1,k} \left\|s_i^k\right\|.$$

Note that the induction proof of step 1 and 2 implies that Eq. (20a) additionally holds when switching between the forward and adjoint Jacobian update formulas. A similar induction-based proof can be used to show the result of Eq. (20b) for the dynamic block-TR1 Jacobian updates. $\qquad\square$

Now, we present the resulting theorem on the convergence of the Jacobian approximation for the block-wise TR1 scheme under any sequence of decision rules that select the adjoint or forward updates at every iteration k and for each block $i = 0, \dots, N-1$.

Theorem 3.5. *Given (AS1)-(AS6) in Assumption 3.3, then the following holds for each Jacobian block matrix $i = 0, \dots, N-1$*

$$\lim_{k\to\infty}\left\|\left(A_i^k - \frac{\partial F_i}{\partial w_i}(w_i^*)\right) N_i\right\| = 0, \tag{30}$$

such that the following holds for the complete Jacobian approximation

$$\lim_{k \to \infty} \left\| \left(A^k - \frac{\partial F}{\partial w}(w^*) \right) N \right\| = 0. \tag{31}$$

Proof. Based on the inequality $\|w_i^p - w_i^s\| \leq \|w_i^p - w_i^*\| + \|w_i^s - w_i^*\|$ and using the definition $\eta_i^{l,k} = \max\{\|w_i^p - w_i^s\| \mid k \leq s \leq p \leq l\}$, one obtains

$$\eta_i^{k+l+1,k} \leq 2\,\nu_i^k \quad \text{for} \quad \nu_i^k = \max\{\|w_i^s - w_i^*\| \mid k \leq s \leq k+l+1\}, \tag{32}$$

for $l \geq q_i$ and q_i is defined as in Assumption 3.3. In addition, the following holds

$$\left\| y_i^j - \frac{\partial F_i}{\partial w_i}(w_i^*)s_i^j \right\| = \left\| \left(\int_0^1 \frac{\partial F_i}{\partial w_i}(w_i^j + ts_i^j)\mathrm{d}t \right) s_i^j - \frac{\partial F_i}{\partial w_i}(w_i^*)s_i^j \right\| \tag{33a}$$

$$= \left\| \left(\int_0^1 \frac{\partial F_i}{\partial w_i}(w_i^j + ts_i^j)\mathrm{d}t - \frac{\partial F_i}{\partial w_i}(w_i^*) \right) s_i^j \right\| \tag{33b}$$

$$\leq c_3 \nu_i^k \left\| s_i^j \right\|, \tag{33c}$$

at an iteration j, where $k \leq j \leq k+l$, regardless of whether the forward or adjoint Jacobian update formula has been used. Moreover, from Lemma 3.4, we have that

$$\left\| y_i^j - A_i^{k+l+1}s_i^j \right\| \leq \frac{c_3}{c_1} \left(\frac{2}{c_1^2} + 1 \right)^{k+l+1-j} \eta_i^{k+l+1,j} \|s_i^j\|, \quad k \leq j \leq k+l,$$
$$\leq 2\frac{c_3}{c_1} \left(\frac{2}{c_1^2} + 1 \right)^{l+1} \nu_i^k \|s_i^j\|. \tag{34}$$

We use the triangle inequality to obtain

$$\left\| \left(A_i^{k+l+1} - \frac{\partial F_i}{\partial w_i}(w_i^*) \right) \frac{s_i^j}{\|s_i^j\|} \right\| \leq \frac{1}{\|s_i^j\|} \left(\left\| y_i^j - A_i^{k+l+1}s_i^j \right\| + \left\| y_i^j - \frac{\partial F_i}{\partial w_i}(w_i^*)s_i^j \right\| \right) \tag{35a}$$

$$\leq \left(2\frac{c_3}{c_1} \left(\frac{2}{c_1^2} + 1 \right)^{l+1} + c_3 \right) \nu_i^k, \tag{35b}$$

which holds for a sequence of indices $j = k_i^1, \ldots, k_i^{q_i}$. Then, we use the linear independence condition $(AS6)$ in Assumption 3.3 that guarantees both existence of the inverse $(S_{N_i}^{k_i})^{-1}$ and the upper bound $\|(S_{N_i}^{k_i})^{-1}\| \leq 1/c_4$, such that

$$\left\| \left(A_i^{k+l+1} - \frac{\partial F_i}{\partial w_i}(w_i^*) \right) N_i \right\| \leq \frac{1}{c_4} \left\| \left(A_i^{k+l+1} - \frac{\partial F_i}{\partial w_i}(w_i^*) \right) N_i S_{N_i}^{k_i} \right\| \tag{36a}$$

$$\leq c_5 \, \nu_i^k, \tag{36b}$$

13

where $c_5 = \frac{c_3}{c_4} \left(\frac{2}{c_1} \left(\frac{2}{c_1^2} + 1 \right)^{l+1} + 1 \right) \sqrt{q_i}$ has been defined. Lastly, the result in Eq. (30) follows from the fact that assumption $(AS4)$ implies that ν_i^k tends to zero. Note that this asymptotic result holds regardless of which Jacobian update (adjoint or forward TR1 formula) is performed for each block $i = 0, \ldots, N - 1$. The same convergence result then holds for the complete Jacobian matrix in (31), based on separability of the active inequality constraints and of the nonlinear constraint functions.

$\square$

3.3. Local rate of linear convergence for Gauss-Newton based SQP

One iteration of the adjoint-based Gauss-Newton SQP method solves the linear system in Eq. (6), which can be written in the following compact form

$$\tilde{J}_{\mathrm{IN}}(z^k)\Delta z = -\mathcal{F}(z^k), \tag{37}$$

where $\mathcal{F}(\cdot)$ denotes the KKT optimality conditions in the right-hand side of Eq. (6). Let us define regularity for a local minimizer $z^\star := (w^\star, \lambda^\star, \mu^\star)$ of the NLP, given a particular set of active inequality constraints. For this purpose, we rely on the linear independence constraint qualification (LICQ) and the second order sufficient conditions (SOSC) for optimality, of which the latter requires that the Hessian of the Lagrangian is strictly positive definite in the directions of the critical cone [31].

Definition 3.6. A minimizer of an equality constrained NLP is called a regular KKT point, if both LICQ and SOSC are satisfied at this KKT point.

The convergence of this Newton-type optimization method then follows the classical and well-known local contraction theorem from [9, 15, 20, 32]. We use a particular version of this theorem from [19, 34], providing sufficient and necessary conditions for the existence of a neighbourhood of the solution where the Newton-type iteration converges locally. Let $\rho(P)$ denote the spectral radius, i.e., the maximum absolute value of the eigenvalues for the square matrix P.

Theorem 3.7 (Local Newton-type contraction [19])**.** *We consider the twice continuously differentiable function $\mathcal{F}(z)$ from Eq. (6) and the regular KKT point $\mathcal{F}(z^\star) = 0$ from Definition 3.6. We then apply the Newton-type iteration in Eq. (37), where $\tilde{J}_{\mathrm{IN}}(z) \approx J(z)$ is additionally assumed to be continuously differentiable and invertible in a neighbourhood of the solution. If all eigenvalues of the iteration matrix have a modulus smaller than one, i.e., if the spectral radius satisfies*

$$\kappa^\star := \rho \left(\tilde{J}_{\mathrm{IN}}(z^\star)^{-1} J(z^\star) - \mathbb{1} \right) < 1, \tag{38}$$

then this fixed point $z^\star$ is asymptotically stable. Additionally, the iterates z^k converge linearly to the KKT point $z^\star$ with the asymptotic contraction rate $\kappa^\star$ when initialized sufficiently close. On the other hand, the fixed point $z^\star$ is unstable if $\kappa^\star > 1$.

A proof for Theorem 3.7 can be found in [19, 36], based on nonlinear systems theory. Using this result, let us define the linear contraction rate for a Gauss-Newton method

with exact Jacobian information

$$\kappa_{\text{GN}}^\star := \rho \left(\begin{bmatrix} H & (\frac{\partial F}{\partial w} - E)^\top & P_A^\top \\ \frac{\partial F}{\partial w} - E & & \\ P_A & & \end{bmatrix}^{-1} \begin{bmatrix} \nabla_w^2 \mathcal{L} & (\frac{\partial F}{\partial w} - E)^\top & P_A^\top \\ \frac{\partial F}{\partial w} - E & & \\ P_A & & \end{bmatrix} - \mathbb{1} \right) < 1,$$

(39)

at the local solution point $z^\star := (w^\star, \lambda^\star, \mu^\star)$ of the KKT conditions. In what follows, we show that the local contraction rate for the block-TR1 Gauss-Newton SQP method

$$\kappa_{\text{BTR1}}^\star := \rho \left(\begin{bmatrix} H & (A - E)^\top & P_A^\top \\ A - E & & \\ P_A & & \end{bmatrix}^{-1} \begin{bmatrix} \nabla_w^2 \mathcal{L} & (\frac{\partial F}{\partial w} - E)^\top & P_A^\top \\ \frac{\partial F}{\partial w} - E & & \\ P_A & & \end{bmatrix} - \mathbb{1} \right) < 1,$$

(40)

coincides with the exact Jacobian based linear convergence rate in (39). The following result states that the eigenspectrum of the iteration matrix $\tilde{J}_{\text{IN}}(z^\star)^{-1} J(z^\star) - \mathbb{1}$ at the solution point $z^\star := (w^\star, \lambda^\star, \mu^\star)$ coincides with the eigenspectrum of the iteration matrix $\tilde{J}_{\text{GN}}(z^\star)^{-1} J(z^\star) - \mathbb{1}$, using the notation $\sigma(P)$ to denote the spectrum, i.e., the set of eigenvalues for a matrix P.

Lemma 3.8. *For a regular KKT point $z^\star := (w^\star, \lambda^\star, \mu^\star)$, the eigenvalues of the block-TR1 based iteration matrix $\tilde{J}_{\text{IN}}(z^\star)^{-1} J(z^\star) - \mathbb{1}$ satisfy*

$$\sigma \left(\tilde{J}_{\text{IN}}(z^\star)^{-1} J(z^\star) - \mathbb{1} \right) = \sigma \left(\tilde{J}_{\text{GN}}(z^\star)^{-1} J(z^\star) - \mathbb{1} \right).$$

(41)

Proof. Let us define the eigenvalues s of the iteration matrix $\tilde{J}_{\text{IN}}(z^\star)^{-1} J(z^\star) - \mathbb{1}$ as the zeros of

$$\det \left(\tilde{J}_{\text{IN}}(z^\star)^{-1} J(z^\star) - (s+1)\mathbb{1} \right) = 0,$$

(42)

which, given that the Jacobian approximation $\tilde{J}_{\text{IN}}$ is invertible, this is equivalent to

$$\det \left(J(z^\star) - (s+1)\tilde{J}_{\text{IN}}(z^\star) \right) = 0.$$

(43)

This block matrix then reads as

$$J(z^\star) - (s+1)\tilde{J}_{\text{IN}}(z^\star) = \begin{bmatrix} \nabla_w^2 \mathcal{L} - (s+1)H & (\frac{\partial F}{\partial w} - (s+1)A)^\top + sE^\top & -sP_A^\top \\ (\frac{\partial F}{\partial w} - (s+1)A) + sE & & \\ -sP_A & & \end{bmatrix}.$$

(44)

The result follows from Theorem 3.5 that claims the following asymptotic result for the block-TR1 based Jacobian approximation

$$\lim_{k \to \infty} \left(A^k - \frac{\partial F}{\partial w}(w^*) \right) N = \left(A - \frac{\partial F}{\partial w}(w^*) \right) N = 0,$$

(45)

where N is a null space matrix with appropriate dimensions and orthonormal column

15

vectors such that $N^\top N = \mathbb{1}$ and $P_A\, N = 0$. We rewrite Eq. (43) as follows

$$\det\left(J(z^\star) - (s+1)\tilde{J}_{\mathrm{IN}}(z^\star)\right) =$$

$$(-s)^{2\,n_A}\det\left(\begin{bmatrix} \nabla_w^2\mathcal{L} - (s+1)H & \left(\frac{\partial F}{\partial w} - (s+1)A\right)^\top + sE^\top & P_A^\top \\ \left(\frac{\partial F}{\partial w} - (s+1)A\right) + sE & & \\ P_A & & \end{bmatrix}\right). \qquad (46)$$

It can be verified that $\det\left(J(z^\star) - (s+1)\tilde{J}_{\mathrm{IN}}(z^\star)\right) = 0$ holds for $s = 0$ with an algebraic multiplicity of $2\,n_A$ as well as for the values of s that satisfy

$$\det\left(\begin{bmatrix} N^\top & \mathbb{0} \\ \mathbb{0} & \mathbb{1} \end{bmatrix}\begin{bmatrix} \nabla_w^2\mathcal{L} - (s+1)H & \left(\frac{\partial F}{\partial w} - (s+1)A\right)^\top + sE^\top \\ \left(\frac{\partial F}{\partial w} - (s+1)A\right) + sE & \mathbb{0} \end{bmatrix}\begin{bmatrix} N & \mathbb{0} \\ \mathbb{0} & \mathbb{1} \end{bmatrix}\right)$$

$$= \det\left(\begin{bmatrix} N^\top \Delta H\, N & N^\top\left(\frac{\partial F}{\partial w} - (s+1)A\right)^\top + sN^\top E^\top \\ \left(\frac{\partial F}{\partial w} - (s+1)A\right) N + sE\, N & \mathbb{0} \end{bmatrix}\right) \qquad (47)$$

$$= (-s)^{2\,n_F}\det\left(\begin{bmatrix} N^\top \Delta H\, N & N^\top\left(\frac{\partial F}{\partial w} - E\right)^\top \\ \left(\frac{\partial F}{\partial w} - E\right) N & \mathbb{0} \end{bmatrix}\right) = 0,$$

in the limit for $k \to \infty$, where the compact notation $\Delta H := \left(\nabla_w^2\mathcal{L} - (s+1)H\right)$ has been used for the Gauss-Newton Hessian approximation. Therefore, the eigenvalues of the iteration matrix $\tilde{J}_{\mathrm{IN}}(z^\star)^{-1}J(z^\star) - \mathbb{1}$ for the proposed block-TR1 approach, evaluated at a regular KKT point, are equal to the eigenvalues of the iteration matrix $\tilde{J}_{\mathrm{GN}}(z^\star)^{-1}J(z^\star) - \mathbb{1}$ for the exact Jacobian based Gauss-Newton method.

$\square$

Corollary 3.9. *Based on Lemma 3.8, the linear contraction rate for the block-TR1 based optimization algorithm coincides with the linear contraction rate of the exact Jacobian based Gauss-Newton method $\kappa_{\mathrm{BTR1}}^\star = \kappa_{\mathrm{GN}}^\star$, when the iterates are sufficiently close to the regular KKT point $z^\star := (w^\star, \lambda^\star, \mu^\star)$.*

3.4. Superlinear convergence for SQP with quasi-Newton Hessian updates

Even though the majority of this article is focused on the generalized Gauss-Newton method for nonlinear least squares type optimization problems that occur frequently in predictive control applications, note that superlinear convergence results can be recovered when a block-structure preserving quasi-Newton method is additionally used to approximate the Hessian of the Lagrangian. For example, let us consider the following lemma that represents a block-structured or *partitioned* version [23, 24] of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) [1] or the symmetric rank-one (SR1) formula [13, 29] to approximate the block-diagonal Hessian matrix.

Theorem 3.10. *Given (AS1)-(AS6) in Assumption 3.3, then the following holds for each Hessian block matrix approximation*

$$\lim_{k\to\infty}\left\|\left(H_i^k - \nabla_{w_i w_i}^2\mathcal{L}(w_i^\star, \lambda_i^\star)\right) N_i\right\| = 0, \qquad (48)$$

16

$i = 0, \ldots, N-1$, *such that the following holds for the complete Hessian approximation*

$$\lim_{k \to \infty} \left\| \left(H^k - \nabla^2_{ww} \mathcal{L}(w^\star, \lambda^\star) \right) N \right\| = 0. \tag{49}$$

Theorem 3.10 on the convergence of a separable quasi-Newton type Hessian approximation method in combination with our main result in Theorem 3.5 on the block-structured quasi-Newton type Jacobian update formula can be used directly to prove the following result on convergence of the reduced KKT matrix.

Theorem 3.11. *Given (AS1)-(AS6) in Assumption 3.3, the following holds*

$$\lim_{k \to \infty} \left\| \begin{bmatrix} N^\top H^k N & N^\top A^{k\top} \\ A^k N & \mathbb{0} \end{bmatrix} - \begin{bmatrix} N^\top \nabla^2_{ww} \mathcal{L}(w^\star, \lambda^\star) N & N^\top \frac{\partial F}{\partial w}(w^\star)^\top \\ \frac{\partial F}{\partial w}(w^\star) N & \mathbb{0} \end{bmatrix} \right\| = 0. \tag{50}$$

Based on Theorem 3.2, the above result ensures q-superlinear convergence of the SQP iterates when using a quasi-Newton method to update both the block-structured Hessian and Jacobian matrices. The proof for Theorem 3.11, based on the intermediate convergence results in Theorem 3.5 and 3.10 can be found in [20].

4. Lifted Collocation Algorithm with Block-TR1 Jacobian Updates

As mentioned earlier, implicit integration schemes are often used in direct optimal control because of their relatively high order of accuracy and their improved numerical stability properties [26]. More specifically, problem formulations based on a system of stiff and/or implicit differential or differential-algebraic equations require the use of an implicit integration scheme. Collocation methods are a popular family of implicit Runge-Kutta methods. This section presents a novel lifted collocation algorithm based on tailored block-TR1 Jacobian updates. The standard lifted collocation method with exact Jacobian information was proposed in [37] as a structure-exploiting implementation of direct collocation, even though it shows similarities to multiple shooting.

4.1. Direct collocation for nonlinear optimal control

In direct transcription methods, such as direct collocation [3, 5], the integration scheme and its intermediate variables are directly made part of the nonlinear optimization problem. In this context, where the simulation routine is defined implicitly as part of the equality constraints in the dynamic optimization problem, one typically relies on implicit integration schemes for their relatively high order of accuracy and improved numerical stability properties. The discrete-time optimal control problem can generally

be written as

$$\min_{X, U, K} \quad \sum_{i=0}^{N-1} l_i(x_i, u_i) + l_N(x_N) \tag{51a}$$

$$\text{s.t.} \quad \hat{x}_0 = x_0, \tag{51b}$$

$$x_i + B_i K_i = x_{i+1}, \quad i = 0, \ldots, N-1, \tag{51c}$$

$$G_i(x_i, u_i, K_i) = 0, \quad i = 0, \ldots, N-1, \tag{51d}$$

$$P_i w_i \leq p_i, \quad i = 0, \ldots, N, \tag{51e}$$

where the additional trajectory $K = [K_0^\top, \ldots, K_{N-1}^\top]^\top$ denotes the intermediate variables of the numerical integration method. These variables are defined implicitly by the equations in (51d), such that the continuity condition reads as in Eq. (51c). More specifically, the Jacobian $\frac{\partial G_i}{\partial K_i}(\cdot)$ will generally be invertible for an integration scheme applied to a well-defined set of differential equations in (1c). A popular approach of this type is better known as direct collocation [4]. It relies on a collocation method, a subclass of implicit Runge-Kutta (IRK) methods [26], to accurately discretize the continuous time dynamics. In this case, the equations in (51d) define the collocation polynomial on each control interval $i = 0, \ldots, N-1$.

In a similar fashion as in Section 2, the adjoint-based SQP method can be applied directly to the direct collocation problem in (51) by solving the following convex QP subproblem at each iteration

$$\min_{\Delta W, \Delta K} \quad \sum_{i=0}^{N} \frac{1}{2} \Delta w_i^\top H_i^k \Delta w_i + h_i^{c\top} \begin{bmatrix} \Delta w_i \\ \Delta K_i \end{bmatrix} \tag{52a}$$

$$\text{s.t.} \quad \Delta x_0 = \hat{x}_0 - x_0^k, \tag{52b}$$

$$e_i^k + \Delta x_i + B_i \Delta K_i = \Delta x_{i+1}, \quad i = 0, \ldots, N-1, \tag{52c}$$

$$c_i^k + D_i^k \Delta w_i + C_i^k \Delta K_i = 0, \quad i = 0, \ldots, N-1, \tag{52d}$$

$$P_i \Delta w_i \leq p_i^k, \quad i = 0, \ldots, N, \tag{52e}$$

based on $c_i^k := G_i(w_i^k, K_i^k)$ and the Jacobian approximations $D_i^k \approx \frac{\partial G_i}{\partial w_i}(w_i^k, K_i^k)$ and $C_i^k \approx \frac{\partial G_i}{\partial K_i}(w_i^k, K_i^k)$. The corresponding gradient correction reads as

$$h_i^c := \begin{bmatrix} \nabla_{w_i} l(w_i^k) + \left(\frac{\partial G_i}{\partial w_i}(w_i^k, K_i^k) - D_i^k \right)^\top \omega_i^k \\ \left(\frac{\partial G_i}{\partial K_i}(w_i^k, K_i^k) - C_i^k \right)^\top \omega_i^k \end{bmatrix}, \tag{53}$$

where ω_i^k denotes the current value of the multipliers for the nonlinear constraints in (51d) and λ_i^k again denotes the multipliers for the continuity constraints in (51c).

4.2. Tailored structure exploitation for direct collocation

As mentioned earlier, the Jacobian matrix $\frac{\partial G_i}{\partial K_i}$ for the collocation equations needs to be invertible. Therefore, given an invertible approximation $C_i^k \approx \frac{\partial G_i}{\partial K_i}(w_i^k, K_i^k)$, we can

rewrite the linearized expression in Eq (52d) as follows

$$\Delta K_i = -C_i^{k^{-1}} \left(c_i^k + D_i^k \Delta w_i \right). \tag{54}$$

By substituting the above expression for ΔK_i back into the direct collocation structured QP in (52), one obtains the condensed but equivalent formulation

$$\min_{\Delta W} \quad \sum_{i=0}^{N} \frac{1}{2} \Delta w_i^\top H_i^k \, \Delta w_i \, + \, \tilde{h}_i^{c^\top} \Delta w_i \tag{55a}$$

$$\text{s.t.} \quad \Delta x_0 \; = \; \hat{x}_0 - x_0^k, \tag{55b}$$

$$d_i^k + \Delta x_i - B_i \, C_i^{k^{-1}} D_i^k \Delta w_i \; = \; \Delta x_{i+1}, \quad i = 0, \dots, N-1, \tag{55c}$$

$$P_i \, \Delta w_i \leq p_i^k, \qquad\qquad\qquad i = 0, \dots, N, \tag{55d}$$

where $d_i^k = e_i^k - B_i \, C_i^{k^{-1}} c_i^k$ is defined and the condensed gradient reads as

$$\tilde{h}_i^c = \nabla_{w_i} l(w_i^k) + \left(\frac{\partial G_i}{\partial w_i} - \frac{\partial G_i}{\partial K_i} C_i^{k^{-1}} D_i^k \right)^\top \omega_i^k, \tag{56}$$

given the original gradient correction in (53).

Note that the resulting QP formulation in Eq. (55) is of the same problem dimensions and exhibits the same sparsity as the multiple shooting structured QP subproblem in Eq. (7). Therefore, state of the art block-structured QP solvers can be used, for which an overview can be found in [21]. After solving the condensed QP in (55), the collocation variables can be obtained from the expansion step in Eq. (54). Based on the optimality conditions of the original direct collocation structured QP in (52), the corresponding Lagrange multipliers can be updated as follows

$$\omega_i^{k+1} = \omega_i^k - C_i^{k^{-\top}} \left(\frac{\partial G_i}{\partial K_i}^\top \omega_i^k + B_i^\top \lambda_i^{k+1} \right), \tag{57}$$

where λ_i^{k+1} denote the new values of the Lagrange multipliers for the continuity conditions in (55c) or in (52c).

4.3. Block-TR1 Jacobian update for lifted collocation

The block-TR1 update formula from Eq. (11) can be readily applied to the direct collocation equations, resulting in

$$[D_i^{k+1} \; C_i^{k+1}] = [D_i^k \; C_i^k] + \alpha_i^k \left(y_i^k - [D_i^k \; C_i^k] s_i^k \right) \left(\gamma_i^{k^\top} - \sigma_i^{k^\top} [D_i^k \; C_i^k] \right), \tag{58}$$

where the quantities $\gamma_i^{k^\top} = \sigma_i^{k^\top} \frac{\partial G_i}{\partial(w_i, K_i)}(w_i^{k+1}, K_i^{k+1})$ and $\sigma_i^k = \omega_i^{k+1} - \omega_i^k$ are defined. In addition, $s_i^k := \begin{bmatrix} w_i^{k+1} - w_i^k \\ K_i^{k+1} - K_i^k \end{bmatrix}$ and $y_i^k = G_i(w_i^{k+1}, K_i^{k+1}) - G_i(w_i^k, K_i^k)$ is defined. In order to use this block-TR1 update formula in combination with the lifted collocation method, one needs to be able to efficiently form the condensed QP in Eq. (55). For this

purpose, we need to avoid the costly computations of the inverse matrix $C_i^{k^{-1}}$ as well as the matrix-matrix multiplication $C_i^{k^{-1}} D_i^k$. In what follows, we present a procedure to directly obtain a rank-one update formula for the inverse matrix $C_i^{k+1^{-1}}$ and for the corresponding product $E_i^{k+1} := C_i^{k+1^{-1}} D_i^{k+1}$.

Based on the Sherman-Morrison formula, one can directly update the matrix inverse given the previous invertible approximation $C_i^{k^{-1}} \approx \frac{\partial G_i}{\partial K_i}^{-1}$. Let us first rewrite the block-TR1 update from Eq. (58) as follows

$$D_i^{k+1} = D_i^k + \alpha_i^k \, \rho_i^k \, \tau_{\mathrm{D},i}^{k^\top} \quad \text{and} \quad C_i^{k+1} = C_i^k + \alpha_i^k \, \rho_i^k \, \tau_{\mathrm{C},i}^{k^\top}, \tag{59}$$

where $\rho_i^k = y_i^k - [D_i^k \ C_i^k] s_i^k$ and $[\tau_{\mathrm{D},i}^{k^\top} \ \tau_{\mathrm{C},i}^{k^\top}] = \gamma_i^{k^\top} - \sigma_i^{k^\top} [D_i^k \ C_i^k]$. The Sherman-Morrison formula then reads as

$$C_i^{k+1^{-1}} = C_i^{k^{-1}} - \alpha_i^k \beta_i^k \, C_i^{k^{-1}} \rho_i^k \tau_{\mathrm{C},i}^{k^\top} C_i^{k^{-1}}, \tag{60}$$

where $\beta_i^k = \frac{1}{1 + \alpha_i^k \tau_{\mathrm{C},i}^{k^\top} C_i^{k^{-1}} \rho_i^k}$. Let us define $\tilde{\rho}_i^k = C_i^{k^{-1}} \rho_i^k$ such that we obtain the following update for the condensed Jacobian

$$
\begin{aligned}
E_i^{k+1} = C_i^{k+1^{-1}} D_i^{k+1} &= C_i^{k^{-1}} \left(D_i^k + \alpha_i^k \rho_i^k \tau_{\mathrm{D},i}^{k^\top} \right) \\
&\quad - \alpha_i^k \beta_i^k C_i^{k^{-1}} \rho_i^k \tau_{\mathrm{C},i}^{k^\top} C_i^{k^{-1}} \left(D_i^k + \alpha_i^k \rho_i^k \tau_{\mathrm{D},i}^{k^\top} \right) \\
&= E_i^k + \alpha_i^k \tilde{\rho}_i^k \tau_{\mathrm{D},i}^{k^\top} - \alpha_i^k \beta_i^k \tilde{\rho}_i^k \tau_{\mathrm{C},i}^{k^\top} (E_i^k + \alpha_i^k \tilde{\rho}_i^k \tau_{\mathrm{D},i}^{k^\top}) \\
&= E_i^k + \alpha_i^k \tilde{\rho}_i^k \tilde{\tau}_i^{k^\top},
\end{aligned}
\tag{61}
$$

where $\tilde{\tau}_i^{k^\top} = \tau_{\mathrm{D},i}^{k^\top} - \beta_i^k \tau_{\mathrm{C},i}^{k^\top} (E_i^k + \alpha_i^k \tilde{\rho}_i^k \tau_{\mathrm{D},i}^{k^\top})$ has been defined. It is readily seen that the update for E_i^k in Eq. (61) is a rank-one update for the condensed Jacobian matrix. As proposed in [27], corresponding low-rank update formulas for the condensed Hessian can be obtained for the special case of a pseudospectral method based on a global collocation polynomial.

4.4. Lifted collocation SQP method with block-TR1 Jacobian updates

It is important to stress that the novel block-TR1 update formula for the condensed Jacobian matrix $E_i^{k+1} = C_i^{k+1^{-1}} D_i^{k+1}$ in Eq. (61) provides an efficient manner to directly compute the rank-one update to the matrices in the condensed QP formulation of Eq. (55), without the need for a matrix factorization, inversion and without any matrix-matrix multiplications. Instead, the proposed implementation merely requires matrix-vector multiplications and outer products, resulting in a quadratic instead of cubic computational complexity with respect to the number of optimization variables within each control interval. However, this comes at the cost of a slightly increased memory footprint, since additionally the matrices C_i^{-1} and E_i need to be stored from one iteration to the next. The implementation of the lifted block-TR1 based SQP method for direct collocation is presented in Algorithm 2.

Algorithm 2 One lifted collocation SQP iteration with block-wise TR1 updates.

Input: $w_i^k = (x_i^k, u_i^k)$, K_i^k, λ_i^k, ω_i^k, C_i^k, D_i^k, $C_i^{k^{-1}}$ and E_i^k.

    `Problem linearization and QP preparation`

1: Formulate the QP in (55) with Jacobian matrices E_i^k, Gauss-Newton Hessian approximations H_i^k and vectors d_i^k, p_i^k and $\tilde{h}_i^c$ in (56) for $i = 0, \ldots, N-1$.

    `Computation of Newton-type step direction`

2: Solve the QP subproblem in Eq. (55) to update optimization variables:
$w_i^{k+1} \leftarrow w_i^k + \Delta w_i^k$ and $\lambda_i^{k+1} \leftarrow \lambda_i^k + \Delta \lambda_i^k$.   ▷ full step

    `Block-wise TR1 Jacobian updates`

3: **for** $i = 0, \ldots, N-1$ **do in parallel**

4:     Choose $\alpha_i^k = \alpha_{\mathrm{F},i}^k$ or $\alpha_i^k = \alpha_{\mathrm{A},i}^k$ via some decision rule.

5:     $K_i^{k+1} \leftarrow K_i^k - C_i^{k^{-1}} c_i^k - E_i^k \Delta w_i^k$,

6:     $\omega_i^{k+1} \leftarrow \omega_i^k - C_i^{k^{-\top}} \left(\frac{\partial G_i}{\partial K_i}^\top \omega_i^k + B_i^\top \lambda_i^{k+1} \right)$,

7:     $D_i^{k+1} \leftarrow D_i^k + \alpha_i^k \rho_i^k \tau_{\mathrm{D},i}^{k^\top}$ and $C_i^{k+1} \leftarrow C_i^k + \alpha_i^k \rho_i^k \tau_{\mathrm{C},i}^{k^\top}$,

8:     $C_i^{k+1^{-1}} \leftarrow C_i^{k^{-1}} - \alpha_i^k \beta_i^k \tilde{\rho}_i^k \tau_{\mathrm{C},i}^{k^\top} C_i^{k^{-1}}$,

9:     $E_i^{k+1} \leftarrow E_i^k + \alpha_i^k \tilde{\rho}_i^k \tilde{\tau}_i^{k^\top}$.

10: **end for**

Output: w_i^{k+1}, K_i^{k+1}, λ_i^{k+1}, ω_i^{k+1}, C_i^{k+1}, D_i^{k+1}, $C_i^{k+1^{-1}}$ and E_i^{k+1}.

4.5. Convergence results for block-TR1 based lifted collocation

We observe that the TR1 Jacobian updates of the lifted collocation implementation are equivalent to the updates of the direct collocation method. More specifically, the Jacobian approximation matrices are the same at each SQP iteration, regardless of whether we perform the condensing and expansion procedure for the collocation variables in the proposed lifted implementation of Algorithm 2. Therefore, the convergence properties shown in the previous section also hold for both the standard and lifted collocation based block-TR1 SQP method.

Corollary 4.1. *If the assumptions of Theorem 3.1 and Assumption 3.3 hold, then the lifted collocation SQP method with block-wise TR1 Jacobian updates in Algorithm 2, with a Gauss-Newton Hessian approximation, produces iterates $\{w^k, \lambda^k, \mu^k\}$ that converge q-linearly within a neighbourhood around the KKT point (w^*, λ^*, μ^*) of the NLP.*

Proof. It follows from the equivalence of the SQP iterations between the direct and lifted collocation formulation based on the numerical condensing and expansion of the collocation variables in Eq. (54). In particular, the direct collocation QP subproblem (52) is a special case of the QP formulation in (7), with additional intermediate variables and corresponding equations. The block-TR1 Jacobian matrix convergence results of Theorem 3.5 therefore hold for direct collocation as well as for the proposed lifted implementation in Algorithm 2.

                                                      □

5. Numerical Case Studies of Nonlinear Model Predictive Control

In this section, we illustrate numerically how the proposed block-TR1 SQP method can be used in the context of nonlinear MPC using an algorithm implementation based

on the real-time iterations (RTI), as proposed originally in [18] with exact Jacobian information. The approach is based on one block-TR1 SQP iteration per control time step, and using a continuation-based warm starting of the state and control trajectories from one time step to the next [28]. Each iteration consists of two steps:

(1) *Preparation phase*: discretize and linearize the system dynamics, linearize the remaining constraint functions, and evaluate the quadratic objective approximation to build the optimal control structured QP subproblem.

(2) *Feedback phase*: solve the QP to update the current values for all optimization variables and obtain the next control input to apply feedback to the system.

The proposed block-wise TR1 based Jacobian updates in Algorithm 1 and 2 become part of the preparation step, in order to construct the linearized continuity equations. Therefore, the feedback step remains unchanged and the Jacobian updates do not affect the computational delay between obtaining the new state estimate and applying the next control input value to the system.

We validate the closed-loop performance of these novel block-TR1 based RTI algorithms by presenting numerical simulation results for two NMPC case studies. Motivated by real embedded control applications, we present the computation times for the proposed NMPC algorithms using the ARM Cortex-A53 processor in the Raspberry Pi 3. The block-sparse QP solution in the feedback phase will be carried out by the primal active-set method, called `PRESAS`, that was recently presented in [35].

### 5.1. Nonlinear MPC for a chain of spring-connected masses

In our first case study, the control task is to return a chain of n_{m} masses connected with springs to its steady state, starting from a perturbed initial configuration, without hitting a wall that is placed close to the equilibrium state configuration. The mass at one end is fixed, while the control input $u(t) \in \mathbb{R}^3$ to the system is the direct force applied to the mass at the other end of the chain. The state of each free mass $x^j := [p^{j^\top}, v^{j^\top}]^\top \in \mathbb{R}^6$ consists in its position $p^j := [p^j_x, p^j_y, p^j_z]^\top \in \mathbb{R}^3$ and velocity $v^j \in \mathbb{R}^3$ for $j = 1, \ldots, n_{\mathrm{m}} - 1$, such that the dynamic system can be described by the concatenated state vector $x(t) \in \mathbb{R}^{6(n_{\mathrm{m}}-1)}$. Similar to the work in [37], the nonlinear chain of masses can be used to validate the computational performance and scaling of an optimal control algorithm for a range of numbers of masses n_{m}, resulting in a range of different problem dimensions. The nonlinear system dynamics and the resulting optimal control problem formulation can be found in [39].

5.1.1. Local convergence: Gauss-Newton SQP with block-TR1 Jacobian updates

We illustrate the impact of the proposed block-wise TR1 Jacobian updates on the local convergence rate of the resulting inexact adjoint-based SQP algorithm. Figure 1 shows a comparison of the convergence between different SQP variants for the solution of the nonlinear chain of masses OCP. In particular, the comparison includes the exact Jacobian-based SQP method, the standard dense TR1 update [25], and the good and bad Broyden update formulas [12]. For the proposed block-TR1 based SQP implementation, the figure illustrates both the adjoint and forward variant by using, respectively, the scaling factor in (12) and (13). The performance of the block-TR1 method is additionally illustrated for an implementation where α_i is chosen dynamically, depending on which of the two variants results in the largest denominator in order to avoid the need to skip a block-wise Jacobian update.
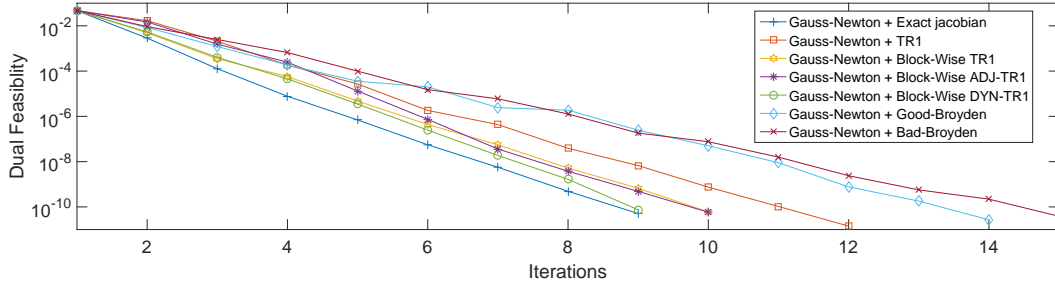
Figure 1. Local convergence analysis: comparison between different variants of the inexact adjoint-based SQP method as described in Algorithm 2 based on lifted collocation, using either the exact Jacobian or different quasi-Newton type Jacobian update formulas for the nonlinear chain of 6 masses.
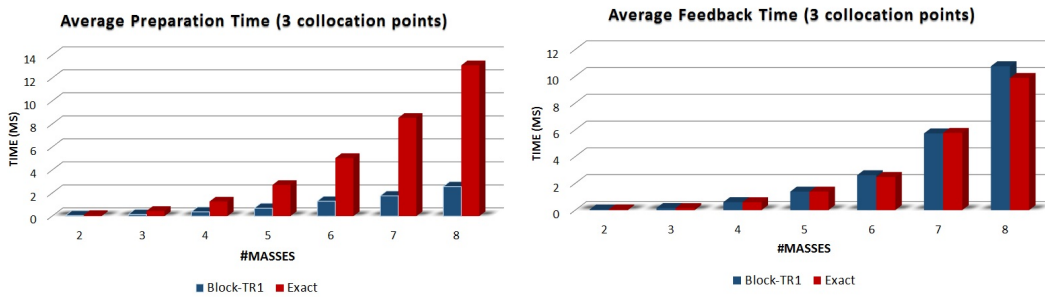
Figure 2. Comparison of the average preparation and feedback computation times (in ms): block-TR1 versus exact Jacobian based lifted collocation SQP method. [2]

It is known that an exact Jacobian-based SQP method with Gauss-Newton type Hessian approximation results in locally linear convergence, for which the asymptotic contraction rate depends on the optimal residual value in the least squares type objective [31]. It can be observed in Figure 1 that all three variants of the proposed block-wise TR1 update formula result in the same asymptotic rate of convergence as for the exact Jacobian based algorithm, i.e., the rate of convergence appears to be the same close to the local solution of the NLP. Note that this confirms numerically the result of Corollary 3.9. In addition, the block-wise TR1 Jacobian updates result in a smaller total number of SQP iterations, compared to the standard dense Jacobian update formulas for the particular example in Figure 1. In the latter case, the direct application of a standard rank-one update formula destroys the block sparsity in the QP subproblems and is therefore computationally unattractive.

5.1.2. Computational timing results for block-TR1 based lifted collocation

Figure 2 illustrates the computation times of both the preparation and feedback steps of an NMPC implementation for a chain of $n_{\mathrm{m}} = 2, \ldots, 8$ masses, using the lifted collocation based SQP method in Algorithm 2. It can be observed that the preparation time scales quadratically with the number of states for the block-TR1 implementation, instead of the cubic computational complexity when using the exact Jacobian. More specifically, the Jacobian evaluation, the factorization and matrix-matrix multiplications are replaced by adjoint differentiation sweeps and matrix-vector operations in Algorithm 2. On the other hand, the feedback time remains essentially the same be-

Table 1. Average computation times (in ms) for nonlinear MPC on a chain of $n_{\mathrm{m}} = 6$ masses, i.e., 30 differential states (4 Gauss collocation nodes versus 10 steps of RK4).

	Explicit (RK4 in Alg. 1)			Implicit (GL4 in Alg. 2)		
	exact	block-TR1		exact	block-TR1	
Linearization	32.36	5.33	**16%**	291.37	35.99	**12%**
QP solution	23.22	37.82		26.33	27.86	
Total RTI step	56.39	43.99	78%	318.58	64.69	20%
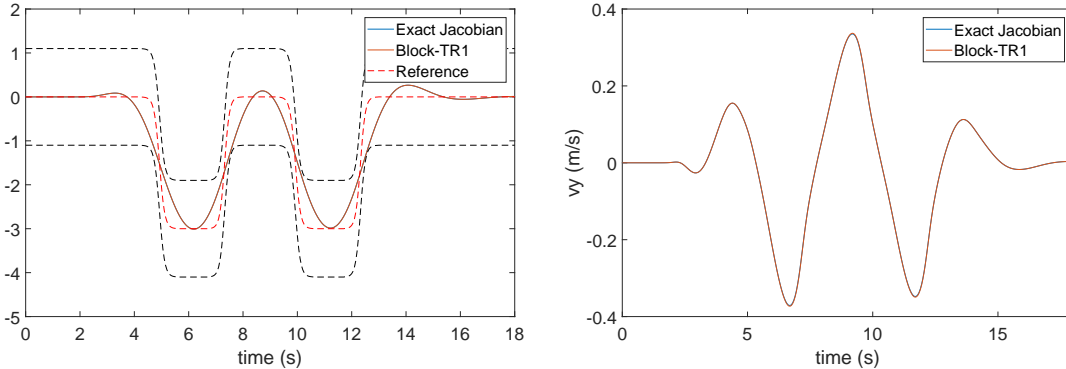
Figure 3. Closed-loop NMPC performance of two double lane changes at a vehicle speed of 10 m/s on snow-covered road conditions, using model parameters from [2] in the nonlinear OCP formulation [33].

cause, after the linearization and QP preparation, both approaches lead to the solution of a similarly structured QP in Eq. (7) or (55).

Table 1 provides a more detailed comparison between the exact Jacobian and the proposed block-TR1 variant of the real-time iterations for NMPC, using an ARM Cortex-A53 processor. The table shows these results for both the explicit Runge-Kutta method of order 4 (RK4) in combination with Algorithm 1 and using the implicit 4-stage Gauss-Legendre (GL4) method within Algorithm 2. The proposed block-TR1 algorithm results in a computational speedup of about factor $6 - 8$ for the problem linearization step. In order to obtain a relatively fair comparison, the number of integration steps for RK4 has been chosen such that the numerical accuracy is close to that of the 4-stage GL method. However, since the system dynamics for the chain of masses are non-stiff, an explicit integration scheme should instead typically perform better in terms of computational efficiency.

5.2. Nonlinear MPC for vehicle control on a snow-covered road

Our second case study considers nonlinear MPC for real-time vehicle control as motivated by automotive applications in autonomous driving. The nonlinear optimal control problem formulation is based on single-track vehicle dynamics with a Pacejka-type tire model [33]. The experimentally validated model parameters can be found in [2]. As often the case in practice, these vehicle dynamics are rather stiff such that an implicit integration scheme should preferably be used. Therefore, it forms an ideal

---

[2]The computation times in Figure 2 have been obtained using an Intel i7-7700k processor @ 4.20 GHz on Windows 10 with 64 GB of RAM.

Table 2. Average computation times (in ms) for vehicle control based on a single-track vehicle model within NMPC (4 Gauss collocation nodes versus 30 steps of RK4).

| | Explicit (RK4 in Alg. 1) | | | Implicit (GL4 in Alg. 2) | | |
	exact	block-TR1		exact	block-TR1	
Linearization	106.73	75.78	**71%**	52.22	18.27	**35%**
QP solution	4.46	4.51		4.59	4.72	
Total RTI step	111.79	80.94	72%	57.43	23.64	41%

case study for the proposed lifted collocation based RTI method of Algorithm 2. Let us perform the closed-loop NMPC simulations as presented in [33], but using the proposed block-TR1 based RTI implementation. We carried out numerical simulations for two successive double lane changes on snow-covered road conditions. The resulting closed-loop trajectories for both the exact Jacobian and the block-TR1 method are indistinguishable from each other, as illustrated in Figure 3.

The corresponding computation times on the ARM Cortex-A53 processor are illustrated in detail by Table 2. Because of the relatively stiff system dynamics, the proposed block-TR1 lifted collocation method from Algorithm 2 becomes attractive and additionally provides a computational speedup of about factor 3 over the standard exact Jacobian based implementation. Note that, even though the Raspberry Pi 3 is not an embedded processor by itself, it uses an ARM core of the same type as those that are used by multiple high-end automotive microprocessors. Therefore, the proposed algorithm implementation as well as the corresponding numerical results form a motivation for real-time embedded control applications that involve a relatively large, implicit and/or stiff system of differential equations.

6. Conclusions and outlook

In this paper, we proposed a block-wise sparsity preserving two-sided rank-one (TR1) Jacobian update for an adjoint-based inexact SQP method to efficiently solve the nonlinear optimal control problems arising in NMPC. We proved local convergence for the block-structured quasi-Newton type Jacobian matrix updates. In case of a Gauss-Newton based SQP implementation, we additionally showed that the asymptotic rate of contraction remains the same. We also presented how this approach can be implemented efficiently in a tailored lifted collocation framework, in order to avoid matrix factorizations and matrix-matrix multiplications. Finally, we illustrated the local convergence properties as well as the computational complexity results numerically for two nonlinear MPC case studies. The effect of the presented contraction properties on the convergence and closed-loop stability of the block-TR1 based real-time iterations is an important topic that is part of ongoing research.

References

[1] J. Asprion, O. Chinellato, and L. Guzzella, *Partitioned Quasi-Newton approximation for direct collocation methods and its application to the fuel-optimal control of a diesel engine*, Journal of Applied Mathematics 2014.

[2] K. Berntorp, B. Olofsson, K. Lundahl, and L. Nielsen, *Models and methodology for optimal trajectory generation in safety-critical road-vehicle manoeuvres*, Vehicle System Dynamics 52 (2014), pp. 1304–1332.

[3] J. Betts, *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, 2nd ed., SIAM, 2010.

[4] L.T. Biegler, *Nonlinear Programming*, MOS-SIAM Series on Optimization, SIAM, 2010.

[5] L. Biegler, *Solution of dynamic optimization problems by successive quadratic programming and orthogonal collocation*, Computers and Chemical Engineering 8 (1984), pp. 243–248.

[6] H.G. Bock, *Recent advances in parameter identification techniques for ODE*, in *Numerical treatment of inverse problems in differential and integral equations*, Springer, 1983, pp. 95–121.

[7] H.G. Bock, M. Diehl, P. Kühl, E. Kostina, J.P. Schlöder, and L. Wirsching, *Numerical methods for efficient and fast nonlinear model predictive control*, in *Proceedings of "Int. Workshop on assessment and future directions of Nonlinear Model Predictive Control"*, Springer, 2005.

[8] H.G. Bock and K.J. Plitt, *A multiple shooting algorithm for direct solution of optimal control problems*, IFAC Proceedings Volumes 17 (1984), pp. 1603–1608.

[9] H. Bock, *Randwertproblemmethoden zur Parameteridentifizierung in Systemen nichtlinearer Differentialgleichungen*, Bonner Mathematische Schriften Vol. 183, Universität Bonn, Bonn, 1987.

[10] P.T. Boggs and J.W. Tolle, *Sequential quadratic programming*, Acta Numerica (1995), pp. 1–51.

[11] C.G. Broyden, *Quasi-Newton methods and their application to function minimization*, Maths. Comp. 21 (1967), pp. 368–381.

[12] C. Broyden, *On the discovery of the good broyden method*, Mathematical programming 87 (2000), pp. 209–213.

[13] A.R. Conn, N.I. Gould, and P.L. Toint, *Convergence of quasi-Newton matrices generated by the symmetric rank one update*, Mathematical Programming 50 (1991), pp. 177–195.

[14] J.E. Dennis and J.J. Moré, *Quasi-Newton Methods, Motivation and Theory*, SIAM Review 19 (1977), pp. 46–89.

[15] P. Deuflhard, *Newton methods for nonlinear problems: affine invariance and adaptive algorithms*, Vol. 35, Springer, 2011.

[16] M. Diehl, H.G. Bock, and J.P. Schlöder, *A real-time iteration scheme for nonlinear optimization in optimal feedback control*, SIAM Journal on Control and Optimization 43 (2005), pp. 1714–1736.

[17] M. Diehl, H.J. Ferreau, and N. Haverbeke, *Efficient numerical methods for nonlinear MPC and moving horizon estimation*, in *Nonlinear model predictive control*, L. Magni, M. Raimondo, and F. Allgöwer, eds., Lecture Notes in Control and Information Sciences Vol. 384, Springer, 2009, pp. 391–417.

[18] M. Diehl, R. Findeisen, F. Allgöwer, H.G. Bock, and J.P. Schlöder, *Nominal stability of the real-time iteration scheme for nonlinear model predictive control*, IEE Proc.-Control Theory Appl. 152 (2005), pp. 296–308.

[19] M. Diehl, *Lecture Notes on Numerical Optimization*, 2016, (Available online: http://cdn.syscop.de/publications/Diehl2016.pdf).

[20] M. Diehl, A. Walther, H.G. Bock, and E. Kostina, *An adjoint-based SQP algorithm with quasi-Newton Jacobian updates for inequality constrained optimization*, Optimization Methods and Software 25 (2010), pp. 531–552.

[21] H.J. Ferreau, S. Almer, R. Verschueren, M. Diehl, D. Frick, A. Domahidi, J.L. Jerez, G. Stathopoulos, and C. Jones, *Embedded Optimization Methods for Industrial Automatic Control*, 2017.

[22] A. Griewank, *Evaluating Derivatives, Principles and Techniques of Algorithmic Differentiation*, no. 19 in Frontiers in Appl. Math., SIAM, Philadelphia, 2000.

[23] A. Griewank and P.L. Toint, *Local convergence analysis for partitioned quasi-Newton updates*, Numerische Mathematik 39 (1982), pp. 429–448.

[24] A. Griewank and P. Toint, *Partitioned variable metric updates for large structured optimization problems*, Numerische Mathematik 39 (1982), pp. 119–137.

[25] A. Griewank and A. Walther, *On constrained optimization by adjoint based quasi-Newton methods*, Optimization Methods and Software 17 (2002), pp. 869–889.

[26] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II – Stiff and Differential-Algebraic Problems*, 2nd ed., Springer, Berlin Heidelberg, 1991.

[27] P. Hespanhol and R. Quirynen, *Quasi-Newton Jacobian and Hessian Updates for Pseudospectral based NMPC*, in *IFAC Conference on Nonlinear Model Predictive Control*. 2018.

[28] P. Hespanhol and R. Quirynen, *A Real-Time Iteration Scheme with Quasi-Newton Jacobian Updates for Nonlinear Model Predictive Control*, in *Proceedings of the European Control Conference (ECC)*. 2018.

[29] D. Janka, C. Kirches, S. Sager, and A. Wächter, *An SR1/BFGS SQP algorithm for non-convex nonlinear programs with block-diagonal Hessian matrix*, Mathematical Programming Computation 8 (2016), pp. 435–459.

[30] D. Mayne and J. Rawlings, *Model Predictive Control*, Nob Hill, 2013.

[31] J. Nocedal and S.J. Wright, *Numerical Optimization*, 2nd ed., Springer Series in Operations Research and Financial Engineering, Springer, 2006.

[32] A. Potschka, *A direct method for the numerical solution of optimization problems with time-periodic PDE constraints*, Ph.D. diss., University of Heidelberg, 2011.

[33] R. Quirynen, K. Berntorp, and S. Di Cairano, *Embedded Optimization Algorithms for Steering in Autonomous Vehicles based on Nonlinear Model Predictive Control*, in *Proceedings of the American Control Conference (ACC)*. 2018.

[34] R. Quirynen, S. Gros, and M. Diehl, *Inexact Newton-type optimization with iterated sensitivities*, SIAM Journal on Optimization 28 (2018), pp. 74–95.

[35] R. Quirynen, A. Knyazev, and S. Di Cairano, *Block Structured Preconditioning within an Active-Set Method for Real-Time Optimal Control*, in *Proceedings of the European Control Conference (ECC)*. 2018.

[36] R. Quirynen, *Numerical simulation methods for embedded optimization*, Ph.D. diss., KU Leuven and University of Freiburg, 2017.

[37] R. Quirynen, S. Gros, B. Houska, and M. Diehl, *Lifted collocation integrators for direct optimal control in ACADO toolkit*, Mathematical Programming Computation 9 (2017), pp. 527–571.

[38] A. Wächter and L.T. Biegler, *On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming*, Mathematical Programming 106 (2006), pp. 25–57.

[39] L. Wirsching, H.G. Bock, and M. Diehl, *Fast NMPC of a chain of masses connected by springs*, in *Proceedings of the IEEE International Conference on Control Applications, Munich*. 2006, pp. 591–596.