

An Adjoint-based Numerical Method for Fast Nonlinear Model Predictive Control [★]

L. Wirsching* J. Albersmeyer* P. Kühl* M. Diehl**
H.G. Bock*

* *Interdisciplinary Center for Scientific Computing, University of
Heidelberg, Germany*

** *Electrical Engineering Department (ESAT), K.U. Leuven, Belgium*

Abstract: The application of optimization-based control methods such as nonlinear model predictive control (NMPC) to real-world process models is still a major computational challenge. In this paper, we present a new numerical optimization scheme suited for NMPC. The SQP-type approach uses an inexact constraint Jacobian in its iterations and is based on adjoint derivatives, that can be computed very efficiently. In comparison to a similar real-time algorithm based on directional sensitivities and an exact constraint Jacobian, the computational complexity is significantly reduced. Both algorithms are applied to the model of a thermally coupled distillation column for disturbance rejection. The results provide a proof-of-principle for the proposed adjoint-based optimization approach.

1. INTRODUCTION

Nonlinear model predictive control (NMPC) is an optimization-based control technique for multivariable processes that directly takes into account nonlinearities, bounds on inputs and states as well as state-dependant constraints. A mathematical process model is used to formulate an optimal control problem on a finite horizon with the current process state as the initial value. The optimal control problem describes the control task (setpoint control, trajectory tracking, economic objective) and incorporates all constraints. To establish closed-loop feedback, only a first part of the solution to the optimal control problem is applied to the process. Then, the problem is reformulated with a new initial value and solved again. The successive solution of a nonlinear constrained optimal control problem implicitly delivers a feedback law. A more detailed introduction can be found in [Allgöwer and Zheng, 2000], an overview of recent research topics is given in [Findeisen et al., 2006].

While *linear* model predictive control (based on linear process models) has become a standard in the petrochemical industry and is widely used in other process industries [Qin and Badgwell, 2003], too, there are still not many practical applications of NMPC. A few exceptions are described in, e.g., [Qin and Badgwell, 2003, Bartusiak, 2005].

This discrepancy seems due to two main reasons:

First, NMPC requires a mathematical process model based on first principles, typically formulated as differential-algebraic equations (DAEs) or partial differential equations (PDEs). Exact modeling, however, is often difficult, time-consuming and expensive. Modern modeling environments and tools bring relief to this bottleneck, see for

* The research was supported by the co-operative project cluster (DFG-Paketantrag) "Optimization-based control of processing plants". M. Diehl also acknowledges support by the Research Council KUL, CoE EF/05/006, and by the Belgian Federal Science Policy Office via IUAP P6/04.

example [Elmqvist et al., 1998, Braunschweig et al., 2000]. The alternative to use data-driven input-output models, e.g. based on artificial neural networks, is not considered here.

Second, NMPC requires the successive numerical solution of nonlinear constrained optimal control problems in real-time. For realistic processes, these optimization problems easily become large while the control action to suppress disturbances or follow setpoint changes must be available fast enough. Therefore, NMPC requires fast numerical methods specifically tailored for real-time control. An overview of optimization methods for NMPC can be found in [Biegler, 2000, Cannon, 2004].

An important conceptual step towards real-time feasible NMPC has been to leave the idea of "optimization until convergence" behind. In classic implementations, the optimal control problem is iteratively solved by a numerical algorithm until a specified convergence criterion is met. In this case one has to hope that convergence is achieved "fast enough". Instead, a new paradigm suggests to compute only as many iterations as the process time constants permit before proceeding to the next problem. This approach eventually led to the concept of real-time iterations as suggested by Diehl [2002]. Here, only one iteration per sampling instant is computed, but the successive optimal control problems are connected via a specific shift of the previous results. Furthermore, the current process state is embedded into the subsequent problems as an initially violated linear constraint. This results in a coupling of process dynamics and numerical optimization as demonstrated for a pilot-plant distillation column in [Diehl et al., 2002]. The real-time iterations have been formulated within the framework of direct multiple shooting [Bock and Plitt, 1984, Leineweber et al., 2003]. An even more efficient extension for processes with many states but only few controls was proposed by Schäfer et al. [2007].

In this article, the ideas presented in [Diehl, 2002, Bock et al., 2005] are further developed. We propose an optimization algorithm for NMPC problems based on adjoints and inexact constraint Jacobians that follows the principles of real-time iterations. The adjoint-based problem formulation allows to reduce the number of sensitivities that need to be computed to a single gradient evaluation. We also briefly describe how this gradient can efficiently be computed within the DAE solver. The computational effort of the algorithm is compared to existing real-time iteration algorithms and the applicability is demonstrated using the model of a thermally coupled distillation column as a process example.

2. PROBLEM FORMULATION

We first state the model class under consideration and then give a detailed formulation of the NMPC problem to be solved at each sampling instant.

2.1 Model class

We consider systems modeled by a set of differential-algebraic equations (DAE)

$$\dot{x}(t) = f(x(t), z(t), u(t)), \quad x(t_0) = x_0, \quad (1a)$$

$$0 = g(x(t), z(t), u(t)) \quad (1b)$$

with differential states $x \in \mathbb{R}^{n_x}$, algebraic states $z \in \mathbb{R}^{n_z}$, and controls $u \in \mathbb{R}^{n_u}$. It is assumed that the matrix $\frac{\partial g}{\partial z}$ is nonsingular, i.e. the DAE is of index 1. We note that a DAE system of higher index can often be transformed to this form by index reduction techniques. Furthermore, full state information is assumed while in practical applications only some of the states are usually measured and state estimation techniques may become an important issue.

2.2 Nonlinear model predictive control problem

In a standard NMPC formulation, the following optimal control problem has to be solved at each sampling instant t_k :

$$\min_{\substack{u(\cdot) \\ x(\cdot), z(\cdot)}} \int_{t_k}^{t_k+T} L(x(t), z(t), u(t)) dt \quad (2a)$$

subject to

$$\dot{x}(t) = f(x(t), z(t), u(t)), \quad (2b)$$

$$0 = g(x(t), z(t), u(t)), \quad (2c)$$

$$0 = x(t_k) - \hat{x}_k, \quad (2d)$$

$$0 \leq c(x(t), z(t), u(t)), \quad (2e)$$

The initial value \hat{x}_k represents the current process state at time t_k that we assume to be known in this article. The inequality constraints (2e) summarize simple bounds on controls and states as well as more complicated path constraints. The Lagrange term $L(\cdot)$ is often formulated as a least-squares objective penalizing the deviation from a given setpoint or reference trajectory. Optimizing formulations, where the cost functional rather reflects the true economic aim of the controller, are an interesting alternative currently under investigation [Engell, 2007].

One may also add an additional Mayer term $M(x(t_k + T), z(t_k + T))$ to the cost functional (2a). This has been identified as one possible measure to guarantee nominal stability of the NMPC scheme. For the stability theory of NMPC we refer to the excellent articles by Mayne et al. [2000] and more recently, by Magni and Scattolini [2004].

2.3 Transformation to a finite-dimensional NLP

We employ the direct multiple shooting approach [Bock and Plitt, 1984] – a simultaneous direct optimization method – to solve the optimal control problem (2): First, the controls are parameterized on a time grid $t_k = \tau_0 < \tau_1 < \dots < \tau_N = t_k + T$ (the grid points typically coincide with the sampling instants). Although any formulation with local support may be chosen, we restrict ourselves to piecewise constant controls, i.e. $u(t) = q_i, t \in [\tau_i, \tau_{i+1}]$. Second, the states are discretized at the grid points by introducing additional degrees of freedom $s_i = (s_i^x, s_i^z)$ as initial values of initial value problems (1) on time intervals $[\tau_i, \tau_{i+1}]$, $i = 0, 1, \dots, N-1$. For the discretized problem to remain consistent with (2), so-called continuity and consistency conditions are introduced as part of the nonlinear program (NLP). The finite dimensional NLP to be solved at each sampling instant then reads as

$$\min_{\substack{q_0, \dots, q_{N-1}, \\ s_0, \dots, s_N}} \sum_{i=0}^{N-1} L_i(s_i^x, s_i^z, q_i) \quad (3a)$$

subject to

$$s_{i+1}^x = x_i(\tau_{i+1}; s_i^x, s_i^z, q_i), \quad i = 0, 1, \dots, N-1, \quad (3b)$$

$$0 = g(s_i^x, s_i^z, q_i), \quad i = 0, 1, \dots, N, \quad (3c)$$

$$0 = s_0^x - \hat{x}_0, \quad (3d)$$

$$0 \leq c(s_i^x, s_i^z, q_i), \quad i = 0, 1, \dots, N. \quad (3e)$$

We can rewrite (3) in the general form

$$\min_w F(w) \quad (4a)$$

$$\text{s. t. } G(w) = 0 \quad (4b)$$

$$H(w) \geq 0. \quad (4c)$$

with $w = (s_0, \dots, s_N, q_0, \dots, q_{N-1})$. The Lagrangian function of this problem is defined as

$$\mathcal{L}(w, \lambda, \mu) = F(w) - \lambda^T G(w) - \mu^T H(w). \quad (5)$$

3. AN ADJOINT-BASED SOLUTION APPROACH

In this section we first review an existing scheme, called PRSQP algorithm, and then explain the adjoint-based approach. In both cases, we treat NLP (4) by a sequential quadratic programming (SQP) approach: Starting with an initial guess w_0 one computes the next iteration as $w_{k+1} = w_k + \Delta w_k$, where Δw_k solves the quadratic subproblem

$$\min_{\Delta w_k} \frac{1}{2} \Delta w_k^T A_k \Delta w_k + m_k^T \Delta w_k, \quad (6a)$$

$$\text{s. t. } B_k \Delta w_k + G(w_k) = 0, \quad (6b)$$

$$C_k \Delta w_k + H(w_k) \geq 0. \quad (6c)$$

A_k denotes the Hessian of the Lagrangian function (5) or an approximation thereof, while the choices of B_k , C_k , and m_k^T make the difference between the PRSQP and the adjoint-based algorithms.

3.1 Classic PRSQP algorithm

For this approach, B_k and C_k are chosen as the constraint Jacobians $\frac{dG}{dw}(w_k)$ and $\frac{dH}{dw}(w_k)$, and $m_k^T = \nabla F(w_k)^T$ is the gradient of the cost function in NLP (4).

Linearization of the continuity conditions (3b) and consistency conditions (3c) yields

$$0 = \frac{\partial x_i}{\partial s_i^x} \Delta s_i^x + \frac{\partial x_i}{\partial s_i^z} \Delta s_i^z + \frac{\partial x_i}{\partial q_i} \Delta q_i - \Delta s_{i+1}^x + c_i, \quad (7a)$$

$$0 = \frac{\partial g_i}{\partial s_i^x} \Delta s_i^x + \frac{\partial g_i}{\partial s_i^z} \Delta s_i^z + \frac{\partial g_i}{\partial q_i} \Delta q_i + g_i, \quad (7b)$$

where we use the shorthands $c_i = x_i(\tau_{i+1}; s_i^x, s_i^z, q_i) - s_{i+1}^x$ and $g_i = g(s_i^x, s_i^z, q_i)$. Following the lines of a partial reduced SQP method (PRSQP) as proposed by Leineweber [1999] and described for NMPC in Diehl [2002], we eliminate the step $(\Delta s_0^z, \dots, \Delta s_N^z)$ in the algebraic variables from the QP subproblem by using the invertibility of $\frac{\partial g}{\partial z}$ due to the index 1 assumption for the DAE model. We obtain

$$\begin{aligned} \Delta s_i^z &= - \left(\frac{\partial g_i}{\partial s_i^z} \right)^{-1} \left(\frac{\partial g_i}{\partial s_i^x} \Delta s_i^x + \frac{\partial g_i}{\partial q_i} \Delta q_i + g_i \right) \\ &=: D_i^x \Delta s_i^x + D_i^q \Delta q_i + d_i^g. \end{aligned} \quad (8)$$

Because the Jacobian $\frac{\partial q_i}{\partial s_i^z}$ is usually sparse, we can efficiently compute D_i^x, D_i^q, d_i^g by applying a sparse direct solver, in our case the Harwell subroutine MA48 [Reid and Duff, 1993]. Inserting this step elimination explicitly in the continuity conditions, one obtains

$$\begin{aligned} 0 &= \left(\frac{\partial x_i}{\partial s_i^x} + \frac{\partial x_i}{\partial s_i^z} D_i^x \right) \Delta s_i^x + \left(\frac{\partial x_i}{\partial q_i} + \frac{\partial x_i}{\partial s_i^z} D_i^q \right) \Delta q_i \\ &\quad - \Delta s_{i+1}^x + \frac{\partial x_i}{\partial s_i^z} d_i^g + c_i, \\ &=: \mathcal{V}_i^x \Delta s_i^x + \mathcal{V}_i^q \Delta q_i - \Delta s_{i+1}^x + \mathcal{V}_i^g + c_i, \end{aligned} \quad (9)$$

where $\mathcal{V}_i^x, \mathcal{V}_i^q, \mathcal{V}_i^g$ can be interpreted as (forward) directional derivatives of $x_i(\tau_{i+1}; s_i^x, s_i^z, q_i)$. This is exploited in the PRSQP approach by using specially tailored integrator schemes, which efficiently compute these directional derivatives. Following the concept of real-time iterations, only one SQP step to NLP (3) is performed with the controls and states of the preceding step as an initial guess and deliberately allowing a violation of the initial value constraint (3d).

3.2 Adjoint-based algorithm

As recently has been investigated in [Wirsching et al., 2006], one can save a considerable amount of computational effort by solving the quadratic subproblem (6) with *approximated* constraint Jacobians B_k and, possibly, C_k and a *modified* gradient

$$\begin{aligned} m(w_k, \lambda_k, \mu_k)^T &:= \nabla F(w_k)^T + \lambda_k^T \left(B_k - \frac{dG}{dw}(w_k) \right) \\ &\quad + \mu_k^T \left(C_k - \frac{dH}{dw}(w_k) \right). \end{aligned} \quad (10)$$

In this case, only the constraint residuals and the adjoint derivatives $\lambda_k^T \frac{dG}{dw}(w_k)$ and $\mu_k^T \frac{dH}{dw}(w_k)$ have to be computed

exactly. The sensitivities involved can be computed very efficiently as described in Section 4. However, conditions on the "closeness" of the approximated Jacobians to the exact ones have to be posed to ensure local convergence of the method, cf. [Wirsching, 2006].

For the adjoint-based NMPC algorithm presented in the article, we suggest to approximate only a subset of B_k , namely the sensitivities $\frac{\partial x_i}{\partial s_i^x}, \frac{\partial x_i}{\partial s_i^z}, \frac{\partial x_i}{\partial q_i}$ in the linearized continuity conditions (7a). This is motivated by the fact that these particular constraints constitute by far the most expensive part in constraint linearization. Let us denote the approximations by X_i^x, X_i^z and X_i^q , respectively. With this at hand, we only have to compute approximations $X_i^x + X_i^z D_i^x, X_i^q + X_i^z D_i^q$ and $X_i^z d_i^g$ of the directional derivatives in (9) to eliminate the algebraic step variables from the QP (6). For the modified gradient, we first compute the adjoint sensitivities $\tilde{\lambda}_i^T \frac{\partial x_i}{\partial s_i^x}, \tilde{\lambda}_i^T \frac{\partial x_i}{\partial s_i^z}$ and $\tilde{\lambda}_i^T \frac{\partial x_i}{\partial q_i}$ ($\tilde{\lambda}$ being the multipliers of the continuity conditions) and then assemble the terms $\tilde{\lambda}_i^T \mathcal{V}_i^x, \tilde{\lambda}_i^T \mathcal{V}_i^q$ and $\tilde{\lambda}_i^T \mathcal{V}_i^g$ by using D_i^x, D_i^q, d_i^g explicitly. We will comment on the specific choice of the approximations X_i^x, X_i^z and X_i^q in Subsection 5.2.

Note that in both algorithms the size of the QP actually solved is further reduced by employing a condensing step [Bock and Plitt, 1984]. In this step one exploits the special structure due to the multiple shooting discretization as well as the initial conditions to eliminate the step $(\Delta s_0^x, \dots, \Delta s_N^x)$ in the differential variables, leaving the step $(\Delta q_0^x, \dots, \Delta q_{N-1}^x)$ in the control variables as remaining degrees of freedom in the QP.

3.3 Computational complexity

Comparing the discussed approaches, we obtain a computational effort of $n_x + n_u + 1$ (forward) directional derivatives per multiple shooting interval needed in the PRSQP approach versus only one gradient evaluation per multiple shooting interval in the new adjoint based approach. Since both a gradient and a directional derivative can be obtained by a small multiply of the cost for the corresponding function evaluation, this shows a significant reduction of computational complexity of our new approach when compared to the PRSQP approach.

4. EFFICIENT COMPUTATION OF ADJOINT DERIVATIVES

A key to an efficient implementation of the proposed algorithm lies in the fast computation of the adjoint derivatives needed in (10). To achieve this goal we employ the idea of internal numerical differentiation (IND), originally invented by Bock [1981], to obtain the sensitivities of the solution of the stiff initial value problems (IVPs) occurring in our algorithm with respect to initial values, parameters, and controls. More specifically we use a recently developed *reverse* mode of IND, which is implemented in the integrator DAESOL-II, a C++ code based on variable-order variable-stepsize BDF-Formulae that is part of the SOLVIND integrator suite [Albersmeyer and Kirches, 2007].

The main theoretical idea of IND is to freeze all adaptive components of the integrator, such as stepsize and

order control, which could introduce indifferentiabilities to the integration process, after computation of the solution of the initial value problem. The resulting discretization scheme can be understood as a sequence of differentiable mappings leading from the start values to the IVP solutions. This sequence of mappings can be differentiated using the chain rule and the knowledge about the derivatives of each of the mappings, which are essentially the individual integration steps.

Doing this in an adjoint way, e.g using the principles of the reverse mode of automatic differentiation, we obtain a numerical scheme for computing reverse sensitivity information of the solution of the IVP. This approach leads to useful sensitivity approximations even for relatively low integration tolerances and is stable in the sense of reverse analysis. A thorough introduction to automatic differentiation can be found in [Griewank, 2000]. For details on the reverse mode of IND see [Albersmeyer and Bock, 2006].

An unavoidable drawback, as in the reverse approach of automatic differentiation, is the need to store the computed solution trajectory, or if employing checkpointing strategies, to recompute parts of the trajectory. Furthermore the storage of intermediate adjoint quantities is needed. In the case of BDF-methods it can be shown that the intermediate storage can be limited to approximately 7 times the trajectory size. Additionally, in our numerical scenarios all the trajectory information needed can be kept in the main memory, so these limitations of the reverse approach do not play a practical role in our case.

Overall, by applying the reverse mode of IND we are able to generate efficiently a directional gradient of the solutions of the IVP. Theoretically, the effort can be bounded by 5 times the cost of a system simulation, independent of the number of variables, parameter and controls. However, a factor well below 5 can be expected in our case, as the main computational cost of the integration of large-scale system lies in the factorization of matrices for the solution of linear systems. Using the IND approach, the factorizations already performed in the computation of the IVP solution can be also be used for the solution of the linear systems occurring during adjoint sensitivity generation. So this work has only to be done once for the solution of the nominal trajectories.

5. APPLICATION TO THERMALLY COUPLED DISTILLATION COLUMNS

The adjoint-based NMPC scheme is applied to the nonlinear DAE model of thermally coupled distillation columns. The system consists of a main column with a side withdrawal to a rectifying column (Fig. 1). The coupled column is used to separate a ternary mixture of Methanol (light boiler, A), Ethanol (intermediate boiler, B) and 1-Propanol (heavy boiler, C). The energy saving potential of thermally coupled distillation columns has attracted interest of the process industries (see [Finn, 1996, Segovia-Hernandez et al., 2004, Demicoli and Stichlmair, 2004] and references therein).

5.1 The process model

The process model that we use for simulation purposes is taken from Itigin et al. [2003]. The same model also

served as an example for the real-time NMPC algorithms described in Schäfer et al. [2007].

The main column consists of 42 stages (including boiler and condenser stage). The side withdrawal is located at stage 11, the feed enters the column at stage 21. The rectifying column consists of 10 stages and an additional condenser stage. The model is derived under a number of typical assumptions:

- chemical and thermal equilibrium on each stage,
- constant liquid holdup on all stages,
- negligible vapor holdup,
- perfect mixing with ideal gas phase,
- constant pressure throughout the columns,
- total condenser behavior,
- saturated feed and reflux liquid flows.

A total mass balance and component mass balances for A and B yield differential equations for the composition on each tray. The phase equilibrium of the ternary mixture on each tray is described by constant volatilities relative to 1-Propanol in form of algebraic equations. The concentration of component C in both liquid and vapor phase is computed with the summation equation, leading to another set of algebraic equations. The interplay of temperatures and concentrations on each tray are described by the Antoine equation.

The overall nonlinear DAE model comprises 106 differential states describing molar concentrations of two components on all 53 trays, and 159 algebraic states for the concentrations of A and B in the gas phase and for concentration of C in the liquid phase.

The column system is controlled in a D-V configuration with the two distillate flow rates D_1 and D_2 and the boil-up rate V serving as manipulating variables (MVs), thus $u = (D_1, D_2, V)^T$. The controlled variables (CVs) are the molar concentrations in the liquid phase of Methanol in the main column distillate (x_A), Ethanol in the rectifying column distillate (x_B), and 1-Propanol in the main column bottoms (x_C), and we write $y = (x_A, x_B, x_C)^T$.

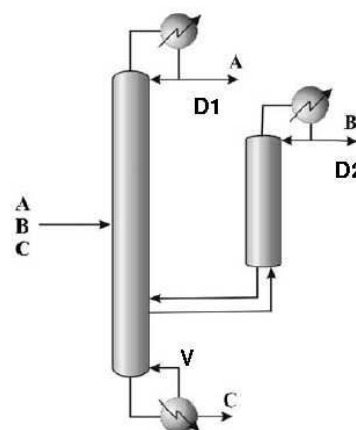


Fig. 1. Coupled distillation column with side withdrawal for the separation of a ternary mixture of A, B, and C (the schematic drawing is taken from Segovia-Hernandez et al. [2004]). The controls are the boil-up V and the distillate flows D_1 and D_2 .

5.2 NMPC formulation

To illustrate the adjoint-based real-time algorithm, the following scenario is simulated: The distillation column is started in steady-state. Then, at time $t = 600$ a measured disturbance in the feed flow composition occurs: The feed concentration of Methanol drops from 0.3 to 0.2, while the Ethanol concentration in the feed is increased from 0.1 to 0.15. The task for the NMPC algorithm is to steer the CVs back to their steady state values. A crucial question is the choice of the approximations X_i^x , X_i^z and X_i^q . In the example presented here, the adjoint-based algorithm uses the exact sensitivities $\frac{\partial x_i}{\partial s_i^x}$, $\frac{\partial x_i}{\partial s_i^z}$, $\frac{\partial x_i}{\partial q_i}$, evaluated at $t = 1000$, for all steps later than $t = 1000$. For any steps before, the exact PRSQP algorithm is used. This choice is motivated by the observation, that the system dynamics and sensitivities change quite significantly in the steps immediately after the disturbance. Thus, fixing the approximations too early after the disturbance would lead to bad behaviour or even failure of the approach, because the approximations then differ too much from the exact sensitivities in the later steps. A possible remedy, also for the case of frequently occurring disturbances, is to recalculate the sensitivities each M steps, where M is a small positive integer.

Further, a number of choices have to be made with respect to the NMPC formulation (2). The cost function is chosen to be of quadratic type as $L(x, z, t) = y(t)^T Q y(t) + u(t)^T R u(t)$ with diagonal weighting matrices Q , R . The diagonal elements are $Q_{ii} = 100$ and $R_{11} = 0.1$, $R_{22} = 0.1$, and $R_{33} = 10$, respectively. The sampling time is given as $\Delta t = 100$ s and the prediction and control horizon is chosen as $T = 3600$ s. This horizon is divided into six equidistant multiple shooting intervals with piecewise constant controls. Most of these NMPC parameters are chosen according to the time constants of the process to be controlled.

The process itself is open-loop stable and it proved unnecessary to use especially designed stabilizing NMPC schemes. In the example we consider state feedback, i.e. we assume that all needed states are measured.

5.3 Numerical results

The simulations have been run on an Intel Pentium 4 machine with 2.8 GHz, 1024 kB L2 cache, 1 GB main memory, under Linux operating system Suse 10.1. The integration accuracy was set to 10^{-4} . A comparison of the results for the test scenario can be seen in Fig. 2 for the CVs and in Fig. 3 for the MVs. Both the PRSQP and the adjoint-based algorithm qualitatively result in the same control moves. The CVs, particularly x_A , show some differences which are due to the high sensitivity of the dynamics to changes in the MVs.

6. CONCLUSIONS AND OUTLOOK

We presented a real-time iteration scheme for NMPC based on adjoint sensitivity generation and inexact constraint linearization. Both the new iteration scheme and an existing iteration scheme were applied to a test problem from the field of chemical process engineering. Producing almost identical profiles in the manipulating variables and quite similar profiles in the controlled variables, the new

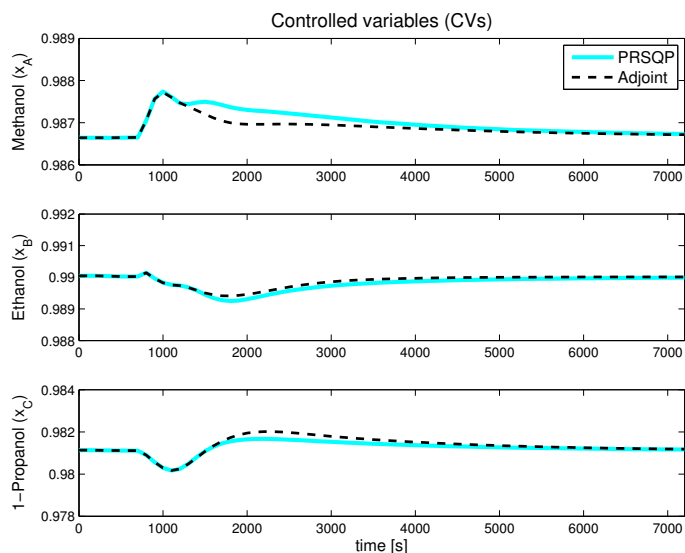


Fig. 2. A comparison of the NMPC closed loop responses for CVs x_A , x_B , and x_C to a step disturbance in the feed composition: PRSQP algorithm (light solid) vs. adjoint-based algorithm (dark dashed).

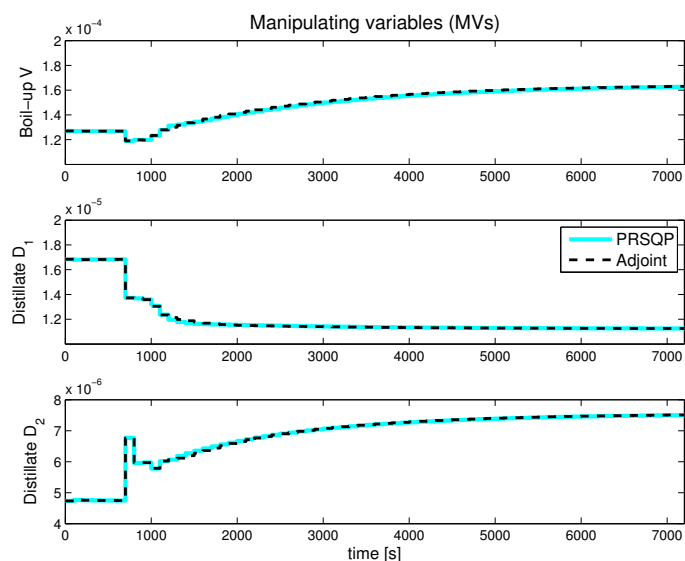


Fig. 3. Corresponding manipulating variables (MVs) for the disturbed feed composition: PRSQP algorithm (light solid) vs. adjoint-based algorithm (dark dashed).

approach has a significantly lower computational complexity than the existing approach. The performance of the new approach, however, strongly depends on the quality of the constraint Jacobian approximation and hence on the state the linearization is based on. It remains an open problem to decide *online* whether the approximated constraint Jacobian is sufficiently close to the exact one. Instead of keeping the linearizations fixed for all times, an appealing idea is to update the constraint linearization at times. The detailed nature of such an updating scheme is subject to ongoing research.

ACKNOWLEDGEMENTS

The authors would like to thank all members of the co-operative project cluster (DFG-Paketantrag) "Optimization-based control of processing plants" comprising of the research groups from Aachen (W. Marquardt), Dortmund (S. Engell), and Stuttgart (F. Allgöwer) for the fruitful and inspiring discussions.

REFERENCES

- J. Albersmeyer and H.G. Bock. Sensitivity Generation in an Adaptive BDF-Method. In *Proc. of Conference on High Performance Scientific Computing (HPSC)*, Hanoi, Vietnam, 2006.
- J. Albersmeyer and C. Kirches. The SOLVIND webpage, 2007. URL <http://www.solvind.org>.
- F. Allgöwer and A. Zheng. *Nonlinear Predictive Control*, volume 26 of *Progress in Systems Theory*. Birkhäuser, Basel Boston Berlin, 2000.
- R. Donald Bartusiak. NLMPC: A platform for optimal control of feed- and product-flexibel manufacturing. In *Proc. Int. Workshop on Assessment and Future Directions of NMPC*, 2005.
- L.T. Biegler. Efficient solution of dynamic optimization and NMPC problems. In F. Allgöwer and A. Zheng, editors, *Nonlinear Predictive Control*, volume 26 of *Progress in Systems Theory*, pages 219–244, Basel Boston Berlin, 2000. Birkhäuser.
- H.G. Bock. Numerical treatment of inverse problems in chemical reaction kinetics. In K.H. Ebert, P. Deuffhard, and W. Jäger, editors, *Modelling of Chemical Reaction Systems*, volume 18 of *Springer Series in Chemical Physics*, pages 102–125. Springer, Heidelberg, 1981.
- H.G. Bock and K.J. Plitt. A multiple shooting algorithm for direct solution of optimal control problems. In *Proceedings 9th IFAC World Congress Budapest*, pages 243–247. Pergamon Press, 1984.
- H.G. Bock, M. Diehl, P. Kühn, E. Kostina, J.P. Schlöder, and L. Wirsching. Numerical methods for efficient and fast nonlinear model predictive control. In *Proc. of "Int. Workshop on assessment and future directions of Nonlinear Model Predictive Control"*, 2005.
- B.L. Braunschweig, C.C. Pantelides, H.I. Britt, and S. Sama. Process modeling: the promise of open software architectures. *Chemical Engineering Progress*, 96:65–76, 2000.
- Mark Cannon. Efficient nonlinear model predictive control algorithms. *Annual Reviews in Control*, 28:229–237, 2004.
- D. Demicoli and J. Stichlmair. Separation of ternary mixtures in a batch distillation column with side withdrawal. *Computers and Chemical Engineering*, 28(1): 643–650, 2004.
- M. Diehl. *Real-Time Optimization for Large Scale Nonlinear Processes*, volume 920 of *Fortschr.-Ber. VDI Reihe 8, Meß-, Steuerungs- und Regelungstechnik*. VDI Verlag, Düsseldorf, 2002. Download also at: <http://www.ub.uni-heidelberg.de/archiv/1659/>.
- M. Diehl, H.G. Bock, J.P. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer. Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. *J. Proc. Contr.*, 12(4): 577–585, 2002.
- H. Elmqvist, S.E. Mattsson, and M. Otter. Modelica: The new object-oriented modeling language. In *Presented at the 12th European Simulation Multiconference*, Manchester, UK, 1998.
- S. Engell. Feedback control for optimal process operation. *Journal of Process Control*, 17:203–219, 2007.
- R. Findeisen, F. Allgöwer, and L. Biegler, editors. *Assessment and Future Directions of Nonlinear Model Predictive Control*. Lecture Notes in Control and Information Sciences. Springer, 2006.
- A. J. Finn. Rapid assessment of thermally coupled side columns. *Gas. Sep. Purif.*, 10(3):169–175, 1996.
- A. Griewank. *Evaluating Derivatives, Principles and Techniques of Algorithmic Differentiation*. Number 19 in *Frontiers in Appl. Math.* SIAM, Philadelphia, 2000.
- A. Itigin, J. Raisch, T. Moor, and A. Kienle. A two-level hybrid control strategy for the start-up of a coupled distillation plant. In *European Control Conference*, Cambridge, UK, September 1-4 2003.
- D.B. Leineweber. *Efficient reduced SQP methods for the optimization of chemical processes described by large sparse DAE models*, volume 613 of *Fortschritt-Berichte VDI Reihe 3, Verfahrenstechnik*. VDI Verlag, Düsseldorf, 1999.
- D.B. Leineweber, I. Bauer, A.A.S. Schäfer, H.G. Bock, and J.P. Schlöder. An efficient multiple shooting based reduced SQP strategy for large-scale dynamic process optimization (Parts I and II). *Computers and Chemical Engineering*, 27:157–174, 2003.
- L. Magni and R. Scattolini. Stabilizing model predictive control of nonlinear continuous systems. *Annual Reviews in Control*, 28:1–11, 2004.
- D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained model predictive control: stability and optimality. *Automatica*, 26(6):789–814, 2000.
- S.J. Qin and T.A. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, 11:733–764, 2003.
- J.K. Reid and I.S. Duff. MA48, A fortran code for direct solution of sparse unsymmetric linear systems of equations. Technical Report RAL 93 072, Rutherford Appleton Laboratory, 1993.
- A. Schäfer, P. Kuehl, M. Diehl, J.P. Schlöder, and H.G. Bock. Fast reduced multiple shooting methods for nonlinear model predictive control. *Chemical Engineering and Processing*, 46(11):1200–1214, 2007.
- J. G. Segovia-Hernandez, S. Hernandez, V. Rico-Ramirez, and A. Jimenez. A comparison of the feedback control behavior between thermally coupled and conventional distillation columns. *Computers and Chemical Engineering*, 28:811–819, 2004.
- L. Wirsching. An SQP algorithm with inexact derivatives for a direct multiple shooting method for optimal control problems. Master's thesis, University of Heidelberg, 2006.
- 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*, pages 591–596, 2006.