DYNAMIC OPTIMIZATION OF CONSTRAINED SEMI-BATCH PROCESSES USING PONTRYAGIN'S MINIMUM PRINCIPLE – AN EFFECTIVE QUASI-NEWTON APPROACH

Erdal Aydin [a], Dominique Bonvin [b], Kai Sundmacher [a,c,*]

[a] *Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstraße 1, 39106 Magdeburg, Germany*

[b] *Laboratoire d'Automatique, École Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland*

[c] *Otto-von-Guericke University Magdeburg, Universitätplatz 2, 39106 Magdeburg, Germany*

* Corresponding author: sundmacher@mpi-magdeburg.mpg.de ; Tel: +49 391 6110 350, Fax: +49 391 6110 353

ABSTRACT

This work considers the numerical optimization of constrained batch and semi-batch processes, for which direct as well as indirect methods exist. Direct methods are often the methods of choice, but they exhibit certain limitations related to the compromise between feasibility and computational burden. Indirect methods, such as Pontryagin's Minimum Principle (PMP), reformulate the optimization problem. The main solution technique is the shooting method, which however often leads to convergence problems and instabilities caused by the integration of the co-state equations forward in time.

This study presents an alternative indirect solution technique. Instead of integrating the states and co-states simultaneously forward in time, the proposed algorithm parameterizes the inputs, and integrates the state equations forward in time and the co-state equations backward in time,

thereby leading to a gradient-based optimization approach. Constraints are handled by indirect adjoining to the Hamiltonian function, which allows meeting the active constraints explicitly at every iteration step. The performance of the solution strategy is compared to direct methods through three different case studies. The results show that the proposed PMP-based quasi-Newton strategy is effective in dealing with complicated constraints and is quite competitive computationally.

Keywords: Constrained dynamic optimization, Pontryagin's Minimum Principle, indirect optimization methods, quasi-Newton algorithm, semi-batch processes

1. INTRODUCTION

The optimization of batch and semi-batch processes is becoming more and more popular due to industrial competitiveness and strict environmental regulations. If a reliable dynamic process model is available, dynamic optimization is considered as a promising method for reducing production costs, improving product quality and meeting safety as well as environmental regulations.

The dynamic optimization problem (also called optimal control) for batch processes is often stated as follows (Srinivasan et al., 2003):

$$\begin{aligned} &\min_{t_f, u(t)} J = \phi(x(t_f)) \\ \text{s.t.} \quad &\dot{x} = F(x,u), \quad x(0) = x_0 \\ &S(x,u) \leq 0, \quad T(x(t_f)) \leq 0 \end{aligned} \tag{1}$$

where J is a scalar performance index that depends on the values of the states at the final time t_f, ϕ is the objective function, x is the n_x-dimensional state vector with the corresponding initial conditions x_0, u is the n_u-dimensional input vector, S is the n_S-dimensional vector of inequality path constraints that include input bounds, and T is the n_T-dimensional vector of inequality terminal constraints

The nonlinear differential equations describing the system dynamics are included in the formulation as equality constraints. The solution methods that are available in the literature for this dynamic optimization problem can be classified in two major categories, namely, the direct and indirect (or PMP) approaches (Srinivasan et al., 2003).

In direct optimization approaches, the solution methodology is applied directly to the original optimization problem, Eq. (1), by using either sequential or simultaneous numerical techniques. In the class of direct sequential methods, the input vector is parameterized using polynomial functions, the state equations are integrated from the given initial conditions up to the final time, where the states are needed for evaluating the objective function. The optimal input parameters are computed by a NLP solver (Vassiliadis et al., 1994); (Schlegel & Marquardt, 2006b). The use of time integration is the reason for also naming sequential techniques "feasible-path methods". However, depending on the type of problem and the NLP solver available, a sequential method can be slow and thus computationally expensive, in particular when dealing with path constraints (Srinivasan et al., 2003). Furthermore, in direct sequential methods, the input profiles are often represented using a coarse discretization grid to ensure computational efficiency (Schlegel & Marquardt, 2004). Note, however, that a fine input discretization may be necessary to accurately detect switching times.

In contrast, in the class of direct simultaneous methods, the entire optimization problem (system equations, input profiles, objective function and constraints) is discretized w.r.t. time, using for example collocation techniques, thus resulting in a large system of algebraic equations. Then, an NLP solver simultaneously solves the governing dynamic system equations and optimizes the cost (Cervantes & Biegler, 1998), (Biegler et al., 2002), (Biegler, 2007). Because the system equations are not integrated in time, but approximated at discrete time instants, this approach is called "infeasible-path method". Although simultaneous techniques allow the efficient solution of large-scale optimization problems, the trade-off between approximation and optimization must be considered carefully (Srinivasan et al., 2003).

In indirect optimization approaches, the original optimization task, Eq.(1), is reformulated as the minimization of a Hamiltonian function. The reformulated problem is then solved to satisfy the necessary conditions of optimality that are expressed via Pontryagin's Minimum Principle (PMP) (Bryson, 1975). For simple problems, the optimal solution can sometimes be expressed and computed analytically. More complex and, in particular, constrained problems, require a numerical solution, which is often computed using the shooting method. The main problem of this method is that the integration of the co-state equations forward in time may introduce instabilities that prevent fast convergence unless a good initial guess is available. However, instead of integrating the states and co-states simultaneously forward in time, the inputs can be parameterized and then, sequentially, the state equations are integrated forward in time and the co-states backward in time. Optimization is performed using a gradient-based algorithm, for which a good initial guess is beneficial but not required for convergence (Bryson, 1975), (Srinivasan et al., 2003), (Hartl et al., 1995), (Chachuat, 2007). For a comprehensive overview of the dynamic optimization literature until 2003, the reader is referred to (Srinivasan et al., 2003). More recent publications are given in Table 1.

Table 1. Selection of recent publications.

Publication	Method of Choice	Subject
(Schlegel et al., 2005)	Direct Sequential	Reducing the problem size using adaptive control vector
(Schlegel & Marquardt, 2006a) (Schlegel & Marquardt, 2004)	Direct Sequential	Reducing the problem size using adaptive switching times and structures
(Srinivasan & Bonvin, 2007)	NCO Tracking	Real-Time-Optimization using NCO tracking
(Kadam et al., 2007)	NCO Tracking	Robust optimization using measurements and solution models
(Srinivasan et al., 2008)	NCO Tracking	NCO Tracking using barrier-functions for active constraints
(Biegler, 2007) (Kameswaran & Biegler, 2006)	Direct Simultaneous	Overview of recent direct simultaneous strategies
(Logist et al., 2011)	Direct Multiple Shooting	Robust, multi-objective dynamic optimization
(Assassa & Marquardt, 2014)	Direct Multiple Shooting	Adaptive multiple shooting

The PMP approach has been applied to various engineering optimization problems since the 70's. (Jaspan & Coull, 1971) suggested a boundary condition iteration (BCI) solution scheme for unconstrained chemical reactor optimization problems. For input-affine systems, (Visser et al., 2000) proposed an online optimizing structure that uses a switching function along with the PMP-based optimality conditions; then, a cascade optimization scheme that tracks the necessary conditions of optimality was designed and tested on a fed-batch penicillin fermentation process. (Cannon et al., 2008) designed a model predictive control strategy for input-constrained linear systems using PMP. In this approach, the inputs can be represented in terms of co-states, and the problem can then be solved using active-set methods. (Kim & Rousseau, 2012) used Pontryagin's

principle for the optimal control of hybrid electric vehicles. (Palanki & Vemuri, 2005) proposed an end-point dynamic optimization scheme using PMP for semi-batch processes with a single reaction. (Roubos et al., 1997) studied the use of PMP with an unconstrained gradient-based solution technique for the optimization of fed-batch biological problems. In order to account for path constraints, they penalized the value of the objective function in case of a constraint violation. (Ali & Wardi, 2015) implemented a shooting method, where the inputs are expressed analytically in terms of the states and co-states. (Hannemann-Tamás & Marquardt, 2012) used PMP to verify the inputs computed by a direct sequential method. For a given optimal control problem, they obtained what they called "the true solution" using a PMP-based multiple shooting algorithm for the purpose of verifying the results of the direct sequential optimization algorithm.

Solving dynamic optimization problems that include nonlinear path constraints is a challenging task for PMP-based approaches. Unfortunately, there exists no fast-convergent solution strategy for constrained problems besides the shooting method, whose convergence is dependent on many conditions (Chachuat, 2007), (Hartl et al., 1995). This is the motivation for the present work that proposes a PMP-type gradient-based solution method for optimization problems of semi-batch processes that include nonlinear path and terminal constraints. For this purpose, indirect adjoining is used which augments the Hamiltonian function with mixed state-input constraints. This allows computing certain inputs explicitly so as to satisfy the active path constraints.

The paper is organized as follows. Section 2 presents the solution methodology using the indirect PMP-based approach. The three case studies investigated in Section 3 illustrate the PMP-based quasi-Newton approach developed in this work, and compares it to the direct simultaneous strategy, while Section 4 concludes our study.

2. SOLUTION METHODOLOGY

The dynamic optimization problem given in Eq. 1 can be reformulated using PMP as follows (Srinivasan et al., 2003):

$$\min_{t_f,u(t)} H(t) = \lambda^T F(x,u) + \mu^T S(x,u)$$

$$\text{s.t.} \quad \dot{x} = F(x,u); \quad x(0) = x_0;$$

$$\dot{\lambda}^T = -\frac{\partial H}{\partial x}, \quad \lambda^T(t_f) = \frac{\partial \phi}{\partial x}\Big|_{t_f} + \upsilon^T \frac{\partial T}{\partial x}\Big|_{t_f};$$

$$\mu^T S = 0; \; \upsilon^T T = 0 \tag{2}$$

where H is the Hamiltonian function, λ is the n_x-dimensional vector of Lagrange multipliers (also called co-states or adjoints) for the system equations, μ is the n_S-dimensional vector of Lagrange multipliers for the path constraints, and υ is the n_T-dimensional vector of Lagrange multipliers for the terminal constraints. $\mu^T S = 0$ and $\upsilon^T T = 0$ are the complementary slackness conditions that will be satisfied at the optimum. Moreover, the following necessary conditions must hold at any optimum of H:

$$\frac{\partial H(t)}{\partial u} = \lambda^T \frac{\partial F}{\partial u} + \mu^T \frac{\partial S}{\partial u} = 0 \tag{3}$$

$$H(t_f) = (\lambda^T F + \mu^T S)|_{t_f} = 0 \tag{4}$$

Eq. 3 indicates that the partial derivatives of the Hamiltonian function with respect to the inputs must all be equal to zero at an optimum. If the final time of the dynamic optimization problem is fixed, then Eq. 4 (called transversality condition) is not required (Biegler, 2010). Note that, for each input u_i, the first term on the right-hand side of Eq. 3 is the switching function, $s_{u_i} := \lambda^T \frac{\partial F}{\partial u_i}$, which can be used to determine whether a given arc of the input $u_i(t)$ is constraint-seeking ($s_{u_i} \neq 0$) or sensitivity-seeking ($s_{u_i} = 0$).

Assuming that the system and adjoint equations are differentiable, the proposed solution strategy is as follows:

1) Cast the problem into the solution of differential equations for both the states and the co-states. The latter are obtained by differentiation of the Hamiltonian function with respect to the states as given in Eq. 2. The Matlab Symbolic Toolbox can be used for this purpose. This is only necessary for the initialization of the problem.
2) Use indirect adjoining to deal with pure-state path constraints of the form $S(x) \leq 0$. In this method, the state constraints are differentiated with respect to time until at least one of the inputs appears explicitly (Hartl et al., 1995). The resulting expression is $S^{\{n\}}(x,u) \leq 0$, where n represents the relative degree of a constraint with respect to an input, that is, the number of differentiations required for an input to appear explicitly (Srinivasan & Bonvin, 2007). Then, instead of the original state constraints $S(x) \leq 0$, the differentiated version $S^{\{n\}}(x,u) \leq 0$ is used to construct the Hamiltonian. The method helps dealing with path constraints when they become active (Onori et al., 2016). Consequently, the Hamiltonian function reads $H(t) = \lambda^T F(x,u) + \mu^T S^{\{n\}}(x,u)$. Because of the complementary slackness $\mu^T S^{\{n\}}(x,u) = 0$, the penalty term $\mu^T S^{\{n\}}(x,u)$ vanishes when all the constraints are satisfied. However, if some of the constraints are not satisfied during the course of optimization, the penalty term $\mu^T S^{\{n\}}(x,u)$ will be positive, thereby forcing convergence to occur through the feasible region.

   Remark 1. Input saturation can also be implemented such that

   $$u(t) = \begin{cases} u_{min}, & \text{if a lower constraint is violated} \\ u_{max}, & \text{if an upper constraint is violated} \end{cases}.$$

3) Discretize the input profiles as $u(t) = U(U)$, where U is a (n_u x N) matrix that contains N discrete input values for the n_u inputs. For example, the input profiles can be

approximated by piecewise-constant functions. The choice of *N* will depend on the nature of the problem.

4) Initialize *U* corresponding to the initial input profiles, and first integrate the state equation forward in time. Then, integrate the co-state equations backward in time.

5) Discretize the Lagrange multiplier vectors as μ(t) = M (*M*), where *M* is a (n_S x *N*) matrix. If the condition $S_j^{\{n\}}(x, U(.,k)) = 0$ is satisfied at the discrete time instant *k*, set *M*(j,k) =*K* > *0*, and compute the value of *U*(.,*k*) that makes $S_j^{\{n\}}(x, U(.,k)) = 0$. Otherwise, set *M*(j,k) = 0 (Harvey Jr et al., 2012).

Remark 2. Here, the choice of the value of K is arbitrary. Yet, (Onori et al., 2016) suggested to choose *K* as large as possible to guarantee the feasibility of the path constraints.

6) Update the input values *U* via optimization until a pre-defined optimality criterion is satisfied, such as the threshold value ε for $norm(\frac{\partial H}{\partial U})$. A quasi-Newton algorithm is proposed for optimization. Adjoining the inequality path constraints into the Hamiltonian enables the original constrained optimization problem to be solved as an unconstrained problem. Furthermore, the penalty terms $\mu^T S^{\{n\}}(x,u)$ ensure that the update direction goes through the feasible region. Because the use of a Hessian might be problematic and result in singularities, a robust BFGS update algorithm that ensures the positive-definiteness of the Hessian matrix is used (Biegler, 2010). Furthermore, the Hessian matrix is updated if the iteration is inside the feasible region. Otherwise, the Hessian matrix remains the same, and the optimization direction is set as $\alpha\left(\frac{\partial H}{\partial U}\right)$, that is, a steepest-descent algorithm is used.

The optimization algorithm can be formulated as follows:

PMP-based Quasi-Newton Optimization Algorithm

Initialize the iteration counter $h = 0$ and the corresponding input elements U_0 and the Hessian matrix $B_0 := \boldsymbol{I}$.

for h = 1 → ∞

I. Solve the state equations by forward integration and the co-state equations by backward integration. If the *j*-th constraint is satisfied at the discrete time instant *k*, set M_h (j,k) := 0. Otherwise, set M_h (j,k) := *K* > *0*, for *k*=1,..,N.

II. Compute the value of the first-order gradients $(\frac{\partial H}{\partial U})_h$ by using pre-computed analytical expressions.

III. **If** all constraints are satisfied, set $U_{h+1} := U_h - \alpha({B_h}^{-1}\frac{\partial H}{\partial U})_h$ and update the Hessian matrix B_h as follows:

$y := \nabla H(U_{h+1}) - \nabla H(U_h),\ s := U_{h+1} - U_h$,

if $s^T y \geq \beta\|s\|^2$, set $B_{h+1} := B_h + \frac{yy^T}{s^T y} - \frac{B_h s s^T B_h}{s^T B_h s}$,

else set $B_{h+1} := B_h$

end if

else set $U_{h+1} := U_h - \alpha(\frac{\partial H}{\partial U})_h$ and $B_{h+1} := B_h$, and compute the value of *U*(.,*k*) that makes the violated path constraint $S_j^{\{n\}}(x, U(.,k)) = 0$ at the discrete time instant *k*.

end if

IV. **If** $norm(\frac{\partial H}{\partial U})_h < \varepsilon$ and $h > 35$, set $U_{opt} := U_h$, **stop**

end for

Remark 3. Since the choice of K is arbitrary, the computed gradients might be ill conditioned. To avoid this, the gradients used in the optimization are scaled.

Remark 4. Regarding the choice of the step size α, values between 0.01 and 0.1 are usually effective for scaled problems.

Remark 5. The optimal solution often contains active constraints. However, if the proposed algorithm computes $M_1 = 0$ at the first iteration, it is likely that the iterative scheme converges to a sub-optimal solution with all the constraints satisfied but inactive. To prevent this, a threshold on the number of iterations (e.g., $h > 35$) is used, which would enable the algorithm to search for active constraints. With the examples given in this paper, it is observed that 35 iterations suffice; however, for difficult problems, larger values might be necessary.

Remark 6. A line-search algorithm can also be used as opposed to a fixed step size. Furthermore, constraint softening is highly recommended, because errors due to numerical integration or round off might contribute to larger convergence times.

3. CASE STUDIES

For illustrating the application of the proposed methodology to the dynamic optimization of constrained fed-batch processes, three case studies are presented in this section. The first problem is a dynamic reactor optimization taken from (Srinivasan et al., 2003). The second problem is the dynamic optimization of a batch binary distillation column with terminal purity constraints. The

third problem involves the dynamic optimization of a complex fed-batch chemical process taken from (Hentschel et al., 2015).

All problems were solved using both a direct simultaneous method and the indirect PMP-based quasi-Newton proposed in this work. The CasADi toolbox (Andersson & Diehl, 2012) was applied for the implementation of the direct simultaneous method, along with the nonlinear programming solver IPOPT (Wächter & Biegler, 2006). All computational results, including the initialization of the problems, were obtained with an Intel i-3-2100 machine (CPU 3.10 GHz 4 GB RAM).

3.1. PROBLEM - 1: *Non-isothermal semi-batch reactor with a heat-removal constraint*

Consider a fed-batch reactor in which the following series reactions take place:

$$A + B \overset{k_1}{\rightarrow} C \overset{k_2}{\rightarrow} D \tag{5}$$

The objective is to maximize the molar content of the desired product C at a specified final time (Srinivasan et al., 2003). The two inputs are the feedrate of $B, u(t)$, and the reactor temperature, $T(t)$. The path constraints include input bounds as well as upper limits on the heat generated by the chemical reactions, q_{rx}, and the reactor volume, V. Note that an energy balance is not considered explicitly, but the temperature effect is included in q_{rx} as proposed by (Srinivasan et al., 2003). The final time t_f is fixed at 0.5 h. Accordingly, the optimization problem can be formulated as follows:

$$\max_{u(t),T(t)} J = c_c(t_f)V(t_f)$$

s.t.

$$\dot{c_A} = -k_1 c_A c_B - \frac{u}{V} c_A; \qquad c_A(0) = c_{A0}\ ;$$

$$\dot{c_B} = -k_1 c_A c_B + \frac{u}{V}\left(c_{B,in} - c_B\right); \qquad c_B(0) = c_{B0}\ ;$$

$$\dot{c_C} = k_1 c_A c_B - k_2 c_C - \frac{u}{V} c_C\ ; \qquad c_C(0) = c_{C0}\ ;$$

$$\dot{V} = u\ ; \quad V(0) = V_0\ ;$$

$$k_1 = k_{10} e^{\frac{-E_1}{RT}}\ ; \qquad k_2 = k_{20} e^{\frac{-E_2}{RT}}\ ;$$

$$T_{min} \le T(t) \le T_{max}\ ;$$

$$u_{min} \le u(t) \le u_{max}\ ;$$

$$(-\Delta H_1) k_1 c_A(t) c_B(t) V(t) + (-\Delta H_2) k_2\, c_C(t) V(t) \le q_{rx,max}\ ;$$

$$V(t) \le V_{max} \tag{6}$$

The model parameters, initial conditions and constraints are given in Table 2.

Table 2. Model parameters, initial conditions and constraints for Problem 1.

k_{10}	4 l/(mol h)	T_{min}	293.15 K
k_{20}	800 l/h	T_{max}	323.15 K
E_1	6×10^3 J/mol	V_{max}	1.1 L
E_2	20×10^3 J/mol	$q_{rx,max}$	1.5×10^5 J/h
R	8.314 J/(mol K)	c_{A0}	10 mol/L
ΔH_1	-3×10^4 J/mol	c_{B0}	1.1685 mol/L
ΔH_2	-10^4 J/mol	c_{C0}	0 mol/L
u_{min}	0 L/h	V_0	1 L
u_{max}	1 L/h	$c_{B,in}$	20 mol/L

Computed optimal solutions

There are several local solutions, three of which are given analytically by (Binette et al., 2016). In fact, any feasible combination of the arcs ($u_{\min}$, u_{path}, $u_{\max}$) and ($T_{\min}$, T_{path}, T_{sens}, $T_{\max}$) described in that paper can be a local solution to the problem.

The optimal input and state profiles computed with different numerical techniques are given in Figs 1 and 2. Fig. 1 shows the PMP-based solutions for the discretization levels N=50 and N=500, along with the analytical solution 2 (Binette et al., 2016). The parameter values α = 0.025, K = 50 and ε = 0.05 are used in the PMP-based approach. Similarly, Fig. 2 shows the direct simultaneous solution for N=50 and N=500 along with the analytical solution 3 (Binette et al., 2016). A few remarks are given at this point:

1. Although all strategies converge to a solution with nearly the same cost (between 2.050 and 2.053 moles of reactant B, as seen in Table 3), there are significant differences in the computed optimal profiles, which is an indication that the two numerical strategies converge to *different* local solutions.
2. The heat-removal constraint is active during the first part of the run. The volume constraint is active in the second part of the run (batch mode with zero feed).
3. With the PMP-based solution strategy, the input profiles are not too close to the analytical solution 2 in the second arc characterized by T_{sens}. This is due to the lack of sensitivity of the objective function with respect to the inputs.
4. In the direct simultaneous solution for N=50, the heat-removal constraint is not active initially because the time discretization is too coarse. However, the constraint becomes active when N is increased.

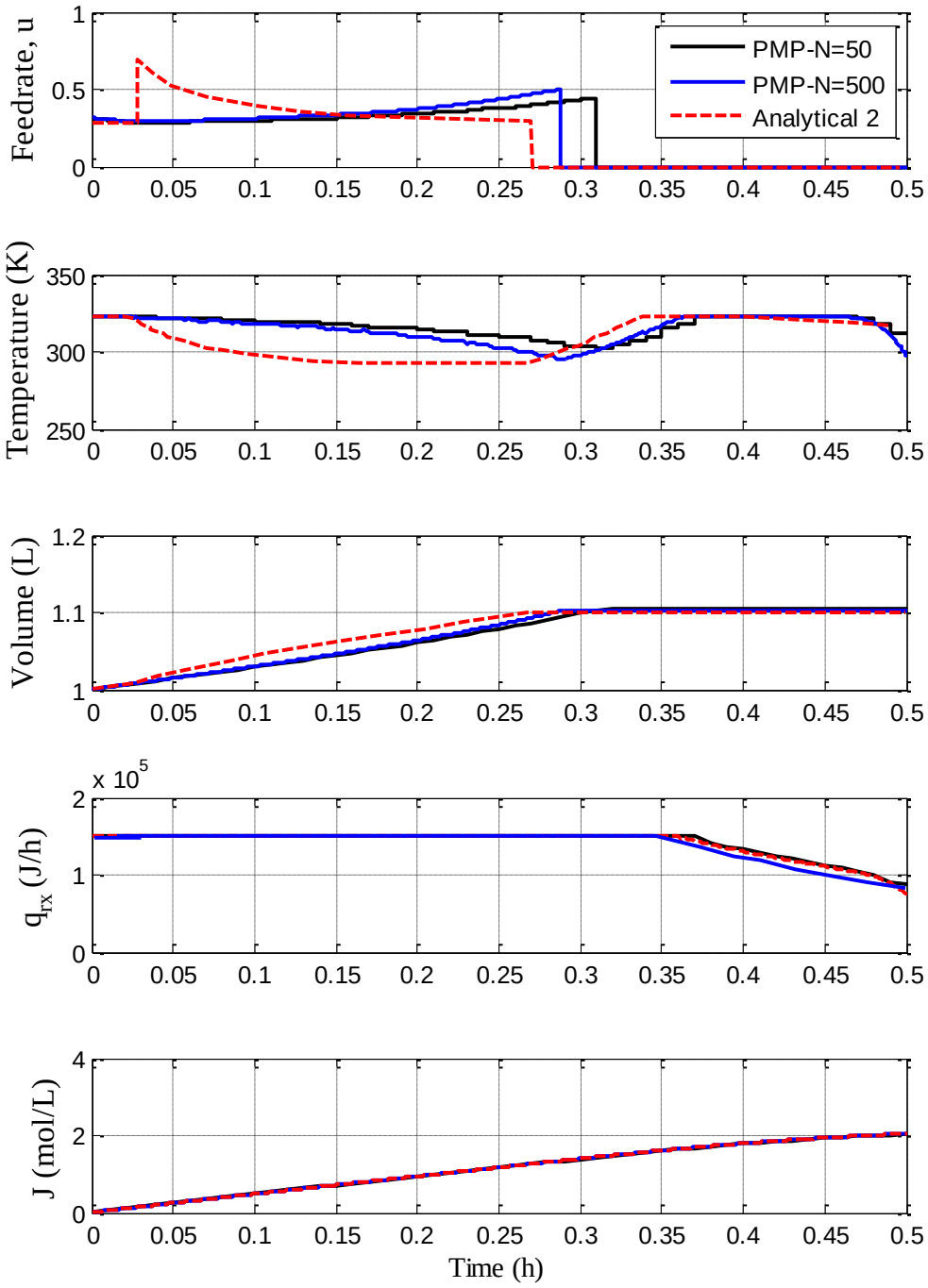

Figure 1. Optimal input and state profiles computed via the PMP-based method and the analytical solution 2 for Problem 1.

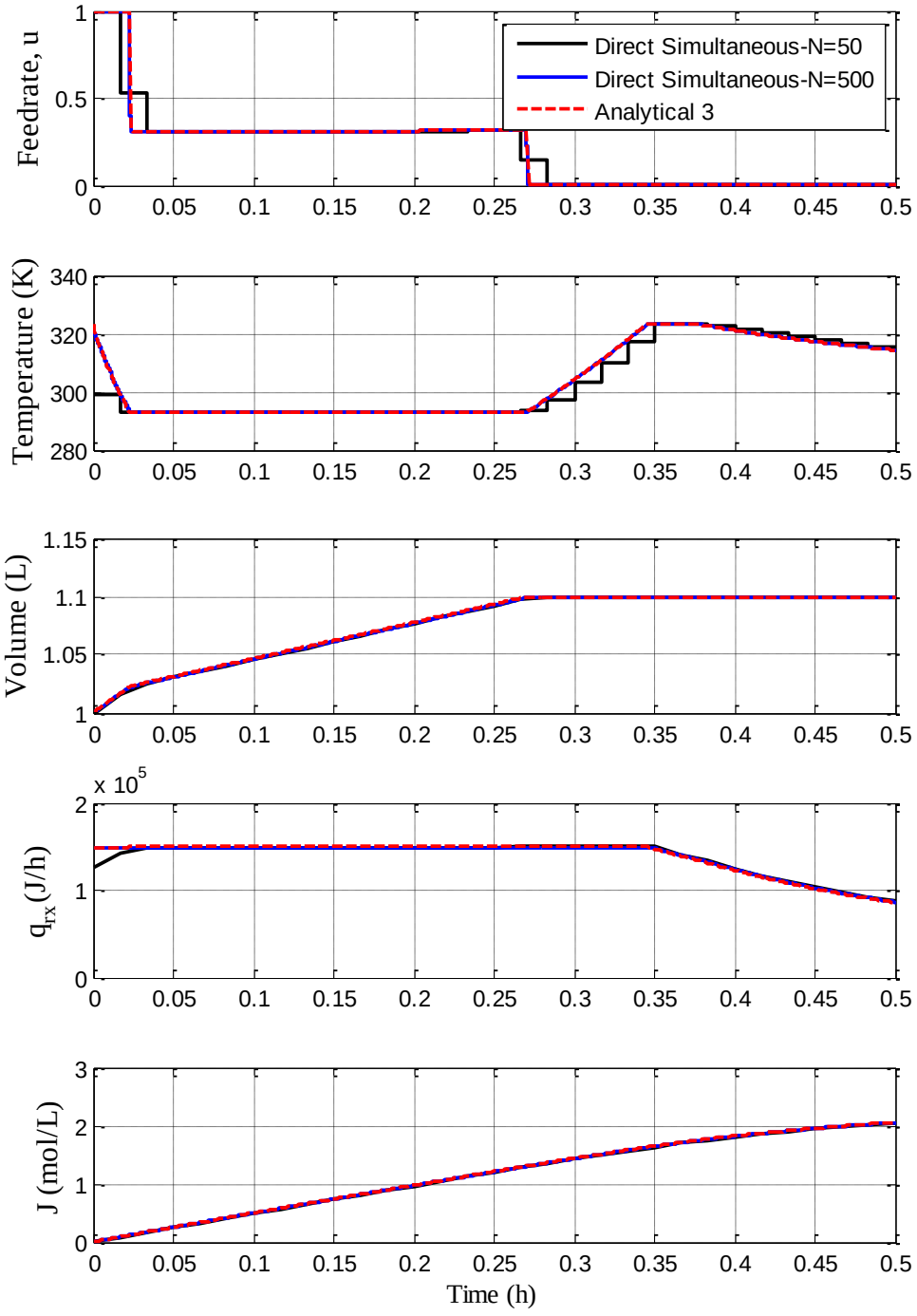

Figure 2. Optimal input and state profiles computed via the direct simultaneous method and the analytical solution 3 for Problem 1.

The switching functions s_u and s_T computed at the optimal solution are given in Figs 3 and 4 for the PMP-based solution and the direct simultaneous solution, respectively. One can see that s_u is never zero, which means that the feed rate u is never sensitivity seeking. In contrast, $s_T = 0$ in certain intervals, which are therefore sensitivity seeking.

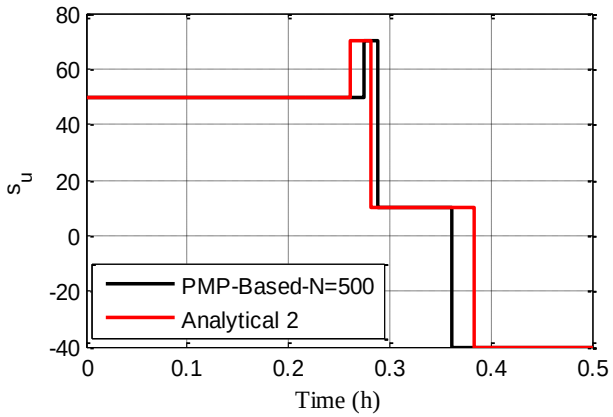

Figure 3a. Switching function s_u for the PMP-based solution and the analytical solution 2.

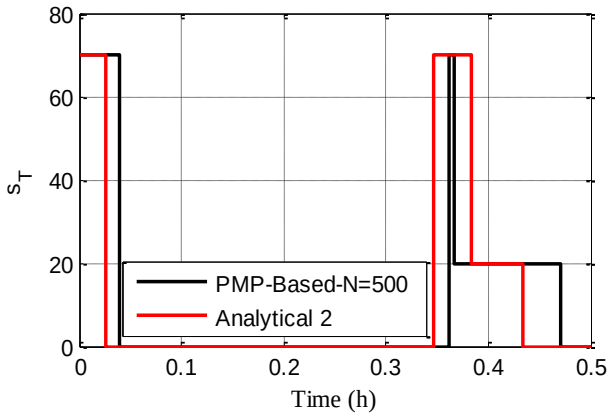

Figure 3b. Switching function s_T for the PMP-based solution and the analytical solution 2.

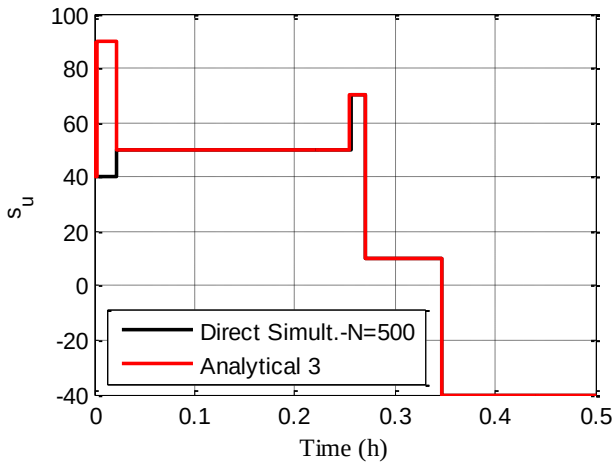

Figure 4a. Switching function s_u for the direct simultaneous solution and the analytical solution 3.

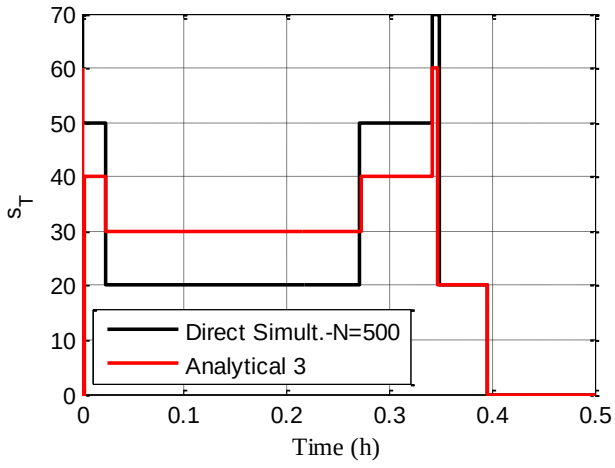

Figure 4b. Switching function s_T for the direct simultaneous solution and the analytical solution 3.

Table 3. Comparison of the indirect PMP-based, direct simultaneous and analytical solution strategies for Problem 1.

Optimization Strategy	**Solution Structure**				
	Arc 1	**Arc 2**	**Arc 3**	**Arc 4**	**Arc 5**
Indirect PMP-based N=50, J=2.050 N=500, J=2.051 *Analytical Solution 2* J=2.050	u_{path} T_{max}	u_{path} T_{sens}	u_{min} T_{path}	u_{min} T_{max}	u_{min} T_{sens}
Direct simultaneous N=50, J=2.051 N=500, J=2.053 *Analytical Solution 3* J=2.053	u_{max} T_{path}	u_{path} T_{min}	u_{min} T_{path}	u_{min} T_{max}	u_{min} T_{sens}

The comparison of the cost values and solution structures obtained with the various strategies are given in Table 3. The two numerical schemes converge to different solutions, which happen to be the analytical solution 2 and 3 given in Binette et al. (2016). The PMP-based solution suggests that the reactor temperature profile starts at its upper bound, with the feed rate u(t) adjusted to satisfy the heat-removal constraint. Then, the temperature follows T_{sens}(t) to find a compromise between producing much of the desired C and not too much of the undesired by-product D. Once the reactor is completely filled, the feed rate is set to zero and the temperature adjusted to still keep the path constraint active. Once T_{max} is reached, the temperature is kept there until there is some advantage in reducing it and following T_{sens}(t) again. Although the input profiles of the PMP-based solution and the analytical solution 2 (Binette et al., 2016) are different, the arc types and sequence are exactly the same.

On the other hand, the direct simultaneous solution comes fairly close to the analytical solution 3. Optimal operation starts with maximal feeding of reactant B, with the temperature being used to meet the heat-removal constraint. When the minimal temperature is reached, it is kept there, and the feed rate is adjusted to keep the path constraint active. Once the reactor is filled, the feed rate is set to zero and the temperature is increased to still keep the path constraint active. From that point on, the sequence of arcs is the same as for the PMP-based solution.

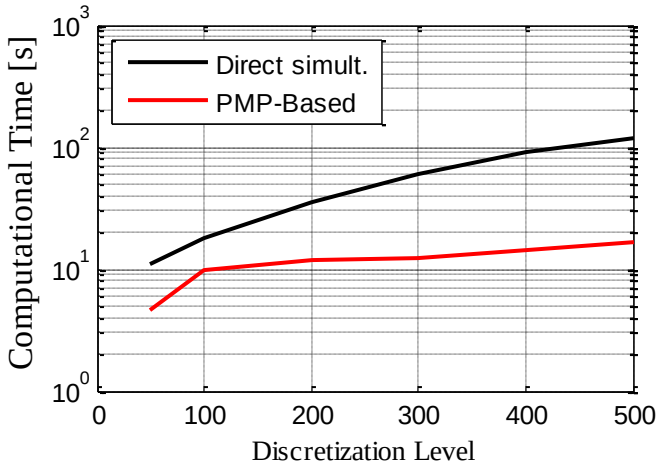

Figure 5. Computational times for different discretization levels N of Problem 1.

Fig. 5 shows the computational time required to obtain the solution with the PMP-based and direct simultaneous methods. It is clearly seen that the PMP-based method requires significantly less computational time when the grid gets finer.

3.2. PROBLEM - 2: *Binary batch distillation column with terminal purity constraints*

The optimization of batch distillation columns using PMP is well documented in the literature. For example, (Coward, 1967) solved a time-optimal problem for a binary batch distillation column using PMP. The solution was based on an adaptive shooting strategy that requires good initial guesses for the adjoints and therefore is difficult to implement. (Mayur &

Jackson, 1971) studied PMP for binary and multicomponent batch distillation problems with adaptive solution techniques. (Welz et al., 2005) used the PMP-based necessary conditions of optimality to design an implicit optimization scheme for a binary batch distillation column. In the present paper, we propose to compare the PMP-based quasi-Newton method with a direct simultaneous approach to optimize the operation of a binary batch distillation column.

Consider a batch distillation column with only three equilibrium plates, in which the components A and B (more volatile) are separated from each other. The objective is to maximize the molar amount of B in the distillate for a given batch time, while satisfying the terminal purity constraints of at least 80 mol.% of B in the distillate and at most 20 mol% of B in the bottom product. The final time t_f is fixed at 3.0 h. The only path constraints are input bounds on the reflux ratio.

A schematic of the column is given in Fig. 6, with the molar amounts B and D in the bottoms and in the distillate tank, respectively, the vapor flow rate V and the liquid flow rate L in the column. The internal reflux ratio $r = \frac{L}{V}$ is the input variable. Assuming perfect mixing on all stages, negligible vapor hold-up, constant vapor flow through the column, total condensation in the condenser of negligible hold-up, constant liquid hold-up on all trays and constant relative volatility, the optimization problem can be stated as follows:

$$\max_{r(t)} \quad J = D(t_f)$$

s.t.

$$\dot{D} = V(1-r); \quad D(0) = 0;$$

$$\dot{B} = V(r-1); \quad B(0) = B_0;$$

$$\dot{x}_B = \frac{V}{B}\left(x_B - y_B + r(x_1 - x_B)\right); \quad x_B(0) = x_{B\,0};$$

$$\dot{x}_k = \frac{V}{M}\left(y_{k-1} - y_k + r(x_{k+1} - x_k)\right); \quad x_k(0) = x_{B_0}; \;\; k = 1,\ldots,3;$$

$$\dot{x}_D = \frac{V(1-r)}{D}(y_3 - x_D); \quad x_D(0) = x_{B_0};$$

$$y_0 = y_B; \quad x_4 = y_3; \quad y_k = \frac{\alpha x_k}{1+(\alpha-1)x_k}; \;\; k = B, 1,\ldots,3;$$

$$x_D(t_f) \geq 0.8;$$

$$x_B(t_f) \leq 0.2;$$

$$0 \leq r(t) \leq 1 \tag{7}$$

where B_0 is the initial charge, x_{B_0} the mole fraction of B in the initial charge, x_k the mole fraction of B in the liquid phase of the k-th tray, y_k the mole fraction of B in the vapor phase leaving the *k*-th tray, x_D the mole fraction of B in the distillate tank, x_B the mole fraction of B in the bottoms, y_B the mole fraction of B in the vapor leaving the bottoms, α the relative volatility, and M the liquid hold-up on each tray. The material balances are written in terms of the more volatile component B. The trays are numbered from the bottom to the top of the column. Because of total condensation, the composition of the refluxed liquid is equal to the vapor composition leaving the upper plate. It is also assumed that all plates are initially charged with the same liquid mixture as the reboiler and thus the initial concentration of B on each tray is x_{B_0}. The dynamic column model has 7 state variables and a single input. The model parameters and initial conditions are given in Table 4.

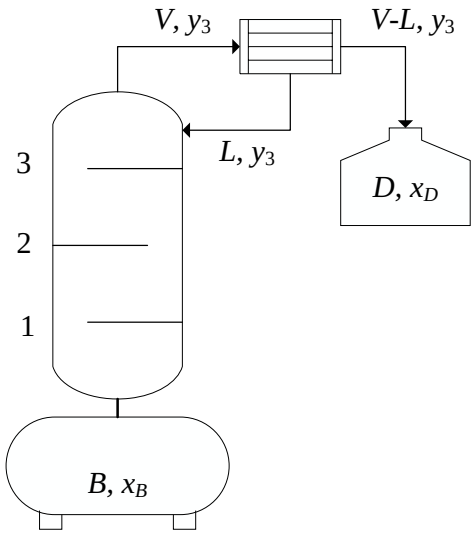

Figure 6. Schematic of the batch distillation column.

Table 4. Model parameters and initial conditions for Problem 2.

Vapor flow rate, V	50 kmol/h
Relative volatility, α	2.25
Initial charge, B_0	115 kmol
Concentration of B in the charge, $x_{B\,0}$	0.4
Molar hold-up per plate, M	5 kmol

Computed optimal solutions

The input vector is parameterized using N equidistant piecewise-constant elements. The terminal constraints are enforced in the PMP-based solution by setting the final values of the adjoints as stated in Eq. 2. The parameter values $\alpha = 0.1$, $K = 100$ and $\varepsilon = 0.05$ are used in the PMP-based approach.

Fig. 7 shows the optimal input and state profiles computed with the two strategies for the discretization level N=500. One sees that both the PMP-based and the direct simultaneous

solutions converge to a 3-arc solution. Both solutions suggest total reflux in the beginning to increase the composition at the top of the column. Then, they both follow a sensitivity-seeking arc to produce as much distillate as possible with the required purity. Finally, a short third arc with zero reflux is used to recover the high-purity material that is still in the column.

The corresponding switching functions are shown in Fig. 8. The following remarks can be made:

1. Since there are no path constraints besides the input bounds, the only possible arcs are r_{max}, r_{min} and r_{sens}.
2. Although the two numerical schemes lead to the same sequences and types of arcs, namely r_{max}, followed by r_{sens} and r_{min}, and nearly the same optimal cost J (cf. Table 5), the computed input profiles are noticeably different. This is due to the lack of sensitivity of the objective function with respect to the input $r_{sens}(t)$. This is a common feature of sensitivity-seeking arcs, which significantly complicates the numerical computation of optimal solutions.

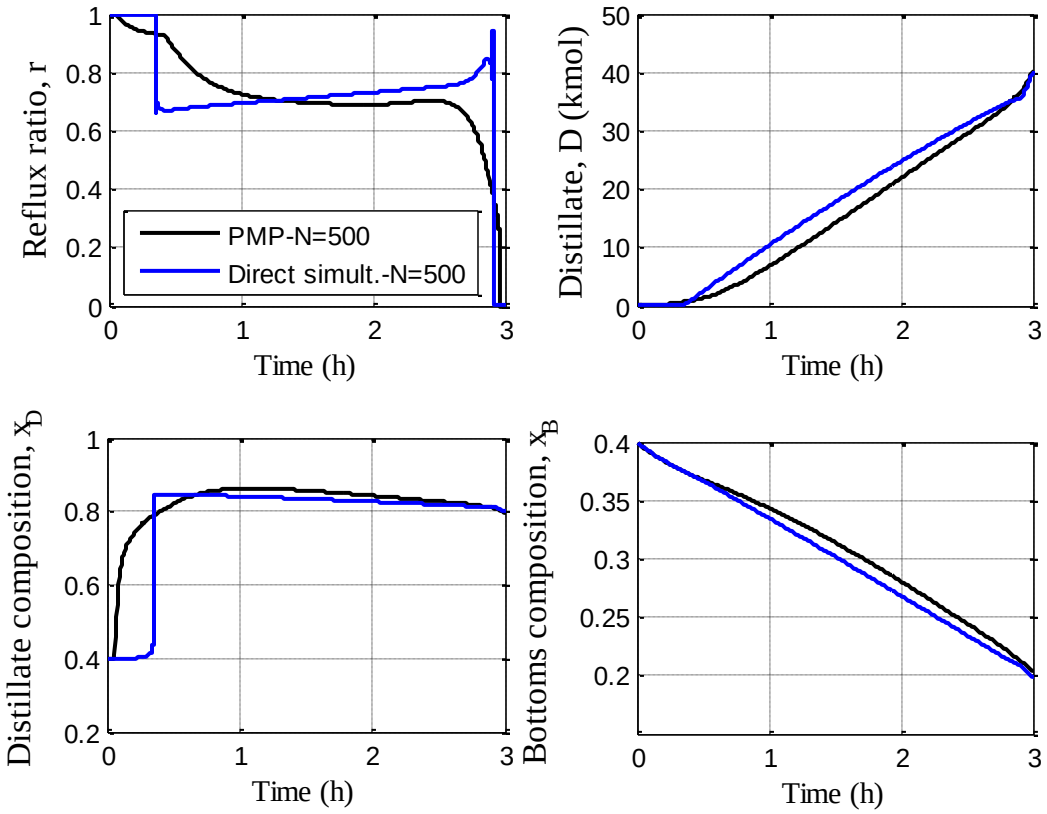

Figure 7. Optimal input and state profiles computed via the PMP-based method and the direct simultaneous strategy for Problem 2.

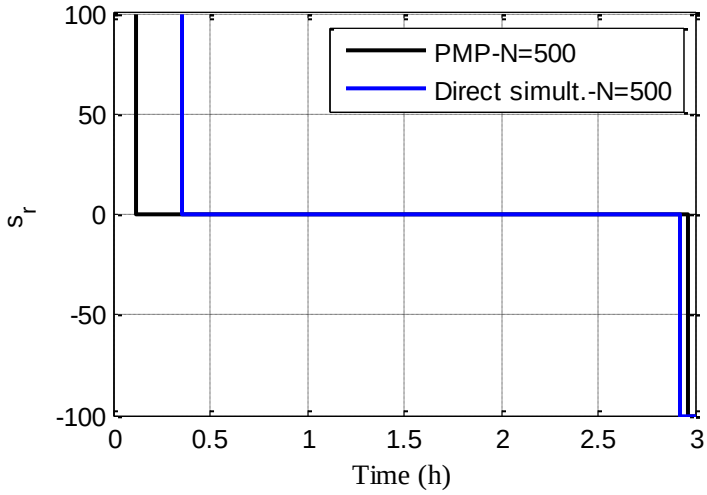

Figure 8. Switching function s_r for Problem 2.

Table 5. Comparison of the indirect PMP-based and direct simultaneous strategies for Problem 2.

Optimization Strategy	**Solution Structure**		
	Arc 1	**Arc 2**	**Arc 3**
Indirect PMP-based N=50, J=40.01 kmol N=500, J=40.02 kmol	r_{max}	r_{sens}	r_{min}
Direct simultaneous N=50, J=40.02 kmol N=500 J=40.03 kmol	r_{max}	r_{sens}	r_{min}

Finally, Fig. 9 compares the computational times needed for the two numerical schemes as functions of the discretization level. One sees that the indirect PMP-based method has a clear advantage when finer grids are applied.

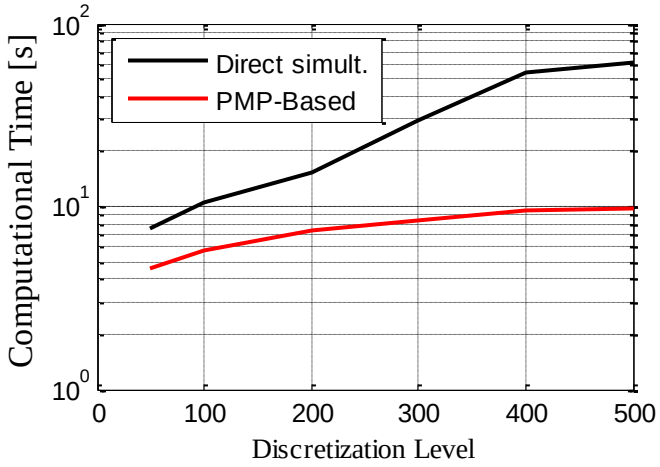

Figure 9. Computational times for different discretization levels of Problem 2.

3.3. PROBLEM - 3: *Fed-batch hydro-formylation reactor with path constraints*

Consider the optimization of a fed-batch reactor to maximize the production of n-tridecanal (nC13al) from 1-dodecene (nC12en) that reacts with syngas ($H_2 + CO$). The reaction network is illustrated in Fig. 10 (Hentschel et al., 2015).

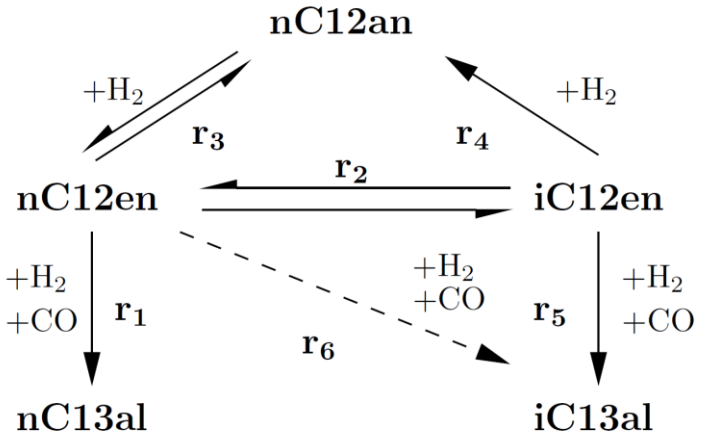

Figure 10. Hydroformulation reaction network.

A stirred tank reactor with gas feeding is used in semi-batch mode of operation. The input variables are the reactor temperature $T(t)$ and the feedrate $u(t)$ of syngas ($H_2 + CO$.) The gas and liquid phases are modeled as ideally mixed phases. The model parameters have been trained and validated using experimental data (Hentschel et al., 2015). The aim is to maximize the concentration of n-tridecanal (nC13al) at a specified final time. In addition to the bounds on the input variables, the total pressure of the gas phase must be kept within specified limits. The dynamic optimization problem is formulated as follows:

$$\max_{u(t),T(t)} J = c_{nc13al}(t_f)$$

s.t.

$$\dot{c}_{liq,i} = {j_i}^{GL} + c_{cat} M_{cat} \sum_{j\in\mathbb{R}} v_{j,i} r_j \; ; \quad c_{liq,i}(0) = c_{liq,i0}; \quad \text{i=1, 2,…,7;}$$

$$\dot{p}_i = \frac{RT}{V_{gas}}\left(u\, x_i - V_{liq}\, {j_i}^{GL}\right) \;\; (i \in gas)\,; \;\; p_i(0) = p_{i0}\,; \;\; x_i \;=\; 0.5\;\left(\frac{\text{mol}}{\text{mol}}\right); \;\; \text{i} = 1, 2;$$

$${j_i}^{GL} = \begin{cases} (k_L a)_i\left({c_i}^* - c_{liq,i}\right), \;\; (if\; i \;\in gas); & \text{i} = 1, 2 \\ 0, \qquad\quad (else); & \text{i} = 3,4, \dots, 7 \end{cases};$$

$$r_1 = \frac{k_{1,0}(T) c_{nC12en} c_{H_2} c_{CO}}{1 + K_{1,1} c_{nC12en} + K_{1,2} c_{nC13al} + K_{1,3} c_{H_2}} \; ;$$

$$r_2 = \frac{k_{2,0}(T)\left(c_{nC12en} - \frac{c_{iC12en}}{K_{p,2}}\right)}{1 + K_{2,1} c_{nC12en} + K_{2,2} c_{iC12en}} \; ;$$

$$r_3 = \frac{k_{3,0}(T)\left(c_{nC12en} c_{H_2} - \frac{c_{nC12an}}{K_{p,3}}\right)}{1 + K_{3,1} c_{nC12en} + K_{3,2} c_{nC13an} + K_{3,3} c_{H_2}} \; ;$$

$$r_4 = k_{4,0}(T) c_{iC12en} c_{H_2} \; ;$$

$$r_5 = k_{5,0}(T) c_{iC12en} c_{H_2} c_{CO} \; ;$$

$$r_6 = k_{6,0}(T) c_{nC12en} c_{H_2} c_{CO} \; ;$$

$$k_j(T) = k_{0,j} \exp\left(-\frac{E_{A,j}}{R}\left(\frac{1}{T} - \frac{1}{T_{ref}}\right)\right) \; ;$$

$$K_{p,j} = \exp\left(\frac{-\Delta G_j}{RT}\right);$$

$$-\Delta G_j = a_{0,j} + a_{1,j} T + a_{2,j} T^2 \; ;$$

$$c_{cat} = \frac{c_{cat,tot}}{1 + K_{cat,1} {c_{CO}}^{K_{cat,3}} + K_{cat,2} \frac{{c_{CO}}^{K_{cat,3}}}{c_{H_2}}} \; ;$$

$${c_i}^* = \frac{p_i}{H_i} \; ;$$

$$H_i = H_i{}^0 \exp\left(\frac{-E_{A,H,i}}{RT}\right);$$

$$p_{total}(t) = p_{H_2}(t) + p_{CO}(t)\ ;$$

$$1\ bar \leq p_{total}(t) \leq 20\ bar\ ;$$

$$0\ \leq u(t)\ ;$$

$$368.15\ \text{K}\ \leq T(t) \leq 388.15\ \text{K} \qquad (8)$$

where i represents the component index (i=1,2,…,7 for the liquid phase and i=1,2 for the gas phase), j stands for the reaction index and R is the reaction set. The final time t_f is fixed at 80 min. All related parameters are given in Table 6. Equal molar content of *CO* and H_2 in the syngas is assumed. The liquid volume V_{liq} and the gas volume V_{gas} inside the reactor are assumed constant, namely 900 mL each. The initial molar amount of the main reactant 1-dodecene is 0.85 mol, while all other initial conditions for the chemical species in the liquid phase are set to zero. The initial partial pressures of the *CO* and H_2 in the gas phase are 10 bar.

Table 6. Model parameters for Problem 3.

Reaction Kinetics						
component	E_A (kJ/mol)	k_0	Unit	K_1 (mL/mol)	K_2 (mL/mol)	K_3 (mL/mol)
r_1	113.08	4.904×10^{16}	$mL^3/(g.min.mol^2)$	574876	3020413	11732838
r_2	136.89	4.878×10^{16}	mL/(g.min)	38632	226214	
r_3	76.11	5.411×10^{8}	$mL^2/(g.min.mol)$	2661.2	7100	1280
r_4	102.26	2.958×10^{4}	$mL^2/(g.min.mol)$			
r_5	120.84	7.619×10^{10}	$mL^3/(g.min.mol^2)$			
r_6	113.08	3.951×10^{10}	$mL^3/(g.min.mol^2)$			
c_{cat}				3.041×10^{4}	0	0.644

Equilibrium Constants			
component	a_0 (kJ/mol)	a_1 (kJ/mol/K)	a_2 (kJ/mol/K^2)
ΔG_2	-11.0034	0	0
ΔG_3	-126.275	0.1266	6.803×10^{-6}

Solubility			
component	H_0 (bar.mL/mol)	$E_{A,H}$(kJ/mol)	$k_L a$ (min^{-1})
H_2	66400	-3.06	9.57
CO	73900	-0.84	7.08

Computed optimal solutions

The parameter values $\alpha = 0.02$, $K = 100$ and $\varepsilon = 0.05$ are used in the PMP-based approach. Due to the lack of sensitivity of the cost function with respect to the fine shape of the input profiles in some of the arcs, a relatively fine input discretization ($N \geq 100$) is necessary for accurate results. The optimal input trajectories and the corresponding concentration of the desired product and total pressure are given in Fig. 11 for N=500. The temperature is initially at the lower bound to favor the desired reaction. With the effect of gas feeding, the concentration of the desired product increases and approaches its maximal value after about 50 min. Then, the temperature is set to its upper limit to suppress the undesired side reactions. This results in a relatively small increase in the concentration of n-tridecanal in the last part of the batch run.

The switching functions s_u and s_T are illustrated in Fig. 12. The solution structure and the performance of both numerical schemes for two discretization levels are given in Table 7.

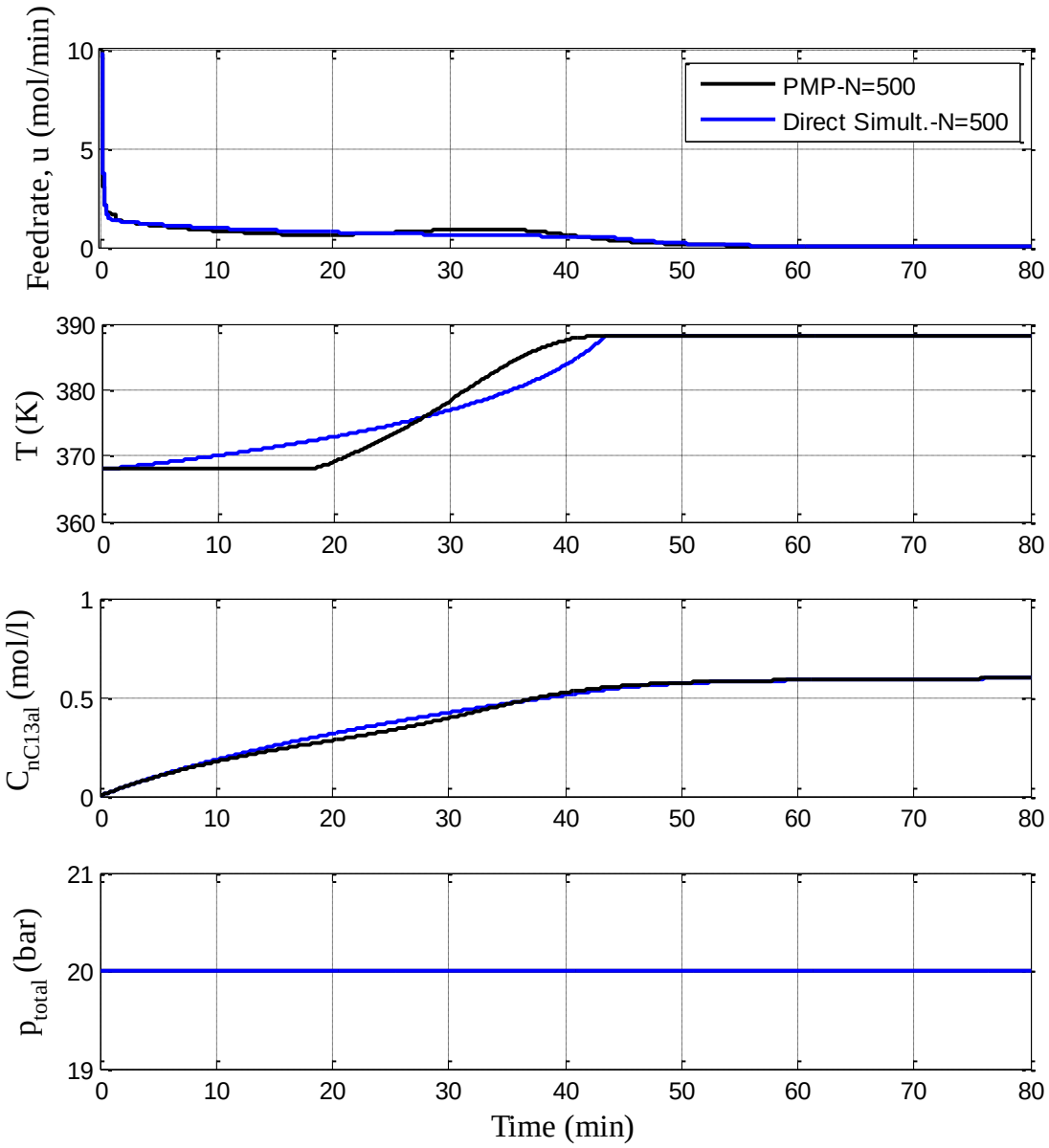

Figure 11. Optimal input and state profiles computed via the PMP-based method and the direct simultaneous strategy for Problem 3.

Table 7. Comparison of the indirect PMP-based and direct simultaneous strategies for Problem 3.

Algorithm	**Solution Structure**		
	Arc 1	**Arc 2**	**Arc 3**
Indirect PMP-based N=100, J=0.593 mol/L N=500, J=0.595 mol/L	u_{path} T_{min}	u_{path} T_{sens}	u_{path} T_{max}
Direct simultaneous N=100, J=0.595 mol/L N=500, J=0.596 mol/L	u_{path} T_{min}	u_{path} T_{sens}	u_{path} T_{max}

A few remarks can be made at this point:

1. Both numerical schemes exhibit a 3-arc solution for different discretization levels as shown in Table 7.
2. The pressure upper bound is active throughout the batch run. This is enforced by adjusting the gas feed rate *u*(t), which is therefore constraint seeking throughout. Toward the end of the batch, the feed rate is very close, but not exactly equal, to zero.
3. There is a significant difference in the two sensitivity-seeking temperature profiles. Again, this is due to the lack of sensitivity of the objective function with respect to the temperature $T_{sens}(t)$.

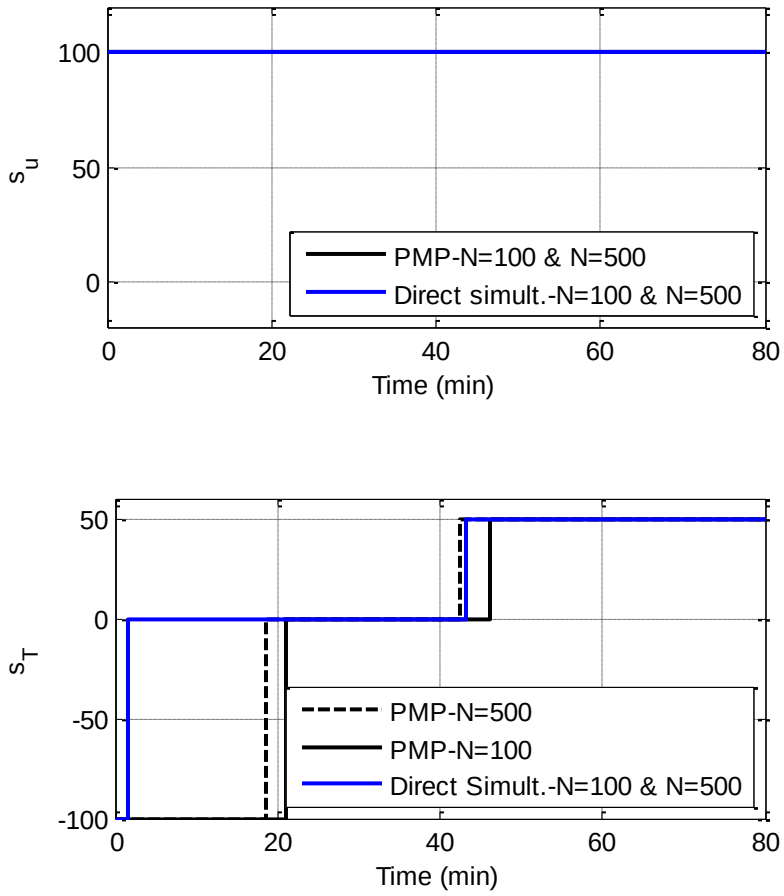

Figure 12. Switching functions s_u and s_T obtained for Problem 3.

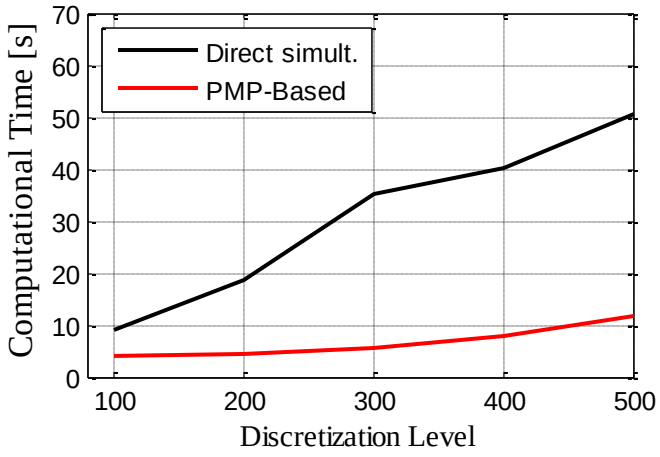

Figure 13. Computational times for different discretization levels of Problem 3.

Fig. 13 shows that the computational time of the PMP-based strategy is significantly shorter than that of the direct simultaneous method.

4. CONCLUSIONS

A PMP-based quasi-Newton algorithm has been proposed for solving constrained dynamic optimization of semi-batch chemical processes. This algorithm constructs the Hamiltonian function by indirectly adjoining the inequality path constraints via their time derivatives so that the inputs u can be easily enforced to fulfill the active path constraints at each iteration step. Symbolic differentiation of the Hamiltonian function with respect to the states is only necessary at the initialization step. The results show that the proposed PMP-based quasi-Newton algorithm can solve the investigated constrained optimization problems significantly faster than direct simultaneous methods as the discretization grid gets finer. Our study also shows that, although there are only negligible differences between the optimal costs determined with various strategies, the actual input profiles can differ significantly and even correspond to different local solutions. The main reason for this observation is the lack of sensitivity of the objective function with respect to the sensitivity-seeking parts of certain inputs. Hence, it may be useful to parameterize these input profiles in a so-called parsimonious way, for example by using switching times and low-order polynomial approximations rather than piecewise-constant or piecewise-linear approximations.

5. ACKNOWLEDGEMENTS

This work was conducted in part in cooperation with the Collaborative Research Centre "Integrated Chemical Processes in Liquid Multiphase Systems". The financial support from the Deutsche Forschungsgemeinschaft (DFG) under the grant SFB/TRR 63 is gratefully acknowledged. Furthermore, the suggestions and support by Jens Bremer on the PMP-Based

method and the direct simultaneous formulation and the help by Andreas Himmel on the CasADi toolbox are gratefully acknowledged.

6. REFERENCES

Ali, U., & Wardi, Y. (2015). Multiple shooting technique for optimal control problems with application to power aware networks. IFAC-PapersOnLine, 48, 286-290.

Andersson, J., & Diehl, M. (2012). Dynamic optimization with CasADi. In 2012 IEEE 51st IEEE Conference on Decision and Control (CDC) (pp. 681-686).

Assassa, F., & Marquardt, W. (2014). Dynamic optimization using adaptive direct multiple shooting. Computers & Chemical Engineering, 60, 242-259.

Biegler, L. T. (2007). An overview of simultaneous strategies for dynamic optimization. Chemical Engineering and Processing: Process Intensification, 46, 1043-1053.

Biegler, L. T. (2010). Nonlinear programming: concepts, algorithms, and applications to chemical processes (Vol. 10): SIAM.

Biegler, L. T., Cervantes, A. M., & Wächter, A. (2002). Advances in simultaneous strategies for dynamic process optimization. Chemical Engineering Science, 57, 575-593.

Binette, J., Srinivasan, B., & Bonvin, D. (2016). On the various local solutions to a two-input dynamic optimization problem. Computers & Chemical Engineering, 95, 71-74.

Bryson, A. E. (1975). Applied optimal control: optimization, estimation and control: CRC Press.

Cannon, M., Liao, W., & Kouvaritakis, B. (2008). Efficient MPC optimization using Pontryagin's minimum principle. International Journal of Robust and Nonlinear Control, 18, 831-844.

Cervantes, A., & Biegler, L. T. (1998). Large-scale DAE optimization using a simultaneous NLP formulation. AIChE Journal, 44, 1038-1050.

Chachuat, B. (2007). Nonlinear and dynamic optimization: from theory to practice. Lecture Notes.

Coward, I. (1967). The time-optimal problem in binary batch distillation. Chemical Engineering Science, 22, 503-516.

Hannemann-Tamás, R., & Marquardt, W. (2012). How to verify optimal controls computed by direct shooting methods? – A tutorial. Journal of Process Control, 22, 494-507.

Hartl, R. F., Sethi, S. P., & Vickson, R. G. (1995). A survey of the maximum principles for optimal control problems with state constraints. SIAM review, 37, 181-218.

Harvey Jr, P. S., Gavin, H. P., & Scruggs, J. T. (2012). Optimal performance of constrained control systems. Smart Materials and Structures, 21, 085001.

Hentschel, B., Kiedorf, G., Gerlach, M., Hamel, C., Seidel-Morgenstern, A., Freund, H., & Sundmacher, K. (2015). Model-based identification and experimental validation of the optimal reaction route for the hydroformylation of 1-dodecene. Industrial & Engineering Chemistry Research, 54, 1755-1765.

Jaspan, R., & Coull, J. (1971). Trajectory optimization techniques in chemical reaction engineering: I. Boundary condition iteration methods. AIChE Journal, 17, 111-115.

Kadam, J., Schlegel, M., Srinivasan, B., Bonvin, D., & Marquardt, W. (2007). Dynamic optimization in the presence of uncertainty: From off-line nominal solution to measurement-based implementation. Journal of Process Control, 17, 389-398.

Kameswaran, S., & Biegler, L. T. (2006). Simultaneous dynamic optimization strategies: Recent advances and challenges. Computers & Chemical Engineering, 30, 1560-1575.

Kim, N., & Rousseau, A. (2012). Sufficient conditions of optimal control based on Pontryagin's minimum principle for use in hybrid electric vehicles. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, 226, 1160-1170.

Logist, F., Houska, B., Diehl, M., & Van Impe, J. F. (2011). Robust multi-objective optimal control of uncertain (bio) chemical processes. Chemical Engineering Science, 66, 4670-4682.

Mayur, D., & Jackson, R. (1971). Time-optimal problems in batch distillation for multicomponent mixtures and for columns with holdup. The Chemical Engineering Journal, 2, 150-163.

Onori, S., Serrao, L., & Rizzoni, G. (2016). Pontryagin's Minimum Principle. In Hybrid Electric Vehicles (pp. 51-63): Springer.

Palanki, S., & Vemuri, J. (2005). Optimal operation of semi-batch processes with a single reaction. International Journal of Chemical Reactor Engineering, 3.

Roubos, J., De Gooijer, C., Van Straten, G., & Van Boxtel, A. (1997). Comparison of optimization methods for fed-batch cultures of hybridoma cells. Bioprocess Engineering, 17, 99-102.

Schlegel, M., & Marquardt, W. (2004). Direct sequential dynamic optimization with automatic switching structure detection. In Proc. of DYCOPS (Vol. 7).

Schlegel, M., & Marquardt, W. (2006a). Adaptive switching structure detection for the solution of dynamic optimization problems. Industrial & Engineering Chemistry Research, 45, 8083-8094.

Schlegel, M., & Marquardt, W. (2006b). Detection and exploitation of the control switching structure in the solution of dynamic optimization problems. Journal of Process Control, 16, 275-290.

Schlegel, M., Stockmann, K., Binder, T., & Marquardt, W. (2005). Dynamic optimization using adaptive control vector parameterization. Computers & Chemical Engineering, 29, 1731-1751.

Srinivasan, B., Biegler, L. T., & Bonvin, D. (2008). Tracking the necessary conditions of optimality with changing set of active constraints using a barrier-penalty function. Computers & Chemical Engineering, 32, 572-579.

Srinivasan, B., & Bonvin, D. (2007). Real-time optimization of batch processes by tracking the Necessary Conditions of Optimality. Industrial & Engineering Chemistry Research, 46, 492-504.

Srinivasan, B., Palanki, S., & Bonvin, D. (2003). Dynamic optimization of batch processes: I. Characterization of the nominal solution. Computers & Chemical Engineering, 27, 1-26.

Vassiliadis, V. S., Sargent, R. W. H., & Pantelides, C. C. (1994). Solution of a class of multistage dynamic optimization problems. 2. Problems with path constraints. Industrial & Engineering Chemistry Research, 33, 2123-2133.

Visser, E., Srinivasan, B., Palanki, S., & Bonvin, D. (2000). A feedback-based implementation scheme for batch process optimization. Journal of Process Control, 10, 399-410.

Wächter, A., & Biegler, L. T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Mathematical programming, 106, 25-57.

Welz, C., Srinivasan, B., Marchetti, A., Bonvin, D., & Ricker, N. (2005). Validation of a solution model for the optimization of a binary batch distillation column. In Proceedings of the 2005, American Control Conference, 2005. (pp. 3127-3132): IEEE.