An Efficient Multiple Shooting Based Reduced SQP Strategy for Large-Scale Dynamic Process Optimization

Part I: Theoretical Aspects

Daniel B. Leineweber

Process Technology, Bayer AG, D-51368 Leverkusen, Germany

Irene Bauer** Hans Georg Bock* Johannes P. Schlöder

Interdisciplinary Center for Scientific Computing (IWR)
University of Heidelberg, D-69120 Heidelberg, Germany

Abstract

Optimal design and operation of complex chemical processes often require the solution of intricate dynamic optimization problems. A tailored simultaneous solution strategy based on multiple shooting and reduced SQP is presented. This *reduced-space boundary value problem (BVP) approach* allows an efficient and robust solution of multistage optimal control and design optimization problems for large, sparse DAE process models of index one. The current paper describes the theoretical aspects of the method. Utilizing the natural decomposition of the states into differential and algebraic variables, the structured NLP problem which results from the multiple shooting discretization of the optimization BVP is projected onto the reduced space of differential variables and control parameters. It is shown that this projection can be obtained very efficiently through direct computation of the reduced linearized constraint system via directional sensitivities. Like the original full-space BVP approach, the reduced-space formulation lends itself well to parallel computation. An implementation of the new strategy is provided by the modular optimal control package MUSCOD-II. Software aspects and applications are discussed in a second paper (Leineweber *et al.*, 2002 [10]).

Keywords

Large-Scale Optimal Control, Index One DAEs, Multiple Shooting, Structured Reduced SQP Methods, Directional Derivatives, Sparse Equation Systems, Parallel Computation.

1 Introduction

Today most practical optimal control problems are solved with NLP methods (or *direct* methods) based on some discretization of the original continuous problem. Sequential strategies just parametrize the control functions and employ numerical integration to discretize the dynamic model in a black box manner (e.g., Sargent and Sullivan, 1978 [17]; Vassiliadis *et al.*, 1994 [19]). Simultaneous strategies, on the other hand, explicitly include a suitable discretization of the dynamic model equations in their NLP formulation (e.g., Biegler, 1984 [3]), i.e., simulation and optimization may proceed simultaneously if an infeasible path optimization algorithm is employed. However, the great potential of simultaneous strategies in terms of both efficiency and robustness seems hard to realize in practice due to the very large size of the resulting NLP problems and the difficulty of adaptively controlling the discretization error, especially if *collocation* is applied (Logsdon and Biegler, 1989 [12], 1992 [13]).

In this context, a promising alternative way of discretizing the dynamic model is provided by *multiple shooting* (Bock and Plitt, 1984 [6]; Bock *et al.*, 1988 [7]; Leineweber, 1999 [9]). Unlike the more common collocation based strategies, direct use is made of existing advanced, fully adaptive DAE solvers. Therefore, a *fixed* discretization can be employed at the optimization level by restricting adaptivity to the error-controlled evaluation of the NLP functions and gradients. The NLP problems to be solved are typically much smaller than in case of collocation. A further advantage lies in the fact that the integrations on different multiple shooting intervals are completely decoupled and can therefore be performed in parallel.

**current address: Process Systems Enterprise Ltd., London W6 9DA, United Kingdom
*corresponding author (Email: `bock@iwr.uni-heidelberg.de`)

In order to exploit the structure of large, sparse DAE process models with many algebraic variables, a new partially reduced SQP strategy has been developed (Leineweber, 1999 [9]). The structured NLP problem resulting from the multiple shooting DAE discretization is projected onto the reduced space of differential variables plus control parameters, using the fact that the algebraic variables are uniquely determined by the consistency conditions due to the index one assumption. It is demonstrated that the expensive calculation of the full set of differential state sensitivities can be avoided – instead, it suffices to compute a reduced number of *directional* sensitivities if the projection and gradient evaluation steps are suitably intertwined.

A comprehensive implementation of this *reduced-space BVP approach* is provided by the modular optimal control package MUSCOD-II (Leineweber, 1999 [9]; Leineweber *et al.*, 2002 [10]). The BDF code DAESOL (Bauer *et al.*, 1997 [1]; Bauer, 2000 [2]) is used for the efficient calculation of the required directional derivatives.

2 Problem Formulation

2.1 A Multistage Optimal Control Problem in DAEs

Many dynamic process optimization problems in chemical engineering can be expressed as multistage optimal control problems in DAEs. Throughout this paper, we will consider the following general class of M-stage optimal control problems, where the time horizon of interest $[t_0, t_M]$ is divided into M model stages corresponding to the subintervals $[t_i, t_{i+1}]$, $i = 0, 1, \ldots, M-1$:

$$\min_{x_i,\, z_i,\, u_i,\, p,\, t_i} \sum_{i=0}^{M-1} \phi_i(x_i(t_{i+1}), z_i(t_{i+1}), p, t_{i+1}) \tag{1a}$$

subject to the DAE model stages

$$\left.\begin{array}{rcl} B_i(\,\cdot\,)\,\dot{x}_i(t) & = & f_i(x_i(t), z_i(t), u_i(t), p, t) \\ 0 & = & g_i(x_i(t), z_i(t), u_i(t), p, t) \end{array}\right\}, \quad t \in [t_i, t_{i+1}], \quad i = 0, 1, \ldots, M-1, \tag{1b}$$

the control and path constraints

$$h_i(x_i(t), z_i(t), u_i(t), p, t) \;\geq\; 0, \quad t \in [t_i, t_{i+1}], \quad i = 0, 1, \ldots, M-1, \tag{1c}$$

the stage transition conditions

$$x_{i+1}(t_{i+1}) \;=\; c_i(x_i(t_{i+1}), z_i(t_{i+1}), p, t_{i+1}), \quad i = 0, 1, \ldots, M-2, \tag{1d}$$

and the multipoint boundary conditions

$$\sum_{i=0}^{M-1} \left(r_i^s(x_i(t_i), z_i(t_i), p, t_i) \;+\; r_i^e(x_i(t_{i+1}), z_i(t_{i+1}), p, t_{i+1}) \right) \left\{ \begin{array}{c} = \\ \geq \end{array} \right\} 0. \tag{1e}$$

The performance index (1a) of generalized Mayer type is minimized with respect to the differential and algebraic state profiles x_i and z_i, respectively, the control profiles u_i, a time-independent global parameter vector p, and the model stage grid points t_i. The vectors $x_i(t)$, $z_i(t)$, $u_i(t)$, and p are of dimensions n_i^x, n_i^z, n_i^u, and n^p (typically, n_i^z is larger than n_i^x, and n_i^x is larger than n_i^u, n^p). Of course, the dynamic model equations as well as the relevant dimensions may change from one model stage to the next. For the quasilinear-implicit nonlinear DAEs (1b) we assume *(differential) index one*, i.e., B_i and $\partial g_i/\partial z_i$ must be nonsingular. In most practical cases, B_i as well as the Jacobians of f_i and g_i are quite sparse.

The possibility to directly consider sequences of *dynamic model stages* as shown in (1) is valuable particularly for the optimization of coupled (or integrated) batch processes, e.g., a batch reactor connected to a batch distillation column. Observe that additional "recycle loops" can be easily specified for a given linear sequence of coupled batch processes by means of the multipoint boundary conditions (1e) and the stage transition conditions (1d). This enables the optimization of cyclic batch operations, where material and energy can be transferred from one batch to the next, e.g., a batch distillation process with recycling of slop cuts.

2.2 Problem Discretization

The original continuous problem (1) is reformulated as an NLP problem with a finite number of optimization variables. On each model stage $i = 0, 1, \ldots, M-1$, we employ the time transformation

$$\theta_i(\tau, v) \;:=\; t_i + \tau d_i, \quad t_i = t_0 + \sum_{k=0}^{i-1} d_k, \quad \tau \in [0,1] \tag{2}$$

with $v := (t_0, d_0, d_1, \ldots, d_{M-1})$, and we choose a fixed, dimensionless discretization grid

$$0 \;=\; \tau_{i,0} \;<\; \tau_{i,1} \;<\; \ldots \;<\; \tau_{i,m_i} = \;1. \tag{3}$$

Then we define a piecewise approximation $\hat{u}_i$ of the control vector u_i by

$$\hat{u}_i(\tau) \;:=\; \varphi_{ij}(\tau, q_{ij}), \quad \tau \in I_{ij} = [\tau_{ij}, \tau_{i,j+1}], \quad j = 0, 1, \ldots, m_i - 1, \tag{4}$$

using "local" control parameters q_{ij}. The functions φ_{ij} are given basis functions, typically vectors of polynomials (e.g., constant, linear, or cubic).

The DAEs (1b) are discretized on the same grid (3) by a specific variant of *multiple shooting* which has been proposed by Bock *et al.* (1988 [7]). We introduce additional optimization parameters s^x_{ij}, s^z_{ij} and solve the following set of relaxed decoupled initial value problems (IVPs),

$$\left.\begin{aligned} B_i(\,\cdot\,)\, dx_i(\tau)/d\tau \;&=\; f_i(x_i(\tau), z_i(\tau), \varphi_{ij}(\tau, q_{ij}), p, \theta_i(\tau, v))\, d_i \\ 0 \;&=\; g_i(x_i(\tau), z_i(\tau), \varphi_{ij}(\tau, q_{ij}), p, \theta_i(\tau, v)) \\ &\quad - \alpha_{ij}(\tau)\, g_i(s^x_{ij}, s^z_{ij}, \varphi_{ij}(\tau_{ij}, q_{ij}), p, \theta_i(\tau_{ij}, v)) \end{aligned}\right\}, \quad \tau \in I_{ij}, \tag{5}$$

using $x_i(\tau_{ij}) = s^x_{ij}$ and $z_i(\tau_{ij}) = s^z_{ij}$ as initial values. Note that *relaxed* versions of the original DAEs (1b) appear in (5) to allow inconsistent values of s^x_{ij} and s^z_{ij}. The damping factor α_{ij} is given by a scalar function which is nonincreasing and nonnegative on I_{ij} and satisfies $\alpha_{ij}(\tau_{ij}) = 1$, e.g.,

$$\alpha_{ij}(\tau) \;=\; \exp\left(-\bar{\alpha}\, \frac{\tau - \tau_{ij}}{\tau_{i,j+1} - \tau_{ij}}\right), \quad \bar{\alpha} \geq 0. \tag{6}$$

Let $x_i(\tau_{i,j+1}; s^x_{ij}, s^z_{ij}, q_{ij}, p, v)$ denote the differential state values at $\tau = \tau_{i,j+1}$ obtained by numerical integration of (5). Then, by including the continuity conditions

$$x_i(\tau_{i,j+1}; s^x_{ij}, s^z_{ij}, q_{ij}, p, v) - s^x_{i,j+1} \;=\; 0, \quad j = 0, 1, \ldots, m_i - 1 \tag{7}$$

and the consistency conditions

$$g_i(s^x_{ij}, s^z_{ij}, \varphi_{ij}(\tau_{ij}, q_{ij}), p, \theta_i(\tau_{ij}, v)) \;=\; 0, \quad j = 0, 1, \ldots, m_i \tag{8}$$

as equality constraints in the NLP problem, we ensure that the optimal solution will satisfy the original DAEs (1b) on each model stage $i = 0, 1, \ldots, M-1$. However, since the NLP problem is solved by an infeasible path algorithm, the intermediate iterates generated in the solution process are in general *infeasible* with respect to the above DAE discretization (7)–(8), i.e., optimization and simulation may proceed simultaneously.

Allowing inconsistent values of the discretized state variables at the multiple shooting nodes through the relaxed DAE formulation (5) is an important feature of our BVP approach, because quite frequently, discontinuities are introduced at the discretization grid points τ_{ij} (e.g., if a piecewise constant control approximation is used). A feasible path approach would normally require a consistent initialization of the DAE model after each discontinuity, and this has been found to become very expensive for larger process models, even in the context of dynamic simulation. Therefore, we deal with the consistency issue at the level of the optimization problem, and our infeasible path strategy requires consistency only for the final optimal solution. It is interesting to note that the use of the damping factor $\alpha_{ij}(\tau)$ within the relaxed IVPs (5) in general amounts to a homotopy-like strategy which forces the solution more

and more toward the "correct" manifold during integration. Our experience indicates that damping is indeed beneficial, i.e., $\bar{\alpha}$ in (6) should be nonzero, but there fortunately seems to be little dependence on the exact magnitude of $\bar{\alpha}$. For practical nonlinear DAE problems, $\bar{\alpha} = 5.0$ provides an adequate choice (Leineweber, 1999 [9]).

Finally, the continuous control and path constraints (1c) are discretized at the points τ_{ij} of the grid (3), leading to a set of constraints of the form

$$h_i(s^x_{ij}, s^z_{ij}, \varphi_{ij}(\tau_{ij}, q_{ij}), p, \theta_i(\tau_{ij}, v)) \;\geq\; 0, \quad j = 0, 1, \ldots, m_i \tag{9}$$

for each model stage $i = 0, 1, \ldots, M-1$. Note that the pointwise discretization (9) in general allows violations of the original constraints *between* the grid points τ_{ij}. In principle, this can be avoided by applying one of the well-known techniques to enforce strict feasibility, e.g., the transformation of each nonlinear path constraint to an equivalent terminal constraint for an artificial state variable (Sargent and Sullivan, 1978 [17]). Such a strategy can be easily realized within the general framework of formulation (1). However, since techniques of this kind may give rise to numerical difficulties (introduction of non-differentiabilities, zero constraint gradients at the solution), we do not generally recommend their use in the context of our approach – we rather suggest refining the grid (3) if required.

2.3 Structured NLP Problem

The reformulation of the continuous M-stage optimal control problem (1) leads to the following structured NLP problem:

$$\min_{s^x_{ij},\, s^z_{ij},\, q_{ij},\, p,\, v} \;\; \sum_{i=0}^{M-1} \phi_i(s^x_{i,m_i},\, s^z_{i,m_i},\, p,\, \theta_i(\tau_{i,m_i}, v)) \tag{10a}$$

subject to the continuity and consistency conditions

$$\left.\begin{array}{rcll} x_i(\tau_{i,j+1}; s^x_{ij}, s^z_{ij}, q_{ij}, p, v) - s^x_{i,j+1} & = & 0, & j = 0, 1, \ldots, m_i - 1 \\ g_i(s^x_{ij}, s^z_{ij}, \varphi_{ij}(\tau_{ij}, q_{ij}), p, \theta_i(\tau_{ij}, v)) & = & 0, & j = 0, 1, \ldots, m_i \end{array}\right\}, \quad i = 0, 1, \ldots, M-1, \tag{10b}$$

the discretized control and path constraints

$$h_i(s^x_{ij}, s^z_{ij}, \varphi_{ij}(\tau_{ij}, q_{ij}), p, \theta_i(\tau_{ij}, v)) \;\geq\; 0, \quad j = 0, 1, \ldots, m_i, \quad i = 0, 1, \ldots, M-1, \tag{10c}$$

the stage transition conditions

$$c_i(s^x_{i,m_i}, s^z_{i,m_i}, p, \theta_i(\tau_{i,m_i}, v)) - s^x_{i+1,0} \;=\; 0, \quad i = 0, 1, \ldots, M-2, \tag{10d}$$

and the linearly coupled multipoint constraints

$$\sum_{i=0}^{M-1} \left(r^s_i(s^x_{i,0}, s^z_{i,0}, p, \theta_i(\tau_{i,0}, v)) \;+\; r^e_i(s^x_{i,m_i}, s^z_{i,m_i}, p, \theta_i(\tau_{i,m_i}, v)) \right) \left\{ \begin{array}{c} = \\ \geq \end{array} \right\} 0. \tag{10e}$$

A further slight modification of the discretized problem (10) is favorable in practice: the global model parameters (p, v) are "localized" by introducing new NLP variables $(p, v)_{ij}$ together with additional linear constraints enforcing

$$(p, v)_{ij} - (p, v)_{0,0} \;=\; 0 \tag{11}$$

for all $(i, j) \neq (0, 0)$. Then only *linear coupling* occurs between variables corresponding to different discretization intervals (some of the problem functions introduce no coupling at all), i.e., all problem functions are second-order decoupled.

For typical chemical process control applications, the structured NLP problem (10) may have several thousand variables due to the state discretization, but only relatively few degrees of freedom – say, up to a few hundred – according to the number of control parameters and global model parameters. Therefore, the performance of solution strategies very much depends on an adequate treatment of the large, structured system of nonlinear equality constraints (10b).

3 A Natural Decomposition for DAE Optimization

3.1 General Reduced SQP Framework

Large-scale NLP problems like (10) having many equality constraints can be efficiently solved using reduced SQP (RSQP) methods (e.g., Locke and Westerberg, 1983 [11]; Biegler *et al.*, 1995 [4]; Cervantes *et al.*, 2000 [8]). A generalization of the classical RSQP concept is provided by *partially* reduced SQP (PRSQP) methods, where only a *subset* of the equalities is used in defining the range and null space decomposition, and the reduced linearizations of the inequalities and the remaining equalities are kept in the QP subproblem. Thus within a coordinate bases approach, we may consider a generic partitioned NLP problem of the type

$$\min_{w_1,w_2} F(w_1,w_2) \quad \text{subject to} \quad \left\{ \begin{array}{rcl} G_1(w_1,w_2) & = & 0 \\ G_2(w_1,w_2) & = & 0 \\ H(w_1,w_2) & \geq & 0, \end{array} \right. \tag{12}$$

where $w_1 \in \mathbb{R}^{m_1}$ and $w_2 \in \mathbb{R}^{n-m_1}$ denote the basic (or dependent) variables and the nonbasic (or free) variables, respectively. (For simplicity of presentation, we have not included simple bounds here.) Of course, the m_1 basic variables w_1 and the corresponding m_1 constraint functions G_1 are chosen such that $\nabla_{w_1} G_1(w_1,w_2)^T$ is nonsingular. We then obtain the coordinate bases

$$S_k^{\mathcal{N}} \;=\; \begin{pmatrix} -(\nabla_{w_1} G_{1,k}^T)^{-1}\, \nabla_{w_2} G_{1,k}^T \\ I \end{pmatrix} \in \mathbb{R}^{n\times(n-m_1)}, \quad S_k^{\mathcal{R}} \;=\; \begin{pmatrix} I \\ 0 \end{pmatrix} \in \mathbb{R}^{n\times m_1}. \tag{13}$$

For this particular choice of bases, the range and null space decomposition of the correction step becomes

$$\begin{pmatrix} \Delta w_{1,k} \\ \Delta w_{2,k} \end{pmatrix} \;=\; S_k^{\mathcal{N}}\, y_k^{\mathcal{N}} + S_k^{\mathcal{R}}\, y_k^{\mathcal{R}} \;=\; \begin{pmatrix} D_k\, y_k^{\mathcal{N}} + y_k^{\mathcal{R}} \\ y_k^{\mathcal{N}} \end{pmatrix}, \tag{14}$$

where we have introduced the abbreviation

$$D_k \;:=\; -(\nabla_{w_1} G_{1,k}^T)^{-1}\, \nabla_{w_2} G_{1,k}^T \;\in\; \mathbb{R}^{m_1\times(n-m_1)}.$$

By applying the change of variables (14) to the full-space QP subproblem which corresponds to the partitioned NLP problem (12), it directly follows that the range space component $y_k^{\mathcal{R}}$ is given by

$$y_k^{\mathcal{R}} \;=\; -(\nabla_{w_1} G_{1,k}^T)^{-1}\, G_{1,k}, \tag{15}$$

and that the null space component $y_k^{\mathcal{N}} = \Delta w_{2,k}$ solves the QP subproblem

$$\begin{array}{l} \min\limits_{y^{\mathcal{N}}} \;\; (\nabla_{w_2} F_k + D_k^T\, \nabla_{w_1} F_k)^T\, y^{\mathcal{N}} + \dfrac{1}{2}\, y^{\mathcal{N}\,T} B_k^{\mathcal{N}}\, y^{\mathcal{N}} \\ \text{subject to} \quad \left\{ \begin{array}{rcl} G_{2,k} + \nabla_{w_1} G_{2,k}^T\, y_k^{\mathcal{R}} + (\nabla_{w_2} G_{2,k}^T + \nabla_{w_1} G_{2,k}^T\, D_k)\, y^{\mathcal{N}} & = & 0 \\ H_k + \nabla_{w_1} H_k^T\, y_k^{\mathcal{R}} + (\nabla_{w_2} H_k^T + \nabla_{w_1} H_k^T\, D_k)\, y^{\mathcal{N}} & \geq & 0. \end{array} \right. \end{array} \tag{16}$$

In the quadratic objective of (16), we have neglected a cross term which involves both $y_k^{\mathcal{R}}$ and $y_k^{\mathcal{N}}$, and $B_k^{\mathcal{N}}$ denotes a variable metric approximation to the (partially) reduced Hessian

$$S_k^{\mathcal{N}\,T}\, \nabla_w^2 L_k\, S_k^{\mathcal{N}} \;=\; D_k^T\, \nabla_{w_1}^2 L_k\, D_k + 2\, \nabla_{w_1}(\nabla_{w_2} L_k)\, D_k + \nabla_{w_2}^2 L_k. \tag{17}$$

RSQP methods of this type have been shown to be two-step q-superlinearly convergent if the reduced Hessian approximations $B_k^{\mathcal{N}}$ are generated by an asymptotically correct update scheme (e.g., Nocedal and Overton, 1985 [14]). There also exist one-step superlinearly convergent RSQP variants which either ensure that the omitted cross term actually becomes small (at the cost of one additional function or gradient evaluation at an intermediate point) or explicitly include some approximation of this term (e.g., Biegler *et al.*, 1995 [4]).

Consistent first-order estimates $\tilde{\lambda}_{1,k}$ of the Lagrange multipliers corresponding to the constraints G_1 can be obtained from

$$\tilde{\lambda}_{1,k} \;=\; (\nabla_{w_1} G_{1,k})^{-1} \left(\nabla_{w_1} F_k - \nabla_{w_1} G_{2,k}\, \tilde{\lambda}_{2,k} - \nabla_{w_1} H_k\, \tilde{\mu}_k \right), \tag{18}$$

where $\tilde{\lambda}_{2,k}$ and $\tilde{\mu}_k$ denote the multipliers of the reduced QP subproblem (16). The latter formula is motivated by the fact that $\nabla_w L(w^*, \lambda^*, \mu^*) = 0$.

There are various possibilities to approximate the reduced Hessian $S_k^{\mathcal{N}\,T} \nabla_w^2 L_k\, S_k^{\mathcal{N}}$. In our context, we will make use of the *reduced Lagrangian gradient*

$$\nabla_{y^{\mathcal{N}}} L(w_k + S_k^{\mathcal{N}} y^{\mathcal{N}}, \lambda, \mu)\big|_{y^{\mathcal{N}}=0} \;=\; S_k^{\mathcal{N}\,T}\, \nabla_w L(w_k, \lambda, \mu) \tag{19}$$

to directly calculate a variable metric approximation $B_k^{\mathcal{N}}$ of the reduced Hessian.

Starting from an initial guess $(w_{1,0}, w_{2,0})$, the PRSQP method for solving (12) calculates the sequence of iterates

$$\begin{pmatrix} w_{1,k+1} \\ w_{2,k+1} \end{pmatrix} \;=\; \begin{pmatrix} w_{1,k} \\ w_{2,k} \end{pmatrix} + \alpha_k \begin{pmatrix} \Delta w_{1,k} \\ \Delta w_{2,k} \end{pmatrix}, \quad k = 0, 1, \ldots, \tag{20}$$

where the relaxation factor $\alpha_k \in]0, 1]$ is determined by a line search to force global convergence.

The PRSQP approach allows a very flexible tailoring of the decomposition to large, but highly structured optimization problems like (10). While the low-level sparsity structure which may be present in the constraint Jacobian can be directly exploited through the use of coordinate bases, it is possible to conserve certain other structural features of the optimization problem (e.g., the partial separability of the Lagrangian function) in the reduction process. Hence this additional high-level structure can be taken advantage of when formulating and solving the reduced QP subproblem.

3.2 A Partitioned Coordinate Bases Approach

The index one assumption for the DAE model (1b) implies that the algebraic variables s_{ij}^z are completely determined through the consistency conditions for given values of the differential variables s_{ij}^x and parameters $\hat{q}_{ij} = (q_{ij}, p_{ij}, v_{ij})$. This observation provides the basis for using s_{ij}^z as dependent variables in a "natural" decomposition of the discretized dynamic optimization problem. Before outlining our decomposition approach, a few additional points should be made. We first rewrite the continuity and consistency conditions (10b) in the more compact form

$$x_i(\tau_{i,j+1}; s_{ij}^x, s_{ij}^z, \hat{q}_{ij}) - s_{i,j+1}^x \;=\; 0 \tag{21a}$$

$$\hat{g}_i(s_{ij}^x, s_{ij}^z, \hat{q}_{ij}, \tau_{ij}) \;=\; 0. \tag{21b}$$

These conditions lead to a characteristic "multiple shooting structure" in the constraint Jacobian of the structured NLP problem (10),

$$\begin{pmatrix} \ddots & & & & & & & \\ & X_{ij}^x & X_{ij}^z & X_{ij}^{\hat{q}} & -I & & & \\ & \hat{G}_{ij}^x & \hat{G}_{ij}^z & \hat{G}_{ij}^{\hat{q}} & & & & \\ & & & & X_{i,j+1}^x & X_{i,j+1}^z & X_{i,j+1}^{\hat{q}} & -I \\ & & & & \hat{G}_{i,j+1}^x & \hat{G}_{i,j+1}^z & \hat{G}_{i,j+1}^{\hat{q}} & \\ & & & & & & & \ddots \end{pmatrix}, \tag{22}$$

shown here only for two successive interior discretization points j and $j+1$ on model stage i. (Stage transition conditions (10d) lead to exactly the same general structure although they do not involve state and sensitivity integrations.) At this point, it is important to realize that the blocks

$$X_{ij}^z \;:=\; \frac{\partial}{\partial s_{ij}^z} x_i(\tau_{i,j+1}; s_{ij}^x, s_{ij}^z, \hat{q}_{ij})$$

and also X_{ij}^x, $X_{ij}^{\hat{q}}$ are dense, and their calculation is quite expensive since it requires a state and sensitivity integration on subinterval $[\tau_{ij}, \tau_{i,j+1}]$. In contrast, the blocks

$$\hat{G}_{ij}^z := \frac{\partial}{\partial s_{ij}^z} \hat{g}_i(s_{ij}^x, s_{ij}^z, \hat{q}_{ij}, \tau_{ij})$$

as well as $\hat{G}_{ij}^x$, $\hat{G}_{ij}^{\hat{q}}$ are usually sparse and can be calculated cheaply. Since many DAE process models have a large share of algebraic variables, there is a great incentive to exploit the sparsity of the consistency condition Jacobians $\hat{G}_{ij}^x$, $\hat{G}_{ij}^z$, $\hat{G}_{ij}^{\hat{q}}$, and to avoid the expensive calculation of the full set of differential state sensitivities X_{ij}^x, X_{ij}^z, $X_{ij}^{\hat{q}}$. While the former is a standard requirement in large-scale decompositions, the latter is not usually encountered since it relates to very specific features of problem (10), namely, the occurrence of dense and "expensive" blocks in the constraint Jacobian (22).

Let us now give a formal statement of the partially reduced SQP decomposition which is employed by MUSCOD-II. Our aim is to eliminate all algebraic variables from the problem by projecting onto the tangent space of the consistency conditions, thus performing the optimization (and the approximation of the Hessian) only within this tangent space. Based on the ordering of variables

$$w = \begin{pmatrix} w_{0,0} \\ w_{0,1} \\ \vdots \\ w_{\hat{M}-1,\hat{m}} \end{pmatrix} \quad \text{with } w_{ij} := \begin{pmatrix} s_{ij}^x \\ s_{ij}^z \\ \hat{q}_{ij} \end{pmatrix},$$

(where $\hat{M}$ denotes the total number of stages, including nontrivial stage transitions) we choose the *partitioned* coordinate bases

$$S^{\mathcal{N}} = \begin{pmatrix} S_{0,0}^{\mathcal{N}} & & & \\ & S_{0,1}^{\mathcal{N}} & & \\ & & \ddots & \\ & & & S_{\hat{M}-1,\hat{m}}^{\mathcal{N}} \end{pmatrix}, \quad S^{\mathcal{R}} = \begin{pmatrix} S_{0,0}^{\mathcal{R}} & & & \\ & S_{0,1}^{\mathcal{R}} & & \\ & & \ddots & \\ & & & S_{\hat{M}-1,\hat{m}}^{\mathcal{R}} \end{pmatrix} \tag{23a}$$

with diagonal blocks

$$S_{ij}^{\mathcal{N}} = \begin{pmatrix} I & 0 \\ -(\hat{G}_{ij}^z)^{-1}\hat{G}_{ij}^x & -(\hat{G}_{ij}^z)^{-1}\hat{G}_{ij}^{\hat{q}} \\ 0 & I \end{pmatrix}, \quad S_{ij}^{\mathcal{R}} = \begin{pmatrix} 0 \\ I \\ 0 \end{pmatrix}. \tag{23b}$$

Observe that $S_{ij}^{\mathcal{N}}$ and $S_{ij}^{\mathcal{R}}$ may be interpreted as *local* bases which act only on the subset of variables belonging to discretization point (i, j). (These bases always exist since $\hat{G}_{ij}^z$ is nonsingular due to the index one assumption.) Upon substitution of

$$\Delta w = S^{\mathcal{N}} y^{\mathcal{N}} + S^{\mathcal{R}} y^{\mathcal{R}} \tag{24}$$

into the full-space QP subproblem which comes from the structured NLP problem (10), we find that the range space component $y^{\mathcal{R}}$ is directly given by

$$y^{\mathcal{R}} = \begin{pmatrix} y_{0,0}^{\mathcal{R}} \\ y_{0,1}^{\mathcal{R}} \\ \vdots \\ y_{\hat{M}-1,\hat{m}}^{\mathcal{R}} \end{pmatrix} \quad \text{with} \quad y_{ij}^{\mathcal{R}} = -(\hat{G}_{ij}^z)^{-1}\, \hat{g}_i(s_{ij}^x, s_{ij}^z, \hat{q}_{ij}, \tau_{ij}), \tag{25}$$

while the null space component $y^{\mathcal{N}}$ takes the form

$$y^{\mathcal{N}} = \begin{pmatrix} y_{0,0}^{\mathcal{N}} \\ y_{0,1}^{\mathcal{N}} \\ \vdots \\ y_{\hat{M}-1,\hat{m}}^{\mathcal{N}} \end{pmatrix}, \quad \text{where } y_{ij}^{\mathcal{N}} = \begin{pmatrix} \Delta s_{ij}^x \\ \Delta \hat{q}_{ij} \end{pmatrix}, \tag{26}$$

and is obtained as the solution of a reduced QP subproblem. This reduced QP problem – which still contains reduced linearizations of the inequalities and of all equalities except the consistency conditions – has the characteristic structure known from the ODE case, i.e., a *blockdiagonal Hessian* and a *block sparse constraint Jacobian*. Thus it is possible to employ the high-rank block updates of Bock and Plitt (1984 [6]) in a variable metric approximation of the partially reduced Hessian, and existing highly efficient block QP solvers can be used for determining $y^{\mathcal{N}}$ (Leineweber, 1999 [9]).

We briefly summarize the advantages of our specific choice of partitioned coordinate bases (23) for problem (10). These bases

- are guaranteed to exist for index one problems,
- provide a very useful reduction if there are many algebraic variables,
- can be cheaply calculated on the basis of sparse factorizations of $\hat{G}^z_{ij}$,
- allow to conserve the separability structure of the original problem, and
- are constructed from and act on *local* information only.

The final point implies that the whole process of generating the reduced QP subproblem (including function and gradient evaluation, basis construction, and projection) can be directly parallelized, see Leineweber *et al.* (2002 [10]).

4 Generation of the Reduced QP Subproblem

4.1 Intertwining of Constraint Linearization and Projection

We have argued that it should be avoided to calculate all differential state sensitivities X^x_{ij}, X^z_{ij}, $X^{\hat{q}}_{ij}$ when generating the reduced QP subproblem. In order to see how this can be accomplished, let us once again have a closer look at the linearizations of conditions (21),

$$X^x_{ij}\Delta s^x_{ij} + X^z_{ij}\Delta s^z_{ij} + X^{\hat{q}}_{ij}\Delta\hat{q}_{ij} - \Delta s^x_{i,j+1} \;=\; -(x_{ij} - s^x_{i,j+1}) \tag{27a}$$

$$\hat{G}^x_{ij}\Delta s^x_{ij} + \hat{G}^z_{ij}\Delta s^z_{ij} + \hat{G}^{\hat{q}}_{ij}\Delta\hat{q}_{ij} \;=\; -\hat{g}_{ij}, \tag{27b}$$

where we have used the shorthands $x_{ij} = x_i(\tau_{i,j+1}; s^x_{ij}, s^z_{ij}, \hat{q}_{ij})$ and $\hat{g}_{ij} = \hat{g}_i(s^x_{ij}, s^z_{ij}, \hat{q}_{ij}, \tau_{ij})$. Since $\hat{G}^z_{ij}$ is nonsingular, the linearized consistency conditions (27b) can be solved for Δs^z_{ij} in terms of the remaining solution components Δs^x_{ij} and $\Delta\hat{q}_{ij}$, yielding

$$\Delta s^z_{ij} \;=\; -(\hat{G}^z_{ij})^{-1}\left(\hat{G}^x_{ij}\Delta s^x_{ij} + \hat{G}^{\hat{q}}_{ij}\Delta\hat{q}_{ij} + \hat{g}_{ij}\right) \;=:\; D^x_{ij}\Delta s^x_{ij} + D^{\hat{q}}_{ij}\Delta\hat{q}_{ij} + d^g_{ij}. \tag{28}$$

Of course, D^x_{ij}, $D^{\hat{q}}_{ij}$, and d^g_{ij} are determined by solving the linear equation system

$$\hat{G}^z_{ij}\left(D^x_{ij} \;\middle|\; D^{\hat{q}}_{ij} \;\middle|\; d^g_{ij}\right) \;=\; -\left(\hat{G}^x_{ij} \;\middle|\; \hat{G}^{\hat{q}}_{ij} \;\middle|\; \hat{g}_{ij}\right) \tag{29}$$

based on a sparse factorization of $\hat{G}^z_{ij}$, i.e., without calculating $(\hat{G}^z_{ij})^{-1}$. In MUSCOD-II, we employ Harwell subroutine MA48, an advanced direct sparse solver (Reid and Duff, 1993 [16]), for the solution of system (29).

Substituting Δs^z_{ij} from (28) into (27a), we arrive at the *reduced* linearized continuity conditions

$$\left(X^x_{ij} + X^z_{ij}D^x_{ij}\right)\Delta s^x_{ij} + \left(X^{\hat{q}}_{ij} + X^z_{ij}D^{\hat{q}}_{ij}\right)\Delta\hat{q}_{ij} - \Delta s^x_{i,j+1} \;=\; -(x_{ij} - s^x_{i,j+1}) - X^z_{ij}d^g_{ij}, \tag{30}$$

to be included as equality constraints in the reduced QP subproblem. For an efficient computation of these reduced linearizations it is now crucial to observe that

$$(X^x_{ij} + X^z_{ij}D^x_{ij}) \;=:\; \mathcal{V}^x_{ij}, \quad (X^{\hat{q}}_{ij} + X^z_{ij}D^{\hat{q}}_{ij}) \;=:\; \mathcal{V}^{\hat{q}}_{ij}, \quad \text{and} \quad X^z_{ij}d^g_{ij} \;=:\; \mathcal{V}^g_{ij} \tag{31}$$

can be interpreted as *directional derivatives* of $x_i(\tau_{i,j+1}; s^x_{ij}, s^z_{ij}, \hat{q}_{ij})$. The corresponding directions in $(s^x_{ij}, s^z_{ij}, \hat{q}_{ij})$-space are given by the columns of

$$\mathcal{D}^x_{ij} := \begin{pmatrix} I \\ D^x_{ij} \\ 0 \end{pmatrix}, \quad \mathcal{D}^{\hat{q}}_{ij} := \begin{pmatrix} 0 \\ D^{\hat{q}}_{ij} \\ I \end{pmatrix}, \quad \text{and } \mathcal{D}^g_{ij} := \begin{pmatrix} 0 \\ d^g_{ij} \\ 0 \end{pmatrix}, \tag{32}$$

respectively. While it is certainly possible to calculate all $n^x_i + n^z_i + n^{\hat{q}}_i$ differential state sensitivities X^x_{ij}, X^z_{ij}, $X^{\hat{q}}_{ij}$ and combine them according to (31), a direct computation of the required $n^x_i + n^{\hat{q}}_i + 1$ directional sensitivities $\mathcal{V}^x_{ij}$, $\mathcal{V}^{\hat{q}}_{ij}$, $\mathcal{V}^g_{ij}$ is much more efficient if n^z_i is large. Observe that the latter strategy necessitates *intertwining* the constraint linearization and projection steps to some degree (i.e., performing for each discretization point (i, j) first the linearization of consistency conditions, then the solution of linear system (29), and finally the directional linearization of continuity conditions). However, the directional sensitivity calculation can still be parallelized with respect to the discretization points (i, j) since $\mathcal{V}^x_{ij}$, $\mathcal{V}^{\hat{q}}_{ij}$, $\mathcal{V}^g_{ij}$ depend on strictly local information only.

A comparison of (23) and (25) with (32) shows that the local bases $S^{\mathcal{N}}_{ij}$ and range space moves $S^{\mathcal{R}}_{ij}\, y^{\mathcal{R}}_{ij}$ of our PRSQP decomposition can simply be interpreted as

$$S^{\mathcal{N}}_{ij} = \begin{pmatrix} \mathcal{D}^x_{ij} & \mathcal{D}^{\hat{q}}_{ij} \end{pmatrix}, \quad S^{\mathcal{R}}_{ij}\, y^{\mathcal{R}}_{ij} = \mathcal{D}^g_{ij} \tag{33}$$

which provides the obvious connection with the general PRSQP framework presented before. A certain price must be paid, however, for *not* calculating the complete set of constraint gradients. In particular, we can no longer calculate first-order estimates of the Lagrange multipliers for the eliminated consistency conditions from a Lagrangian stationarity condition, compare (18), because this would require explicit knowledge of the differential state sensitivities X^z_{ij}. Consequently, it is not possible to employ update schemes which are based on the original Lagrangian gradient, and a partially multiplier-free globalization strategy must be used.

4.2 Efficient Calculation of Directional Sensitivities

We now provide some background on the efficient calculation of the directional sensitivities $\mathcal{V}^x_{ij}$, $\mathcal{V}^{\hat{q}}_{ij}$, $\mathcal{V}^g_{ij}$ as realized by the BDF integrator DAESOL (Bauer *et al.*, 1997 [1]). For the sake of lucidity, we restrict ourselves to the simplified DAE system

$$\left.\begin{aligned} dx_i/d\tau &= \hat{f}_i(x_i, z_i, \hat{q}_{ij}, \tau) \\ 0 &= \hat{g}_i(x_i, z_i, \hat{q}_{ij}, \tau) - \alpha_{ij}(\tau)\, \hat{g}_i(s^x_{ij}, s^z_{ij}, \hat{q}_{ij}, \tau_{ij}) \end{aligned}\right\}, \quad \tau \in [\tau_{ij}, \tau_{i,j+1}] \tag{34}$$

with initial values $x_i(\tau_{ij}) = s^x_{ij}$ and $z_i(\tau_{ij}) = s^z_{ij}$. (For discussion of the general case (5) which includes the left-hand side matrix $B_i(\,\cdot\,)$, see Bauer, 2000 [2]).

The differential state sensitivities X^x_{ij}, X^z_{ij}, $X^{\hat{q}}_{ij}$ can be obtained by integrating the standard variational DAE system

$$\begin{aligned} \frac{d}{d\tau}\left(X^x_{ij} \mid X^z_{ij} \mid X^{\hat{q}}_{ij} \right) &= \frac{\partial \hat{f}_i}{\partial x_i}\left(X^x_{ij} \mid X^z_{ij} \mid X^{\hat{q}}_{ij} \right) + \frac{\partial \hat{f}_i}{\partial z_i}\left(Z^x_{ij} \mid Z^z_{ij} \mid Z^{\hat{q}}_{ij} \right) + \frac{\partial \hat{f}_i}{\partial \hat{q}_{ij}}\left(0 \mid 0 \mid I \right) \\ 0 &= \frac{\partial \hat{g}_i}{\partial x_i}\left(X^x_{ij} \mid X^z_{ij} \mid X^{\hat{q}}_{ij} \right) + \frac{\partial \hat{g}_i}{\partial z_i}\left(Z^x_{ij} \mid Z^z_{ij} \mid Z^{\hat{q}}_{ij} \right) + \frac{\partial \hat{g}_i}{\partial \hat{q}_{ij}}\left(0 \mid 0 \mid I \right) \\ &\quad - \alpha_{ij}(\tau)\left(\hat{G}^x_{ij} \mid \hat{G}^z_{ij} \mid \hat{G}^{\hat{q}}_{ij} \right) \end{aligned} \tag{35a}$$

from the initial values

$$\left(X^x_{ij} \mid X^z_{ij} \mid X^{\hat{q}}_{ij} \right)\Big|_{\tau=\tau_{ij}} = \left(I \mid 0 \mid 0 \right), \quad \left(Z^x_{ij} \mid Z^z_{ij} \mid Z^{\hat{q}}_{ij} \right)\Big|_{\tau=\tau_{ij}} = \left(0 \mid I \mid 0 \right) \tag{35b}$$

along with the original DAE (34). The algebraic state sensitivities Z^x_{ij}, Z^z_{ij}, $Z^{\hat{q}}_{ij}$ are not required on the optimization level, but must nevertheless be computed. Since the variational DAE (35a) is *linear* in the sensitivities, we can formally multiply from the right by the matrix

$$\mathcal{D}_{ij} \;:=\; \begin{pmatrix} \mathcal{D}^x_{ij} & \mathcal{D}^{\hat{q}}_{ij} & \mathcal{D}^g_{ij} \end{pmatrix} \;=\; \begin{pmatrix} I & 0 & 0 \\ D^x_{ij} & D^{\hat{q}}_{ij} & d^g_{ij} \\ 0 & I & 0 \end{pmatrix}. \tag{36}$$

Defining in addition the directional algebraic sensitivities

$$(Z^x_{ij} + Z^z_{ij} D^x_{ij}) \;=:\; \mathcal{W}^x_{ij}, \quad (Z^{\hat{q}}_{ij} + Z^z_{ij} D^{\hat{q}}_{ij}) \;=:\; \mathcal{W}^{\hat{q}}_{ij}, \quad \text{and} \;\; Z^z_{ij} d^g_{ij} \;=:\; \mathcal{W}^g_{ij} \tag{37}$$

we obtain the "directional" variational DAE

$$\begin{aligned} \frac{d}{d\tau}\left(\mathcal{V}^x_{ij} \mid \mathcal{V}^{\hat{q}}_{ij} \mid \mathcal{V}^g_{ij}\right) &= \frac{\partial \hat{f}_i}{\partial x_i}\left(\mathcal{V}^x_{ij} \mid \mathcal{V}^{\hat{q}}_{ij} \mid \mathcal{V}^g_{ij}\right) + \frac{\partial \hat{f}_i}{\partial z_i}\left(\mathcal{W}^x_{ij} \mid \mathcal{W}^{\hat{q}}_{ij} \mid \mathcal{W}^g_{ij}\right) + \frac{\partial \hat{f}_i}{\partial \hat{q}_{ij}}\left(0 \mid I \mid 0\right) \\ 0 &= \frac{\partial \hat{g}_i}{\partial x_i}\left(\mathcal{V}^x_{ij} \mid \mathcal{V}^{\hat{q}}_{ij} \mid \mathcal{V}^g_{ij}\right) + \frac{\partial \hat{g}_i}{\partial z_i}\left(\mathcal{W}^x_{ij} \mid \mathcal{W}^{\hat{q}}_{ij} \mid \mathcal{W}^g_{ij}\right) + \frac{\partial \hat{g}_i}{\partial \hat{q}_{ij}}\left(0 \mid I \mid 0\right) \\ &\quad - \alpha_{ij}(\tau)\left(0 \mid 0 \mid -\hat{g}_{ij}\right) \end{aligned} \tag{38a}$$

with initial values

$$\left(\mathcal{V}^x_{ij} \mid \mathcal{V}^{\hat{q}}_{ij} \mid \mathcal{V}^g_{ij}\right)\Big|_{\tau=\tau_{ij}} = \left(I \mid 0 \mid 0\right), \quad \left(\mathcal{W}^x_{ij} \mid \mathcal{W}^{\hat{q}}_{ij} \mid \mathcal{W}^g_{ij}\right)\Big|_{\tau=\tau_{ij}} = \left(D^x_{ij} \mid D^{\hat{q}}_{ij} \mid d^g_{ij}\right). \tag{38b}$$

This transformed variational DAE system for the direct calculation of $\mathcal{V}^x_{ij}$, $\mathcal{V}^{\hat{q}}_{ij}$, $\mathcal{V}^g_{ij}$ is the one which is actually solved together with the original DAE system (34). DAESOL uses the principle of *internal numerical differentiation (IND)*, see e.g., Bock and Plitt (1984 [6]) or Bauer (2000 [2]), which makes sure that the sensitivities are the exact derivatives of the *discretized* solution of the original DAE system. To our knowledge, DAESOL is currently the only BDF code capable of automatically generating and solving the directional system (38) given a matrix of directions (36). Due to the use of sparse linear solvers like MA48 (Reid and Duff, 1993 [16]), an efficient order and step size control, and a special monitoring strategy which saves evaluations and factorizations of the iteration matrix, DAESOL allows an efficient integration of large nonlinear DAE systems (Bauer *et al.*, 1997 [1]).

4.3 Approximation of the Reduced Hessian

Since the decomposition based on the partitioned coordinate bases (23) preserves the separability structure of the full-space problem – as mentioned, the partially reduced Hessian $S^{\mathcal{N}\,T}\,\nabla^2_w L\, S^{\mathcal{N}}$ is still block diagonal – we may use a suitable generalization of the partitioned variable metric update strategy of Bock and Plitt (1984 [6]) for the direct approximation of $S^{\mathcal{N}\,T}\,\nabla^2_w L\, S^{\mathcal{N}}$. Specifically, we apply the blockwise high-rank updates

$$(B^{\mathcal{N}}_{ij})_{k+1} \;=\; (B^{\mathcal{N}}_{ij})_k \;+\; U\big(\,(B^{\mathcal{N}}_{ij})_k,\, (\delta^{\mathcal{N}}_{ij})_k,\, (\gamma^{\mathcal{N}}_{ij})_k\,\big), \tag{39}$$

where $(B^{\mathcal{N}}_{ij})_k$ is an approximation for $(S^{\mathcal{N}\,T}_{ij}\,\nabla^2_{w_{ij}} L_{ij}\, S^{\mathcal{N}}_{ij})_k$. The update U is given by the heuristically modified BFGS formula of Powell (1978 [15]). Note that the exact partially reduced Hessian will often be indefinite, hence the heuristic modification of the update scheme is required to ensure positive definite Hessian approximations. In (39), we employ the step

$$(\delta^{\mathcal{N}}_{ij})_k \;:=\; \alpha_k\,(y^{\mathcal{N}}_{ij})_k, \tag{40}$$

and the corresponding difference of reduced Lagrangian gradients is defined by

$$(\gamma^{\mathcal{N}}_{ij})_k := (S^{\mathcal{N}}_{ij})^T_{k+1}\,\nabla_{w_{ij}} L_{ij}((w_{ij})_{k+1},\, \lambda_{k+1},\, \mu_{k+1},\, \cdot\,) - (S^{\mathcal{N}}_{ij})^T_k\,\nabla_{w_{ij}} L_{ij}((w_{ij})_k,\, \lambda_{k+1},\, \mu_{k+1},\, \cdot\,). \tag{41}$$

It should be noted that the reduced Lagrangian gradients which appear in (41) do *not* explicitly depend on the multipliers of the consistency conditions. Hence no difficulty arises in our method, where these multipliers are unknown. For the latter reason, most other possible definitions of $(\gamma^{\mathcal{N}}_{ij})_k$ (e.g., Nocedal and Overton, 1985 [14]) cannot be used.

5 A Partially Multiplier-Free Globalization Strategy

Finally we address the changes required to adapt the globalization strategy to our partially multiplier-free PRSQP approach. To keep it simple, we use the generic partitioned NLP problem (12) as the basis for our presentation. Since we do not calculate the multiplier estimates $\tilde{\lambda}_{1,k}$ corresponding to the constraints G_1, we can no longer employ the l_1 exact penalty function P_P of Powell (1978 [15]) with heuristic penalty weights derived from the multiplier estimates. Instead, we define the slightly modified penalty function

$$\tilde{P}_P(w_k, \rho, \sigma, \tau) \;=\; F_k + \rho \sum_{i=1}^{m_1} |G_{1,k,i}| + \sum_{i=1}^{m-m_1} \sigma_i \, |G_{2,k,i}| + \sum_{j=1}^{l} \tau_j \, |\min(0, H_{k,j})| \tag{42}$$

which contains a single scalar penalty factor ρ for all constraint residuals $G_{1,k,i}$. It is easy to show that the directional derivative $D_{\Delta w_k} \tilde{P}_{P,k}$ with $\Delta w_k = S_k^{\mathcal{N}} \, y_k^{\mathcal{N}} + S_k^{\mathcal{R}} \, y_k^{\mathcal{R}}$ can be expressed as

$$\begin{aligned} D_{\Delta w_k} \tilde{P}_P(w_k, \rho, \sigma, \tau) \;=\; & -y_k^{\mathcal{N}\,T} B_k^{\mathcal{N}} \, y_k^{\mathcal{N}} - \rho \, \|G_{1,k}\|_1 - \tilde{\lambda}_{1,k}^T G_{1,k} \\ & - \sum_{i \in \check{\mathcal{E}}_k} (\sigma_i - \tilde{\lambda}_{2,k,i}) \, (-G_{2,k,i}) - \sum_{i \in \hat{\mathcal{E}}_k} (\sigma_i + \tilde{\lambda}_{2,k,i}) \, G_{2,k,i} \\ & - \sum_{j \in \check{\mathcal{I}}_k} (\tau_j - \tilde{\mu}_{k,j}) \, (-H_{k,j}) - \sum_{j \in \hat{\mathcal{I}}_k} \tilde{\mu}_{k,j} \, H_{k,j}. \end{aligned} \tag{43}$$

Clearly, for positive definite Hessian approximations $B_k^{\mathcal{N}}$, compatibility of the search direction Δw_k with the merit function $\tilde{P}_P$ is obtained if

$$\rho \;\geq\; \|\tilde{\lambda}_{1,k}\|_\infty, \quad \sigma_i \;\geq\; |\tilde{\lambda}_{2,k,i}|, \quad \tau_j \;\geq\; \tilde{\mu}_{k,j} \tag{44}$$

for all k. Furthermore, if the method converges to a KKT point (w^*, λ^*, μ^*) which satisfies the Jacobian uniqueness condition, w^* must be a local minimizer of $\tilde{P}_P$ as well, since (44) then automatically implies that

$$\rho \;\geq\; \|\tilde{\lambda}_1^*\|_\infty, \quad \sigma_i \;\geq\; |\tilde{\lambda}_{2,i}^*|, \quad \tau_j \;\geq\; \tilde{\mu}_j^*. \tag{45}$$

However, a weight selection scheme on the basis of (44) is not applicable in the present case due to the unavailability of $\|\tilde{\lambda}_{1,k}\|_\infty$.

At this point it is important to note that the term $\tilde{\lambda}_{1,k}^T G_{1,k}$ in (43) can be calculated very cheaply in the form

$$\tilde{\lambda}_{1,k}^T G_{1,k} \;=\; -\nabla_{w_1} F_k^T \, y_k^{\mathcal{R}} + \tilde{\lambda}_{2,k}^T \, \nabla_{w_1} G_{2,k}^T \, y_k^{\mathcal{R}} + \tilde{\mu}_k^T \, \nabla_{w_1} H_k^T \, y_k^{\mathcal{R}}, \tag{46}$$

i.e., based on directional derivatives only. This immediately follows from the Lagrange multiplier formula (18). The gradient terms $\nabla_{w_1} G_{2,k}^T \, y_k^{\mathcal{R}}$ and $\nabla_{w_1} H_k^T \, y_k^{\mathcal{R}}$ must be computed anyway in order to formulate the reduced QP subproblem (16). If we now replace the condition $\rho \geq \|\tilde{\lambda}_{1,k}\|_\infty$ by the weaker requirement

$$\rho \, \|G_{1,k}\|_1 \;\geq\; |\tilde{\lambda}_{1,k}^T G_{1,k}|, \tag{47}$$

we still obtain compatibility of Δw_k and $\tilde{P}_P$, but unless additional precautions are taken, the resulting globalization strategy may fail to force convergence of $G_1 = 0$. In practice, however, the heuristic weight selection scheme

$$\rho_k \;=\; \begin{cases} \rho_{k-1} & \text{if } \rho_{k-1} \|G_{1,k}\|_1 \geq |\tilde{\lambda}_{1,k}^T G_{1,k}| + \hat{\rho} \, \|G_{1,k}\|_1 \\ \dfrac{|\tilde{\lambda}_{1,k}^T G_{1,k}|}{\|G_{1,k}\|_1} + 2\hat{\rho} & \text{otherwise} \end{cases} \tag{48}$$

(where $\hat{\rho}$ is some positive constant) can be safely used if in addition $\|G_{1,k}\|_1$ is monitored over a certain number t of subsequent iterations, and ρ_k is increased whenever

$$\|G_{1,k}\|_1 \;\geq\; \max\left(\|G_{1,k-1}\|_1, \, \|G_{1,k-2}\|_1, \, \ldots, \, \|G_{1,k-t}\|_1\right).$$

Observe that ρ_k is never decreased and eventually should become fixed. The other weights $\sigma_{k,i}$ and $\tau_{k,j}$ can be chosen according to a standard rule like the one of Powell (1978 [15]) since the corresponding multipliers $\tilde{\lambda}_{2,k,i}$ and $\tilde{\mu}_{k,j}$ are known. A similar multiplier-free scheme has been described by Biegler *et al.* (1997 [5]) for the classical RSQP case in order to completely avoid the calculation of multipliers.

The reduced-space variant of MUSCOD-II uses a watchdog line search globalization which is based on the above modified penalty function and penalty weights. Of course, the separability structure of problem (10) allows to calculate a separate penalty weight ρ_{ij} for each discretization point (i, j). Performance on practical problems is further improved by relaxing the weight selection scheme (48) such that ρ_k is allowed to decrease when both $|\tilde{\lambda}_{1,k}^T G_{1,k}|/\|G_{1,k}\|_1$ and $\|G_{1,k}\|_1$ decrease.

6 Summary and Conclusions

We have developed an efficient new decomposition approach for dynamic multistage optimization problems which involve large nonlinear DAE process models of index one. Based on multiple shooting and tailored partially reduced SQP, our strategy provides a natural generalization of the classical full-space BVP approach for ODEs (Bock and Plitt, 1984 [6]), considerably extending the original range of applicability. The new reduced-space BVP approach has the following central features:

- use of fully adaptive DAE solvers (allows to decouple the integration accuracy from the discretization used on the optimization level)
- relaxed DAE formulation with damping factor (allows inconsistent values of state variables at multiple shooting nodes, improves robustness by forcing solution toward consistency during integration)
- use of sparse linear solvers at the optimization level as well as within the integrator (allows to exploit sparsity of large DAE process models)
- direct computation of the reduced linearized constraint set via directional derivatives (allows to significantly decrease the computational work for the linearization of large problems having many algebraic variables)
- complete decoupling of state and sensitivity integrations on different multiple shooting intervals (allows parallel function and gradient evaluations)
- combination of the advantages of simultaneous and sequential strategies (explicit discretization of the dynamic model, but still relatively small NLP problem)
- solution of general multistage problems with control and path constraints and multipoint boundary conditions
- large degree of compatibility with existing dynamic process simulators (only requires a DAE solver which is capable of efficiently generating directional sensitivities)

Our approach is well suited for large-scale chemical process optimization as long as the underlying DAE models do not contain too many differential states. Since both the directional sensitivity matrices and the diagonal Hessian blocks are dense, our current PRSQP strategy is limited to DAE models having no more than a few hundred differential states. There is, of course, no such limitation regarding the number of algebraic states.

While many industrial DAE process models have only a moderate number of differential states, larger DAE models may result, e.g., from a spatial discretization of coupled partial differential and algebraic equations (PDAEs) by the method of lines. In these cases, a further reduction could be obtained by projecting away also the differential variables. As has been shown by Schlöder (1988 [18]), this is possible when the initial values are fixed. A corresponding completely reduced variant of our strategy is currently under development and will be described in a forthcoming publication.

References

[1] Bauer, I., F. Finocchi, W. J. Duschl, H.-P. Gail, and J. P. Schlöder, Simulation of chemical reactions and dust destruction in protoplanetary accretion discs. *Astron. Astrophys.* **317**, 273–289 (1997).

[2] Bauer, I., Numerische Verfahren zur Lösung von Anfangswertaufgaben und zur Generierung von ersten und zweiten Ableitungen mit Anwendungen bei Optimierungsverfahren in Chemie und Verfahrenstechnik. *Ph.D. thesis*, University of Heidelberg (2000).

[3] Biegler, L. T., Solution of dynamic optimization problems by successive quadratic programming and orthogonal collocation. *Comput. Chem. Engng* **8**, 243–248 (1984).

[4] Biegler, L. T., J. Nocedal, and C. Schmid, A reduced Hessian method for large-scale constrained optimization. *SIAM J. Optimization* **5**, 314–347 (1995).

[5] Biegler, L. T., C. Schmid, and D. Ternet, A multiplier-free, reduced Hessian method for process optimization. In L. T. Biegler, T. F. Coleman, A. R. Conn, and F. N. Santosa (eds.), *Large-Scale Optimization with Applications – Part II: Optimal Design and Control.* Springer (1997).

[6] Bock, H. G., and K. J. Plitt, A multiple shooting algorithm for direct solution of optimal control problems. *Proceedings of the 9th IFAC World Congress, Budapest.* Pergamon Press (1984).

[7] Bock, H. G., E. Eich, and J. P. Schlöder, Numerical solution of constrained least squares boundary value problems in differential-algebraic equations. In K. Strehmel (ed.), *Numerical Treatment of Differential Equations.* Teubner, Leipzig (1988).

[8] Cervantes, A. M., A. Wächter, R. Tutuncu, and L. T. Biegler, A reduced space interior point strategy for optimization of differential algebraic systems. *Comput. Chem. Engng* **24**, 39–51 (2000).

[9] Leineweber, D. B., Efficient Reduced SQP Methods for the Optimization of Chemical Processes Described by Large Sparse DAE Models. *Fortschritt-Berichte VDI*, Reihe 3, Nr. 613 (ISBN 3-18-361303-4). VDI Verlag GmbH, Düsseldorf (1999).

[10] Leineweber, D. B., A. Schäfer, H. G. Bock, and J. P. Schlöder, An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization – part II: software aspects and applications. (2002).

[11] Locke, M. H., A. W. Westerberg, and R. H. Edahl, Improved successive quadratic programming optimization algorithm for engineering design problems. *AIChE J.* **29**, 871–874 (1983).

[12] Logsdon, J. S., and L. T. Biegler, Accurate solution of differential-algebraic optimization problems. *Ind. Eng. Chem. Res.* **28**, 1628–1639 (1989).

[13] Logsdon, J. S., and L. T. Biegler, Decomposition strategies for large-scale dynamic optimization problems. *Chem. Engng Sci.* **47**, 851–864 (1992).

[14] Nocedal, J., and M. L. Overton, Projected Hessian updating algorithms for nonlinearly constrained optimization. *SIAM J. Numer. Anal.* **22**, 821–850 (1985).

[15] Powell, M. J. D., A fast algorithm for nonlinearly constrained optimization calculations. In G. A. Watson (ed.), *Numerical Analysis, Dundee 1977.* (Lecture Notes in Mathematics 630.) Springer, Berlin (1978).

[16] Reid, J., I. S. Duff: MA48, a Fortran code for direct solution of sparse unsymmetric linear systems of equations. *Report RAL–93–072*, Rutherford Appleton Laboratory (1993).

[17] Sargent, R. W. H., and G. R. Sullivan, The development of an efficient optimal control package. In J. Stoer (ed.), *Proceedings of the 8th IFIP Conference on Optimization Techniques (1977), Part 2.* Springer, Heidelberg (1978).

[18] Schlöder, J. P., Numerische Methoden zur Behandlung hochdimensionaler Aufgaben der Parameteridentifizierung. *Bonner Mathematische Schriften 187*, University of Bonn (1988).

[19] Vassiliadis, V. S., R. W. H. Sargent, and C. C. Pantelides, Solution of a class of multistage dynamic optimization problems. 1. Problems without path constraints. *Ind. Eng. Chem. Res.* **33**, 2111–2122 (1994).