Computation of Optimal Feed Rates and Operation Intervals for Tubular Reactors [??]

Joachim Birk [a] Michael Liepelt [b] Klaus Schittkowski [b]
Frank Vogel [??]

[a] *BASF AG, Department DWL/LM-L440, D-67056 Ludwigshafen*
[b] *University of Bayreuth, Department of Mathematics, D-95440 Bayreuth*

Abstract

We consider mathematical models for tubular reactors in the form of dynamic distributed parameter systems. The goal is to maximize the overall profit over a fixed time horizon, where the number of cleaning operations, the length of the reactor operation between successive cleanings, and the reactor feed rates for each time interval are to be computed. We assume that product prices and consumer demands are time-dependent. It must be guaranteed that the decrease of the free cross-sectional area of the tube caused by coke deposition never exceeds a certain limit. Moreover, there are time and position dependent constraints for the state and control variables such as a maximum bound for the temperature. The mathematical model and the applied discretization scheme are outlined in detail. Numerical results are presented for a case study, where optimal input feeds and maintenance times of an acetylene reactor are computed. Of special interest is the behaviour of the program under real-time conditions, when changes in the process data or price and user demand functions require a restart of the calculation.

1 Introduction

The computation of optimal feed controls for chemical reactors, especially for tubular reactors, is a well-known technique, see Edgar and Himmelblau [12], Nishida et al. [20], and Buzzi-Ferraris et al. [8,9]. Our intention is to extend the underlying mathematical model structure with the aim to determine also the number of reactor cleanings and the operation length between successive maintenance times under the assumption that time-dependent cost and consumer demand functions are known over the whole time horizon.

The mathematical model is given as a distributed parameter system, strictly speaking in the form of a set of first order partial differential equations in

one space dimension. The chemical reactions and the temperature depend dynamically on the space variable, whereas the dynamic decrease of the cross-sectional area caused by coke deposition is time-dependent. In both cases we know initial values either in the form of time-dependent feed control functions or a constant tube diameter.

Our first objective is to schedule the reactor maintenance, i.e., we want to determine a subdivision into operation intervals, where the reactor is cleaned at the end of each interval. Then we want to compute dynamic feed rates for each interval such that the overall profit of the reactor is maximized. The structure of the input control is quite arbitrary—we allow piecewise constant functions, piecewise linear functions, or piecewise cubic splines.

It must be guaranteed that the free cross-sectional area never falls below a given lower bound, and that the temperature never exceeds a maximum value. As soon as the cross-sectional area reaches this minimum level, the reaction is stopped, the reactor has to be cleaned, and the process is restarted. Moreover, there are bounds for the input controls, mass flows, and operation lengths.

It is essential to understand that the cleaning times are optimization variables, since the price and demand functions may change over time. Changes of these functions are the only causes for differences between the operation intervals, since we always start with the same initial distribution of the cross-sectional area after each reactor cleaning.

This distributed parameter system is discretized w.r.t. the time variable, and the resulting system of ordinary differential equations in the space variable is solved using standard algorithms. In our program we use the routines of Hairer et al. [14] for non-stiff and those of Hairer and Wanner [15] for stiff equations. Stiffness can be introduced by a mixture of fast and slow chemical reactions. This approach is called the method of lines, see Schiesser [23]. Since the cross-sectional area is monotonously decreasing w.r.t. time, we may apply a rather crude approximation, e.g., by Euler's difference formula.

The dynamic constraints are discretized using uniform grids over the space and time interval. Since the input control functions can be described by a finite number of variables, we obtain a finite dimensional and usually large nonlinear programming problem. The resulting optimization problem is solved by the sequential quadratic programming (SQP) code NLPQL of Schittkowski [24,25]. A special advantage of SQP methods is that they can efficiently handle a relatively large number of inequality restrictions. Gradients are approximated by finite differences, where the special sparsity structure of the Jacobian of the problem is exploited.

However, the assumption that time-dependent price and consumer demand functions are a priori known is somewhat unrealistic. In a real-life environment

these functions will permanently change, and even the technical conditions or the chemical process can vary. Thus one has to guarantee that maintenance times and feed rates can be adapted under real-time conditions, i.e., they must be re-evaluated as fast as possible whenever model data are altered. Hence the implemented numerical algorithm allows restarts, where the known solution and additional iteration data of the numerical algorithm are exploited to get a new solution within very few additional iterations.

In Section 2 we give an outline of the considered mathematical model. The discretization of the distributed parameter system, the dynamic constraints, and the control functions is described in Section 3. In Section 4 we give a detailed description of the numerical algorithms that are used for the solution of the problem. In Section 5 we introduce the dynamic equations of an acetylene reactor that serves as a case study. Some numerical results are presented in Section 6 with an special emphasis on stability and real-time properties. Appendix A gives an overview of all variables and constants in this paper and the Appendices B and C contain the normalized parameters of the case study.

2 The Mathematical Model

In this section we consider the chemical reactions in a tubular reactor with length L over a constant time horizon τ. The functions F_1, F_2, and F_3 that are used throughout this section denote suitable model functions depending on the actual reactor type and the chemical reactions to be considered.

The molar concentrations of the r components $C_1, \ldots, C_r$ are described by the differential equations

$$\frac{\partial C}{\partial x}(x,t) = F_1\Big(C(x,t), T(x,t), A(x,t), u(t)\Big) \tag{1}$$

in the space variable x, where $C(x,t) = (C_1(x,t), \ldots, C_r(x,t))^T \in \mathbb{R}^r$, with the initial conditions

$$C_j(0,t) = \frac{\dot{m}_j(0,t)\rho(0,t)}{\dot{m}(t)M_j}, \quad j = 1, \ldots, r_{\mathrm{e}}, \tag{2}$$

and

$$C_j(0,t) = 0, \quad j = r_{\mathrm{e}}+1, \ldots, r, \tag{3}$$

where

$$u_j(t) := \dot{m}_j(0,t), \quad j = 1, \ldots, r_{\mathrm{e}}, \tag{4}$$

are the feeds that are used to control the process, and

$$\dot{m}(t) = \sum_{j=1}^{r_e} \dot{m}_j(0,t) \tag{5}$$

is the total mass flow in the reactor. The corresponding density is given by

$$\rho(x,t) = \sum_{j=1}^{r_e} C_j(x,t) M_j, \tag{6}$$

where M_j is the molar weight of the j-th component. We suppose that for the reactor type we have in mind, the startup of the reaction is so fast that all transient terms can be neglected.

The temperature in the considered reactor is defined by the adiabatic energy balance in its general form,

$$\frac{\partial T}{\partial x}(x,t) = F_2\Big(C(x,t), T(x,t), A(x,t), u(t)\Big), \tag{7}$$

where the initial temperature is assumed to be constant,

$$T(0,t) = T_0, \quad t \in [0,\tau]. \tag{8}$$

It should be pointed out that equations (1) and (7) are based on the justified assumption that material and energy dynamics of the reactor are much faster than the coking mechanism. Therefore all transient accumulation terms are omitted from these equations.

The dynamic decrease of the cross-sectional area of the reactor is described by the differential equation

$$\frac{\partial A}{\partial t}(x,t) = F_3\Big(C(x,t), T(x,t), A(x,t), u(t)\Big) \tag{9}$$

with the initial distribution

$$A(x,t_0) = A_\star(x) \tag{10}$$

at some initial time $t_0 \in [0,\tau]$.

We introduce now additional break points t_c^i for $i = 1, \ldots, n_c$, at which the reaction is stopped and the reactor is cleaned,

$$t_0 < t_c^1 < \cdots < t_c^{n_c} < \tau, \tag{11}$$

and get additional restart values

$$A(x,t_c^i) = A_0, \quad i = 1,\ldots,n_c, \tag{12}$$

for the cross-sectional area of the reactor. We neglect the actual cleaning time, since the dynamic process becomes inactive. In other words, we integrate the reaction equations only within an operation interval $[t_{\mathrm{c}}^i, t_{\mathrm{c}}^{i+1}]$ for $i = 0, \ldots, n_{\mathrm{c}}$ with $t_{\mathrm{c}}^0 := t_0$ and $t_{\mathrm{c}}^{n_{\mathrm{c}}+1} := \tau$.

While the reactor is in operation, it must be possible to change the model or some data and to calculate a new optimal control under real-time conditions, see Jang et al. [17], using the known solution as an initial value. This restart of the process is expressed by an arbitrary initial time $t_0 \in [0, \tau]$ together with an initial distribution of the cross-sectional area $A_\star(x)$ at this time.

If we assume for the time being that the cleaning times $t_{\mathrm{c}} \in \mathbb{R}^{n_{\mathrm{c}}}$ are given, we are looking for input mass feeds $u(t) \in \mathbb{R}^{r_{\mathrm{e}}}$, such that the objective function

$$\begin{aligned} G(C, T, A, u, t_{\mathrm{c}}) = &\int_{t_0}^{t_{\mathrm{c}}^1} \Big(\sum_{j=1}^{r} P_j(t) \dot{m}_j(L,t) - \sum_{j=1}^{r_{\mathrm{e}}} P_j(t) u_j(t) \Big)\, dt \\ &+ \sum_{i=1}^{n_{\mathrm{c}}-1} \int_{t_{\mathrm{c}}^i}^{t_{\mathrm{c}}^{i+1}} \Big(\sum_{j=1}^{r} P_j(t) \dot{m}_j(L,t) - \sum_{j=1}^{r_{\mathrm{e}}} P_j(t) u_j(t) \Big)\, dt \\ &+ \int_{t_{\mathrm{c}}^{n_{\mathrm{c}}}}^{\tau} \Big(\sum_{j=1}^{r} P_j(t) \dot{m}_j(L,t) - \sum_{j=1}^{r_{\mathrm{e}}} P_j(t) u_j(t) \Big)\, dt \\ &- n_{\mathrm{c}} P_{\mathrm{c}} \end{aligned} \tag{13}$$

is maximized, which represents the total profit of the reactor over the whole operation interval $[t_0, \tau]$. We denote by P_{c} the costs for cleaning the reactor, and the real-valued functions P_j for $j = 1, \ldots, r$, are the time-dependent component prices which are assumed to be known in advance.

The resulting optimization problem has a series of constraints. There are lower and upper bounds for the control variables,

$$u_j^{\min}(t) \le u_j(t) \le u_j^{\max}(t), \quad t \in [t_0, \tau], \quad j = 1, \ldots, r_{\mathrm{e}}, \tag{14}$$

there are state constraints for the temperature,

$$T(x,t) \le T^{\max}(t), \quad x \in [0, L], \quad t \in [t_0, \tau] \tag{15}$$

and the reactor yields,

$$\dot{m}_j^{\min}(t) \le \dot{m}_j(L,t) \le \dot{m}_j^{\max}(t), \quad t \in [t_0, \tau], \quad j = 0, \ldots, r. \tag{16}$$

Moreover, we assume that there is a minimum runtime $\Delta t^{\min}$ of the reactor, which leads to the additional restrictions

$$t^{i+1} - t^i \ge \Delta t^{\min}, \quad i = 0, \ldots, n_{\mathrm{c}}. \tag{17}$$

Finally we have to take into consideration the lower bound for the cross-sectional area in the form

$$A(x,t) \geq A^{\min}(t), \quad x \in [0,L], \quad t \in [t_0,\tau]. \tag{18}$$

As soon as the cross-sectional area reaches this minimum level, the reaction is stopped, the reactor has to be cleaned, and the process is restarted.

In summary the optimal control problem consists in minimizing the total profit of the reactor (13) with respect to $u \in U$ and all maintenance times $t_c \in \mathbb{R}^{n_c}$ satisfying the constraints (11) and (17). We denote by U the set of admissible controls, i.e., all piecewise constant functions, piecewise linear functions, or piecewise cubic splines that satisfy the bounds (14). The state variables C, T, and A are given by the system of one-dimensional, first order partial differential equations (2), (7), and (9), with the initial conditions (3), (8), and (10). We note that the mass flow functions $\dot{m}_j(x,t)$ implicitly depend on the control and state variables.

3 Discretization

In this section we describe the discretization scheme used for the numerical solution of the optimal control problem. The main idea is to proceed from a given number of runtime intervals to avoid the formulation of a nonlinear mixed integer optimization problem, and then to adapt this parameter using a simple heuristic. The input control is approximated by functions defined by finitely many parameters and after a suitable discretization of the distributed parameter system with respect to space and time variables we obtain a finite dimensional nonlinear programming problem.

The system equations are given in the form of a distributed parameter system, i.e., a system of first order partial differential equations with spatial dimension one as outlined in the previous section, where the dependencies from time and space variable are explicitly known. A particular feature of the system is that derivatives with respect to the time variable only appear in the expression that describes the change of the cross-sectional area. We assume that the chemical reaction starts so fast that initial transient influences can be neglected. We know furthermore that the cross-sectional area is a monotonously decreasing function with respect to the time variable for any fixed point in the reactor. In other words, the dynamic behaviour of the system in the time variable t is completely different from the behaviour in the space variable x.

Thus we may apply a rather coarse subdivision of the operation intervals,

introducing additional times $t_1^i, \ldots, t_{n_\mathrm{t}}^i$ in each interval $[t_\mathrm{c}^i, t_\mathrm{c}^{i+1}]$, with

$$t_\mathrm{c}^i = t_1^i < t_2^i < \cdots < t_{n_\mathrm{t}}^i = t_\mathrm{c}^{i+1}. \tag{19}$$

Then we discretize equation (9) using a first order Euler scheme,

$$\frac{A(x, t_{k+1}^i) - A(x, t_k^i)}{t_{k+1}^i - t_k^i} = F_3\Big(C(x, t_k^i), T(x, t_k^i), A(x, t_k^i), u(t_k^i)\Big) \tag{20}$$

for $k = 1, \ldots, n_\mathrm{t} - 1$, starting from

$$A(x, t_1^i) := \begin{cases} A_\star(x), & \text{if } i = 0, \\ A_0, & \text{otherwise,} \end{cases} \tag{21}$$

and get a system of ordinary differential equations

$$\frac{\partial C}{\partial x}(x, t_k) = F_1\Big(C(x, t_k^i), T(x, t_k^i), A(x, t_k^i), u(t_k^i)\Big), \tag{22}$$

$$\frac{\partial T}{\partial x}(x, t_k^i) = F_2\Big(C(x, t_k^i), T(x, t_k^i), A(x, t_k^i), u(t_k^i)\Big), \tag{23}$$

with initial values

$$C_j(0, t_k^i) = \frac{u_j(t_k^i)\rho(0, t_k^i)}{\dot{m}(t_k^i) M_j}, \quad j = 1, \ldots, r_\mathrm{e}, \tag{24}$$

$$C_j(0, t_k^i) = 0, \quad j = r_\mathrm{e} + 1, \ldots, r, \tag{25}$$

$$T(0, t_k^i) = T_0. \tag{26}$$

As already mentioned, the control functions are approximated by piecewise constant functions, piecewise linear functions, or piecewise cubic splines. In order to simplify the notation and to avoid additional interpolation, we suppose that the grid points that are used for these approximations are identical to those that are used for the discretization of the cross-sectional area, i.e., $t_1^i, \ldots, t_{n_\mathrm{t}}^i$, for $i = 0, \ldots, n_\mathrm{c}$. For convenience all discrete control values that uniquely define the approximated control function are combined in one vector u. Together with the cleaning times they comprise the $r_\mathrm{e}(n_\mathrm{c} + 1)n_\mathrm{t} + n_\mathrm{c}$ optimization variables in the discretized problem.

In order to discretize the objective function and the constraints, we also need a discretization of the space interval, e.g., in the form

$$0 = x_1 < x_2 < \cdots < x_l = L,$$

where L denotes the total reactor length. Based on this space discretization and the known time discretization (19), we get the following finite dimensional approximations for the dynamic control and state constraints:

(1) Lower and upper bounds for the control:

$$u^{\min} \le u \le u^{\max} \tag{27}$$

with constant vectors $u^{\min}$ and $u^{\max}$ of the same dimension as u.

(2) Upper bounds for the temperature:

$$\begin{aligned} T(x_h, t_k^i) \le T^{\max}(t_k^i), \quad & h = 1, \ldots, l, \quad i = 0, \ldots, n_{\mathrm{c}}, \\ & k = 1, \ldots, n_{\mathrm{t}}. \end{aligned} \tag{28}$$

(3) Lower and upper bounds for the reactor yield:

$$\begin{aligned} \dot{m}_h^{\min}(t_k^i) \le \dot{m}_h(L, t_k^i), \quad & h = 1, \ldots, r, \quad i = 0, \ldots, n_{\mathrm{c}}, \\ & k = 1, \ldots, n_{\mathrm{t}}, \end{aligned} \tag{29}$$

$$\begin{aligned} \dot{m}_h(L, t_k^i) \le \dot{m}_h^{\max}(t_k^i), \quad & h = 1, \ldots, r, \quad i = 0, \ldots, n_{\mathrm{c}}, \\ & k = 1, \ldots, n_{\mathrm{t}}. \end{aligned} \tag{30}$$

(4) Lower bounds for the cross-sectional area:

$$\begin{aligned} A^{\min} \le A(x_h, t_k^i), \quad & h = 1, \ldots, l, \quad i = 0, \ldots, n_{\mathrm{c}}, \\ & k = 1, \ldots, n_{\mathrm{t}}. \end{aligned} \tag{31}$$

(5) Lower and upper bounds for the cleaning times:

$$t_0 \le t_{\mathrm{c}}^1 < \cdots < t_{\mathrm{c}}^{n_{\mathrm{c}}} \le \tau, \tag{32}$$

$$\Delta t^{\min} \le t_{\mathrm{c}}^{i+1} - t_{\mathrm{c}}^i, \quad i = 1, \ldots, n_{\mathrm{c}} - 1. \tag{33}$$

It should be noted that we do not expect singular arcs for this class of reactor models, i.e., any time intervals, where a constraint remains active at its bound.

The approximation of the objective function is obtained by numerical integration using the trapezoidal rule at the known time break points. Finally we obtain a nonlinear programming problem with $r_{\mathrm{e}}(n_{\mathrm{c}}+1)n_{\mathrm{t}} + n_{\mathrm{c}}$ variables and $2l(n_{\mathrm{c}}+1)n_{\mathrm{t}} + 2r(n_{\mathrm{c}}+1)n_{\mathrm{t}} + 1 + (n_{\mathrm{c}}-1)$ inequality constraints, if we neglect simple bounds for the optimization variables. The underlying dynamic system we have to integrate for each function evaluation, consists of $(r+2)(n_{\mathrm{c}}+1)n_{\mathrm{t}}$ ordinary differential equations.

4 Numerical Implementation

Our goal is to achieve a highly modular and flexible numerical implementation in Fortran 77. First we have to solve large and perhaps stiff systems of ordinary differential equations. Since we do not a priori know whether the system is stiff or not and which algorithm could be the most efficient one in a special situation, we allow the user to select one from six available codes:

(1) DOPRI5: Explicit Runge-Kutta code based on the method of Dormand and Prince [11] of order 4(5) with step size control (Hairer et al. [14]).
(2) DOP853: Explicit Runge-Kutta code based on the method of Dormand and Prince [11] of order 8(5,3) with step size control (Hairer et al. [14]).
(3) ODEX: Extrapolation code based on the GBS algorithm with variable order and variable step sizes (Hairer et al. [14]).
(4) RADAU5: Implicit Runge-Kutta method (Radau IIa) of order 5 with step size control (Hairer and Wanner [15]).
(5) SDIRK4: Diagonally implicit Runge-Kutta method of order 4 with step size control (Hairer and Wanner [15]).
(6) SEULEX: Extrapolation algorithm based on the linearly implicit Euler method with step size control and order selection (Hairer and Wanner [15]).

As outlined in the previous section, the optimal control problem is transformed into a finite dimensional parameter optimization problem. For the numerical solution of the resulting nonlinear problem we apply the sequential quadratic programming (SQP) algorithm NLPQL of Schittkowski [25]. A particular advantage of these methods is that they can handle a large number of inequality constraints without severe increase of the number of iterations or computing time.

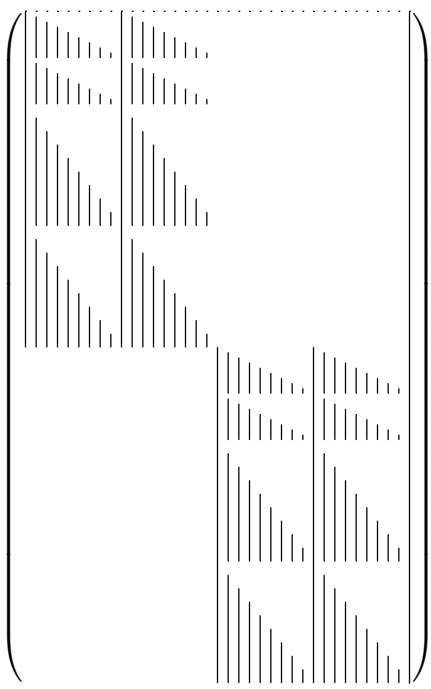
The numerical solution of the optimal control problem requires that a fixed number of cleaning times is given. We start from an arbitrary number n_c specified by the user, and increase or decrease n_c until the total profit (13) cannot be further increased. It can be assumed that the user has sufficient experience and knowledge about the reactor, so that the initial guess is not too far away from the optimal value. The initial values of these cleaning times are chosen equidistant in the whole time interval and then they are adapted during the optimization process.

Gradients of objective function and constraints are evaluated by the forward scaled difference formula. However, the special structure of the Jacobian of constraints is exploited based on the observation that the state variables in different time intervals are linked only by the cleaning times. Moreover the time and space discretization procedures lead to lower triangular sub-matrices for the sensitivities of the control parameters. Figure 1 shows the non-zero structure of the Jacobian for two time intervals, i.e., for $n_c = 1$, and two input control functions.

There is no necessity to exploit this special structure within the SQP algorithm NLPQL. The sparsity pattern belongs to inequality constraints, that are passed to an algorithm for solving certain quadratic programming problems obtained by a quadratic approximation of the Lagrangian function and a linearization of the constraints. However, the full Jacobian of all constraints is evaluated

only at the starting point. Afterwards only gradients belonging to the very few active constraints are re-evaluated. Moreover the quadratic programming problem is solved by the dual method of Goldfarb and Idnani [13], where only active inequality constraints are added to a so-called working set, i.e., very few in our case.

Figure 1. Structure of the Jacobian



It should be noted that there exist at least three important alternative approaches for the solution of optimal control problems.

Multiple shooting proceeds from the maximum principle of Pontryagin and solves two-point boundary value problems representing the necessary optimality conditions, see Bulirsch [7] or Maurer and Gillessen [19] for a theoretical treatment, or Oberle and Grimm [21] for a numerical implementation.

Another possible approach is based on dynamic programming, see Dadebo and McAuley [10], Luus [18], or Vassiliadis et al. [28], and is frequently applied to chemical engineering problems.

Reduced SQP methods use polynomial approximations of the state variables to replace the differential equations by algebraic equations, see Biegler [3] for a reference. This leads to a dramatic increase in the number of optimization variables and nonlinear equality constraints and one has to take advantage of the special structure when solving the large scale nonlinear programming problem by adapted SQP or related methods.

We prefer, however, our direct approach on account of several reasons. First we have to solve distributed parameter systems leading to a large set of ordinary differential equations depending on the desired accuracy of the time discretization and on the number of reaction equations. Large industrial tubular reactors are described by thousands of equations even in the steady-state case, and the additional time discretization would dramatically increase the size of dynamic system. To give an example, the case study in the next section describes the reactions in an acetylene reactor and consists of only 8 reaction equations, but leads to a system of 400 differential equations for 10 grid points and 5 operation intervals.

5 A Case Study

We consider a real reactor producing acetylene (C_2H_2), reacting the methane (CH_4) in natural gas with oxygen. This reaction requires less oxygen compared to complete combustion. The products are quickly quenched to keep the acetylene from being converted entirely to coke, see Wansbrough [29].

During the reaction process a small part of the carbon is deposited in the reactor as coke. The quantity and its distribution in the reactor depend on the reaction equations. Since it is impossible to measure the cross-sectional area directly, we need a mathematical model that describes the functional dependence of the cross-sectional area from other system parameters. If the deposition of coke reaches a certain limit, the reactor must be stopped and the tube is cleaned.

We consider now six chemical reactions. Reactions 1 through 5 are the main ones producing acetylene, but also undesirable byproducts such as coke. Reaction 6 is included only to balance the hydrogen stoichiometry.

$$
\begin{aligned}
&1. \quad CH_4 \rightarrow \tfrac{1}{2}C_2H_2 + \tfrac{3}{2}H_2 \\
&2. \quad CH_4 + O_2 \rightarrow CO + H_2O + H_2 \\
&3. \quad CO + \tfrac{1}{2}O_2 \rightarrow CO_2 \\
&4. \quad C_2H_2 \rightarrow 2C + H_2 \\
&5. \quad H_2 + \tfrac{1}{2}O_2 \rightarrow H_2O \\
&6. \quad C + \tfrac{n}{2}H_2 \rightarrow CH_n
\end{aligned}
\tag{34}
$$

All components in the reaction scheme of the acetylene reactor are numbered in the following way:

$$\begin{aligned}
&\text{Component 1:} \quad \mathrm{CH_4} \\
&\text{Component 2:} \quad \mathrm{O_2} \\
&\text{Component 3:} \quad \mathrm{C_2H_2} \\
&\text{Component 4:} \quad \mathrm{CO_2} \\
&\text{Component 5:} \quad \mathrm{H_2} \\
&\text{Component 6:} \quad \mathrm{CO} \\
&\text{Component 7:} \quad \mathrm{H_2O} \\
&\text{Component 8:} \quad \mathrm{CH}_n
\end{aligned} \tag{35}$$

The reactions can be described by the following system of ordinary differential equations, where C_i denotes the molar concentration of the i-th component,

$$\begin{aligned}
\tfrac{\partial}{\partial x} C_1(x,t) &= (-r_1(x,t) - r_2(x,t))/v(x,t), \\
\tfrac{\partial}{\partial x} C_2(x,t) &= (-r_2(x,t) - \tfrac{1}{2} r_3(x,t) - \tfrac{1}{2} r_5(x,t))/v(x,t), \\
\tfrac{\partial}{\partial x} C_3(x,t) &= (\tfrac{1}{2} r_1(x,t) - r_4(x,t))/v(x,t), \\
\tfrac{\partial}{\partial x} C_4(x,t) &= r_3(x,t)/v(x,t), \\
\tfrac{\partial}{\partial x} C_5(x,t) &= (\tfrac{3}{2} r_1(x,t) + r_2(x,t) + r_4(x,t) \\
&\qquad - r_5(x,t) - n(1-\varepsilon) r_4(x,t))/v(x,t), \\
\tfrac{\partial}{\partial x} C_6(x,t) &= (r_2(x,t) - r_3(x,t))/v(x,t), \\
\tfrac{\partial}{\partial x} C_7(x,t) &= (r_2(x,t) + r_5(x,t))/v(x,t), \\
\tfrac{\partial}{\partial x} C_8(x,t) &= 2(1-\varepsilon) r_4(x,t)/v(x,t),
\end{aligned} \tag{36}$$

with initial conditions

$$\begin{aligned}
C_1(0,t) &= C_1^0(t), \\
C_2(0,t) &= C_2^0(t), \\
C_i(0,t) &= 0, \quad i = 3, \ldots, 8,
\end{aligned} \tag{37}$$

and some reaction parameter ε.

The five reaction rates r_i contain some constant parameters, i.e., reaction constants k_1, ..., k_5, activation energies E_1, ..., E_5, and reaction orders a_1, a_2, and a_4. For the smaller and less important reactions, the stoichiometric order can be used as an estimate for this reaction order. The average reaction temperature T_r is used to scale the exponential functions and R is the gas constant.

Using these parameters we get the following expressions,

$$\begin{aligned}
r_1(x,t) &= k_1 \exp\Big(-\tfrac{E_1}{R}\Big(\tfrac{1}{T(x,t)} - \tfrac{1}{T_\mathrm{r}}\Big)\Big) C_1^{a_1}(x,t),\\
r_2(x,t) &= k_2 \exp\Big(-\tfrac{E_2}{R}\Big(\tfrac{1}{T(x,t)} - \tfrac{1}{T_\mathrm{r}}\Big)\Big) C_1(x,t) C_2^{a_2}(x,t),\\
r_3(x,t) &= k_3 \exp\Big(-\tfrac{E_3}{R}\Big(\tfrac{1}{T(x,t)} - \tfrac{1}{T_\mathrm{r}}\Big)\Big) C_6(x,t) C_2^{0.5}(x,t),\\
r_4(x,t) &= k_4 \exp\Big(-\tfrac{E_4}{R}\Big(\tfrac{1}{T(x,t)} - \tfrac{1}{T_\mathrm{r}}\Big)\Big) C_3^{a_4}(x,t),\\
r_5(x,t) &= k_5 \exp\Big(-\tfrac{E_5}{R}\Big(\tfrac{1}{T(x,t)} - \tfrac{1}{T_\mathrm{r}}\Big)\Big) C_5(x,t) C_2^{0.5}(x,t).
\end{aligned} \tag{38}$$

It is assumed that the total mass flow

$$\dot{m}(t) = \dot{m}_1(0,t) + \dot{m}_2(0,t), \tag{39}$$

is constant during the reaction. The velocity of the mixture in the reactor depends on the cross-sectional area $A(x,t)$ and is given by

$$v(x,t) = \frac{\dot{m}(t)}{\rho(x,t) A(x,t)}. \tag{40}$$

The reactor is controlled by the initial feeds of natural gas and oxygen,

$$\dot{m}_1(0,t), \qquad \dot{m}_2(0,t), \tag{41}$$

from which we derive the initial molar concentrations

$$\begin{aligned}
C_1^0(t) &= \frac{\dot{m}_1(0,t)\rho(0,t)}{\dot{m}(t) M_1},\\
C_2^0(t) &= \frac{\dot{m}_2(0,t)\rho(0,t)}{\dot{m}(t) M_2}.
\end{aligned} \tag{42}$$

The incremental change in temperature is determined by the rate of heat release for all reactions

$$\frac{\partial}{\partial x} T(x,t) = \frac{1}{\rho(x,t) v(x,t) c_\mathrm{p}(x,t)} \sum_{i=1}^{5} r_i(x,t) \Delta H_i \tag{43}$$

with initial condition $T(0,t) = T_0$. It depends on the density

$$\rho(x,t) = \sum_{j=1}^{8} C_j(x,t) M_j \tag{44}$$

and the total heat capacity

$$c_\mathrm{p}(x,t) = \frac{\sum_{j=1}^{8} c_{\mathrm{p}j} M_j C_j(x,t)}{\sum_{j=1}^{8} M_j C_j(x,t)}. \tag{45}$$

The individual heat capacities $c_{\mathrm{p}j}$ are considered to be constant, and the parameters ΔH_i denote the known heats of reaction.

The mass flow of each component can be computed from the solution of the system (36) and (37), i.e.,

$$\dot{m}_j(x,t) = \dot{m}(t)\frac{C_j(x,t)M_j}{\sum_{i=1}^{8} C_i(x,t)M_i}. \tag{46}$$

The deposit of coke is modeled by the time dependent differential equation

$$\frac{\partial}{\partial t}A(x,t) = -\beta r_4(x,t) \tag{47}$$

with initial condition $A(x,0) = A_0$ and some reaction parameter $\beta \in \mathbb{R}$.

6 Numerical Results

In this section we present some numerical results that were obtained for the acetylene reactor described in the previous section. The normalized reaction parameters for this case study are derived from a real reactor at BASF and are listed in Appendix A.

All numerical results were obtained on a PC with an Intel PentiumPro 200 processor and 96 MByte RAM, running Microsoft Windows NT 4.0. The discretized system of ordinary differential equations was integrated using the explicit Runge-Kutta method of Dormand and Prince, i.e., the code DOPRI5 of Hairer et al. [14]. It turns out that a quite crude discretization yields acceptable results and that the ordinary differential equations are not stiff in the present case study.

6.1 Simulation for constant feeds

To give a first impression of the the reactions in this model, we start with a simulation using constant input feeds of oxygen and methane, i.e.,

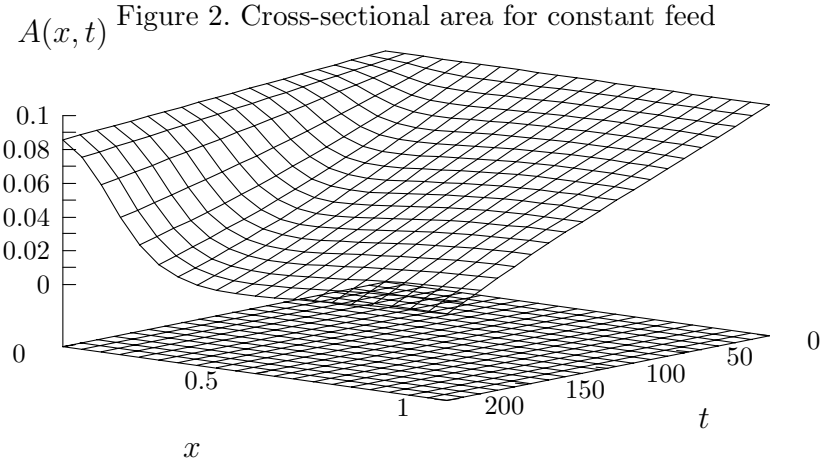
$$\dot{m}_1(0,t) = \dot{m}_2(0,t) = 500, \quad t \in [0,\tau], \tag{48}$$

where we assume that the data of Table 1 are given. The corresponding surface plot of the cross-sectional area is shown in Figure 2. We observe a drastic violation of the minimum admissible cross sectional area of the tube $A^{\min}$.

Table 1. Process parameters

name	*value*	*description*
A_0	0.1	cross-sectional area of the tube
$A^{\min}$	0.08	minimum cross-sectional area
L	1	length of the reactor
T_0	873.15	initial reaction temperature
$T^{\max}$	1300	maximum permissible temperature
τ	200	duration of the process
$\Delta t^{\min}$	60	minimum runtime of the reactor
P_{c}	5000	costs of a reactor cleaning
l	21	number of space discretizations

Figure 2. Cross-sectional area for constant feed



6.2 Calculation of the optimal feeds

Now we want to compute the optimal control of the acetylene reactor, i.e., input feeds that yield feasible values of the state and control variables and maximize the overall profit of the process. Throughout this section we use piecewise linear approximations for the feeds of O_2 and CH_4.

For the start of the optimization algorithm we chose constant values of the control functions, $\dot{m}_1(0,t) = 400$ and $\dot{m}_2(0,t) = 300$. The expected time-dependent price and demand data are given in the form of piecewise linear approximations, see Appendix C. Moreover, we assume that the following

bounds for the feed controls are given:

$$200 \le \dot{m}_1(0,t) \le 800, \quad t \in [0,\tau],$$
$$200 \le \dot{m}_2(0,t) \le 800, \quad t \in [0,\tau].$$

In the first series of tests we want to show the influence of a different number of operation periods. Thus we compute the optimal control for zero to five maintenance breaks, i.e., for $n_c = 0$ to $n_c = 5$. To be able to compare some performance data, we vary the number of time grid points to get discretized optimization problems of approximately the same size, i.e., we require that $(n_c + 1)n_t \approx 24$. Table 2 shows the sizes of the resulting discretized nonlinear programming problems.

Table 2. Size of discretized NLP problems

number of cleaning periods (n_c)	0	1	2	3	4	5
number of time grid points (n)	24	12	8	6	5	4
number of optimization parameters	48	49	50	51	54	53
number of ODEs	216	216	216	216	225	216
number of constraints	1392	1393	1394	1395	1454	1397

The termination accuracy for the nonlinear programming code NLPQL is set to 10^{-8}, and the relative error of the ODE solver DOPRI5 is set to 10^{-6}. Since the starting trajectories are poor and the objective function is very flat in a neighbourhood of an optimal solution, NLPQL requires a large number of iterations to reach a solution. The number of iterations, the obtained objective function values with and without cleaning costs, and some other convergence results are listed in Table 3, using the following abbreviations:

- f – objective function value without cleaning costs
- $f^\star$ – objective function value with cleaning costs
- it – number of iterations
- cv – constraint violation
- ac – number of active constraints
- oc – optimality criterion

For a precise definition of these terms and an description of the applied algorithm see Schittkowski [24,25]. The corresponding control functions for the O_2 feed are shown in Figures 3 to 8. A typical surface plot of the cross-sectional area for two operation intervals is shown in Figure 9. The optimal solution for

the CH_4 feed attains its upper bound in all test runs, i.e., $\dot{m}_1(0,t) = 800$ for $t \in [0,\tau]$.

Table 3. Performance results for different operation intervals

n_c	f	$f^\star$	it	cv	ac	oc
0	39634	39634	292	$0.12 \cdot 10^{-11}$	3	$0.31 \cdot 10^{-8}$
1	52235	47235	60	$0.91 \cdot 10^{-12}$	4	$0.40 \cdot 10^{-9}$
2	58095	48095	101	$0.45 \cdot 10^{-12}$	6	$0.29 \cdot 10^{-8}$
3	60405	45405	129	$0.11 \cdot 10^{-11}$	8	$0.78 \cdot 10^{-8}$
4	60747	40747	335	$0.34 \cdot 10^{-9}$	12	$0.98 \cdot 10^{-8}$
5	60931	35931	279	$0.55 \cdot 10^{-9}$	15	$0.19 \cdot 10^{-8}$

The overall profit increases with the number of reactor cleanings if the corresponding costs are neglected, see the second column of Table 3. So the optimal distribution of the operating times depends on the cleaning costs. If we assume costs of 5000 units as in our case, we maximize the profit for 2 breaks. It is interesting to note that the SQP algorithm NLPQL reaches a quite accurate solution despite of the numerical errors in the gradient approximation.

Figure 3. Optimal O_2 feed for one operation period

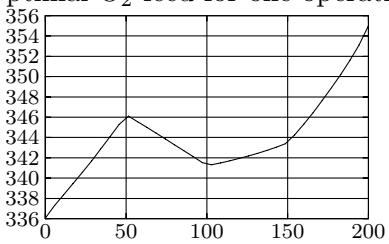


Figure 4. Optimal O_2 feed for two operation periods

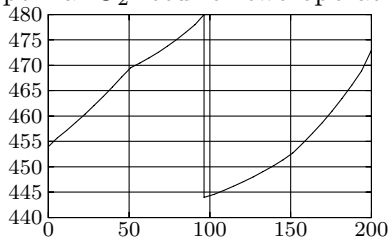


Figure 5. Optimal O_2 feed for three operation periods

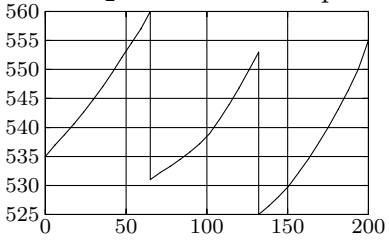


Figure 6. Optimal O_2 feed for four operation periods

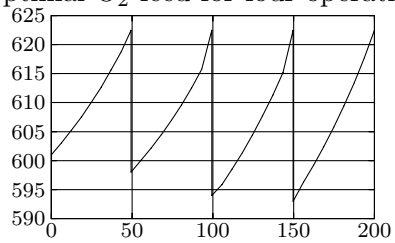


Figure 7. Optimal O_2 feed for five operation periods

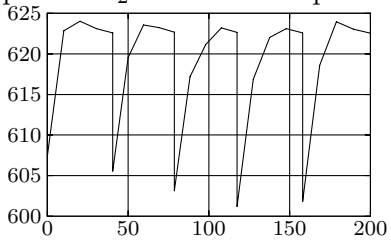


Figure 8. Optimal O_2 feed for six operation periods

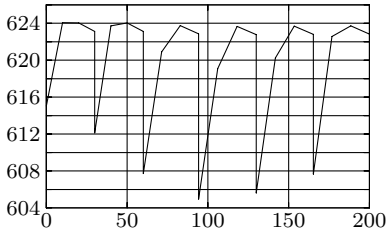
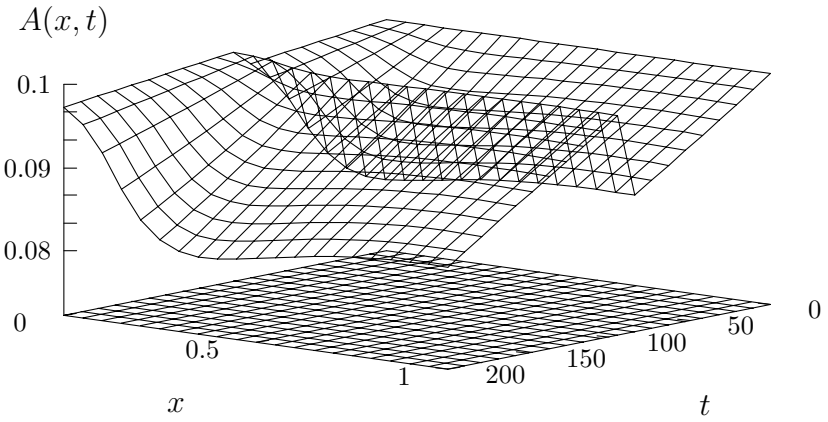


Figure 9. Cross-sectional area for optimal feed



6.3 Stability of the time discretization

Our next aim is to examine the stability of the various dicretizations that are used in our approach. First we want to test the accuracy of the time approximation. We proceed from three operation periods, i.e., $n_c = 2$, and

compute optimal solutions for the control variables with 4, 8, 12, and 16 time grid points and the same number of grid points for the time discretization of the dynamic system n.

The results are listed in Table 4. Objective function values decrease when improving time discretization, as can be expected. The control function is adapted more accurately and the finer constraint approximations lead to smaller feasible domains, thus increasing the objective function. Also the number of iterations is increasing, since we get more and more complex optimization problems.

Table 4. Performance results for different control approximations

n	$f^\star$	it	cv	ac	oc
4	48494	86	$0.37 \cdot 10^{-13}$	6	$0.62 \cdot 10^{-8}$
8	48095	101	$0.45 \cdot 10^{-12}$	6	$0.29 \cdot 10^{-8}$
12	47959	106	$0.83 \cdot 10^{-10}$	6	$0.76 \cdot 10^{-8}$
16	47891	117	$0.62 \cdot 10^{-10}$	6	$0.23 \cdot 10^{-8}$
20	47810	137	$0.22 \cdot 10^{-14}$	6	$0.47 \cdot 10^{-8}$

6.4 Stability of the space discretization

In the subsequent tests we want to test the accuracy of the space discretization. Since the ordinary differential equations are formulated w.r.t. the space variable x, we get very accurate solutions for the underlying dynamic system influenced only by the time discretization. On the other hand we discretize the constraints at equidistant grid points, and the subsequent results show the sensitivity of a solution for different grid sizes l. Again we proceed from three maintenance intervals and eight equidistant grid points for the time variable.

The results are shown in Table 5. Obviously the discretization accuracy influences the final objective function significantly. The finer the grid, the more accurate are the dynamic constraints. The feasible domain shrinks somewhat leading to increasing cost function values.

6.5 Real-time optimization (restarts)

In the preceding tests the optimization algorithm NLPQL required a large number of iterations to reach a solution. This can be explained by the fact that the initial guesses for the optimization variables are too far away from

Table 5. Performance results for different spatial grid sizes

l	$f^\star$	it	cv	ac	oc
5	48091	110	$0.59 \cdot 10^{-10}$	6	$0.16 \cdot 10^{-8}$
10	48096	110	$0.65 \cdot 10^{-10}$	6	$0.21 \cdot 10^{-8}$
20	48095	101	$0.45 \cdot 10^{-12}$	6	$0.29 \cdot 10^{-8}$
40	48095	101	$0.91 \cdot 10^{-12}$	6	$0.31 \cdot 10^{-8}$

the optimal solution. In a real-life environment, however, we may assume that there exists an optimal solution of a running process, and that we need to recompute another one after some minor changes of the model data, e.g., prices or consumer demands. Thus we want to simulate now restarts under real-time assumptions.

Starting from a given optimal control of the tubular reactor with three intervals, we make some small changes of the problem configuration to demonstrate the possibility of computing a new optimal feed control under real-time conditions. For this purpose we use the solution of the original problem, the approximations of the Lagrange multipliers and the BFGS matrix for a restart. The results of these tests are listed in Table 6.

It turns out that the alteration of active or nearly active bounds for the cross sectional area and the temperature leads to some significant alterations in the profit function, in particular the increase of the admissible tube diameter by 10 % leads to a profit loss of about 20 %. An increase of the CH_4 prices leads also to a decrease of the objective function, in contrast to an increase of the C_2H_2 prices.

Table 6. Real-time simulations

changes	it	$f^\star$
set $T^{\max} = 1287$ (−1 %)	14	48089
set $A^{\min} = 0.0808$ (+1 %)	18	47671
set $A^{\min} = 0.088$ (+10 %)	50	41274
increase price of CH_4 by 0.02	27	46369
increase price of C_2H_2 by 0.02	17	56993

7 Conclusions

We extended an optimal control model for tubular reactors with the aim to take into account also the number and position of cleaning times. The partial differential equations are discretized using the method of lines, where the resulting system of ordinary differential equations can be integrated by any standard ODE solver. Since control functions are represented by a finite number of variables and since dynamic constraints are discretized at given time and space grid points, we get a finite dimensional nonlinear programming problem, for which a standard SQP method is applied. Proceeding from a case study in the form of an existing acetylene reactor, is is shown that operating intervals and optimal feed controls can be simultaneously computed. By some further numerical tests we show the stability of the proposed approach and its behaviour under real-time conditions. This will be an essential assumption for several industrial applications of the proposed method.

References

[1] J. S. Albuquerque and L. T. Biegler. Decomposition algorithms for on-line estimation with nonlinear models. *Computers Chem. Engineering*, 19:1031–1039, 1995.

[2] M. Baltes, R. Schneider, C. Sturm, and M. Reuss. Optimal experimental design for parameter estimation in unstructured growth models. *Biotechnol. Prog.*, 10:480–488, 1994.

[3] L. T. Biegler. On the simultaneous solution and optimization of large scale engineering systems. *Computers Chem. Engineering*, 12:357–369, 1988.

[4] L. T. Biegler. Chemical process simulation. *Chemical Engineering Progress*, 85:50–61, 1989.

[5] L. T. Biegler, J. J. Damiano, and G. E. Blau. Nonlinear parameter estimation: A case study comparison. *AIChE Journal*, 32:29–45, 1986.

[6] P. Bilardello, X. Joulia, J. M. Le Lann, H. Delmas, and B. Koehret. A general strategy for parameter estimation in differential algebraic systems. *Computers Chem. Engineering*, 17:517–525, 1993.

[7] R. Bulirsch. Die Mehrzielmethode zur numerischen Lösung von nichtlinearen Randwertproblemen und Aufgaben der optimalen Steuerung. Technical report, Carl-Cranz-Gesellschaft e.V., Oberpfaffenhofen, 1971.

[8] G. Buzzi-Ferraris, G. Facchi, P. Forzetti, and E. Troncani. Control optimization of tubular catalytic decay. *Industrial Engineering in Chemistry*, 23:126–131, 1984.

[9] G. Buzzi-Ferraris, M. Morbidelli, P. Forzetti, and S. Carra. Deactivation of catalyst – mathematical models for the control and optimization of reactors. *International Chemical Engineering*, 24:441–451, 1984.

[10] S. A. Dadebo and K. B. McAuley. Dynamic optimization of constrained chemical engineering problems using dynamic programming. *Computers Chem. Engineering*, 5:513–525, 1995.

[11] J. R. Dormand and P. J. Prince. High order embedded Runge-Kutta formulae. *Journal on Computational Applied Mathematics*, 7:67–75, 1981.

[12] T. F. Edgar and D. M. Himmelblau. *Optimization of Chemical Processes*. McGraw-Hill, New York, 1988.

[13] D. Goldfarb and A. Idnani. A numerically stable method for solving strictly convex quadratic programs. *Mathematical Programming*, 27:1–33, 1983.

[14] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I: Non-stiff Problems*, volume 8 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, Heidelberg, New York, 1993.

[15] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*, volume 14 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, Heidelberg, New York, 1991.

[16] J. Ingham, I. J. Heinzle, and J. E. Prenosil. *Chemical Engineering Dynamics*. VCH Verlagsgesellschaft, Weinheim, 1994.

[17] S.-S. Jang, B. Joseph, and H. Mukai. Online optimization of constrained multivariable chemical processes. *AIChE Journal*, 33, 1987.

[18] R. Luus. Iterative dynamic programming: From curiosity to a practical optimization procedure. *Control and Intelligent Systems*, 26:1–8, 1998.

[19] H. Maurer and W. Gillessen. Application of multiple shooting to the numerical solution of optimal control problems with bounded state constraints. *Computing*, 15:105–126, 1975.

[20] N. Nishida, A. Ichikawa, and E. Tazaki. Optimal design and control in a class of distributed parameter systems under uncertainty. *AIChE Journal*, 18:561–568, 1972.

[21] H. J. Oberle and W. Grimm. BNDSCO – a program for the numerical solution of optimal control problems. Technical Report 515-89/22, DLR, Oberpfaffenhofen, 1989.

[22] L. O. Santos, N. M. C. de Oliveira, and L. T. Biegler. Reliable and efficient optimization strategies for nonlinear model predictive control. In *Proc. Fourth IFAC Symposium on Dynamics and Control of Chemical Reactors, Distillation Columns and Batch Processes, DYCORD+'95*, pages 33–38. Elsevier Science, Oxford, 1995.

[23] W. E. Schiesser. *The Numerical Method of Lines*. Academic Press, San Diego, 1991.

[24] K. Schittkowski. On the convergence of a sequential quadratic programming method with an augmented Lagrangian line search function. *Optimization*, 14:1–20, 1983.

[25] K. Schittkowski. NLPQL: A Fortran subroutine solving constrained programming problems. *Annals of Operations Research*, 5:485–500, 1985.

[26] H. Schuler. *Prozeßsimulation*. VCH Verlagsgesellschaft, Weinheim, 1995.

[27] P. Tanartkit and L. T. Biegler. A nested, simultaneous approach for dynamic optimization problems. *Computers Chem. Engineering*, 20:735–741, 1996.

[28] V. S. Vassiliadis, R. W. H. Sargent, and C. C. Pantelides. Solution of a class of multistage dynamic optimization problems. 2. problems with path constraints. *Industrial Engineering and Chemical Research*, 33:2123–2133, 1994.

[29] R. W. Wansbrough. Modeling chemical reactors. *Chemical Engineering*, 5:95–102, 1985.

A Nomenclature

a_i	Reaction order of the i-th reaction.
$A(x,t)$	Free cross-sectional area of the reactor.
$A^{\min}(t)$	Lower bound for the cross-sectional area.
$A_\star(x)$	Initial distribution of the cross-sectional area at time t_0.
A_0	Cross-sectional area of the cleaned reactor.
β	Parameter that describes the decrease of the free cross-sectional area.
$c_{\mathrm{p}}(x,t)$	Total heat capacity of the mixture.
$c_{\mathrm{p}k}(x,t)$	Heat capacity of the k-th component.
$C(x,t)$	Vector that contains the molar concentrations of all components.
$C_k(x,t)$	Molar concentration of the k-th component.
$C_k^0(t)$	Initial molar concentration of the k-th component.
ε	Reaction parameter that describes the coke formation.
E_i	Activation energy of the i-th reaction.
$F_1(\dots)$	Model function that describes the change of the molar concentrations.
$F_2(\dots)$	Model function that describes the change of the temperature.
$F_3(\dots)$	Model function that describes the change of the cross-sectional area.
$G(\dots)$	Total profit of the reactor over the whole time horizon.
ΔH_i	Heat of reaction of the i-th reaction.
k_i	Reaction constant of the i-th reaction.
l	Number of space discretizations.
L	Length of the reactor.
$\dot{m}(t)$	Total mass flow in the reactor.

$\dot{m}_j(x,t)$	Mass flow of the j-th component.
$\dot{m}_j^{\max}(t)$	Maximum demand of the j-th component.
$\dot{m}_j^{\min}(t)$	Minimum demand of the j-th component.
M_j	Molar weight of the j-th component.
n	Average number of H atoms in CH_n.
n_{c}	Number of reactor cleanings.
n_{t}	Number of discrete times in each runtime interval.
P_{c}	Costs of a reactor cleaning.
$P_j(t)$	Price of the j-th component.
$\rho(x,t)$	Specific gravity of the mixture.
r	Total number of components.
r_{e}	Number of input feeds.
r_i	Reaction rate of the i-th reaction.
R	Gas constant.
τ	Total running time of the process.
t	Time coordinate.
$\Delta t^{\min}$	Minimum length of each runtime interval.
t_0	Starting time of the process.
t_{c}	Vector that contains the times of all reactor cleanings.
t_{c}^i	The time of the i-th reactor cleaning.
t_j^i	The j-th discrete time in the i-th runtime interval.
$T(x,t)$	Temperature.
$T^{\max}(t)$	Upper bound for the temperature.
T_0	Initial temperature at the reactor entry.
T_{r}	Average reaction temperature.
$u(t)$	Vector that contains all control variables.
$u_j(t)$	Input feed of the j-th component.
$u_j^{\max}(t)$	Upper bound on $u_j(t)$.
$u_j^{\min}(t)$	Lower bound on $u_j(t)$.
U	Set of all feasible controls.
$v(x,t)$	Velocity of the mixture in the reactor.
x	Space coordinate.
x_j	The j-th point at which state constraints are evaluated.

B Reaction Data

$$
\begin{array}{rrr}
a_1 = 0.72 & E_3 = 3.4 \cdot 10^4 & k_5 = 3.6 \cdot 10^4 \\
a_2 = 0.56 & E_4 = 3.2 \cdot 10^4 & M_1 = 16.0 \\
a_4 = 0.56 & E_5 = 3.0 \cdot 10^4 & M_2 = 32.0 \\
\beta = 1.4 \cdot 10^{-5} & \varepsilon = 1.0 \cdot 10^{-3} & M_3 = 26.0 \\
c_{\mathrm{p1}} = 2.219 & \Delta H_1 = -263.0 \cdot 10^3 & M_4 = 44.0 \\
c_{\mathrm{p2}} = 0.917 & \Delta H_2 = 277.5 \cdot 10^3 & M_5 = 2.0 \\
c_{\mathrm{p3}} = 1.683 & \Delta H_3 = 283.0 \cdot 10^3 & M_6 = 28.0 \\
c_{\mathrm{p4}} = 0.837 & \Delta H_4 = 226.7 \cdot 10^3 & M_7 = 18.0 \\
c_{\mathrm{p5}} = 14.32 & \Delta H_5 = 241.8 \cdot 10^3 & M_8 = 14.0 \\
c_{\mathrm{p6}} = 1.042 & k_1 = 1.8 \cdot 10^4 & n = 2.0 \\
c_{\mathrm{p7}} = 2.500 & k_2 = 9.0 \cdot 10^6 & R = 8.314 \\
c_{\mathrm{p8}} = 1.595 & k_3 = 2.6 \cdot 10^5 & \rho_0 = 0.047 \\
E_1 = 2.7 \cdot 10^4 & k_4 = 8.9 \cdot 10^2 & T_{\mathrm{r}} = 873.15 \\
E_2 = 3.8 \cdot 10^4 & &
\end{array}
$$

C Price and Demand Data

Table 7. Component prices

component	$t = 0$	$t = 50$	$t = 100$	$t = 150$	$t = 200$
CH_4	0.18	0.21	0.20	0.18	0.15
O_2	0.052	0.053	0.050	0.049	0.048
C_2H_2	1.9	2.0	1.9	1.8	1.7
CO_2	0.0	0.0	0.0	0.0	0.0
H_2	0.054	0.054	0.053	0.052	0.052
CO	0.054	0.055	0.05	0.049	0.048
H_2O	0.0	0.0	0.0	0.0	0.0
CH_n	0.2	0.2	0.15	0.13	0.1

Table 8. Minimum demands

component	$t = 0$	$t = 50$	$t = 100$	$t = 150$	$t = 200$
CH_4	0.0	0.0	0.0	0.0	0.0
O_2	0.0	0.0	0.0	0.0	0.0
C_2H_2	30.0	40.0	30.0	20.0	10.0
CO_2	0.0	0.0	0.0	0.0	0.0
H_2	20.0	20.0	10.0	10.0	10.0
CO	8.0	8.0	6.0	4.0	3.0
H_2O	80.0	80.0	50.0	40.0	3.0
CH_n	0.4	0.4	0.2	0.2	0.1

Table 9. Maximum demands

component	$t = 0$	$t = 50$	$t = 100$	$t = 150$	$t = 200$
CH_4	8000.0	9000.0	7000.0	6000.0	5000.0
O_2	8000.0	9000.0	5000.0	4000.0	3000.0
C_2H_2	3000.0	4000.0	3000.0	2000.0	1000.0
CO_2	8000.0	7000.0	6000.0	5000.0	4000.0
H_2	2000.0	2000.0	1000.0	1000.0	1000.0
CO	8000.0	8000.0	6000.0	4000.0	3000.0
H_2O	8000.0	8000.0	5000.0	4000.0	3000.0
CH_n	4000.0	4000.0	2000.0	2000.0	1000.0