

# Stationary and Dynamic Flowsheeting in the Chemical Engineering Industry

**Edda Eich-Soellner**, München,     **Peter Lory**, Regensburg,  
**Peter Burr**,     **Andreas Kröner**, Höllriegelskreuth

**Summary.** Mathematical models have become indispensable tools in the design and the operation of chemical plants. Practical experience in this field during the last decade has proven modern equation-oriented simulation techniques superior to the classical sequential modular approach. The equation-oriented concept strongly relies on efficient numerical methods for large, sparse algebraic and differential-algebraic systems of equations. The aims of the present paper are to introduce mathematical models for the most important unit operations in chemical engineering and to survey numerical methods for the solution of large systems of equations arising from modeling entire chemical production plants. Special attention is given to numerical problems still encountered in productive applications in the process industry.

**AMS Subject Classifications** (1991): 65C20, 65H10, 65L05, 65M20, 80A99.

**Keywords and Phrases:** Chemical process simulation; Mathematical models; Large nonlinear systems of equations; Sparse matrices; Method of lines; Differential-algebraic equations.

# 1 Introduction

The last decades have seen an increasing competition in the chemical engineering industry. This has forced companies to develop and apply mathematical simulation techniques to improve their products. Main issues in this context are plant economy and safety, minimum feedstock and utilities consumption and environmental impacts. The pursuit of these goals leads to a rapidly growing number of applications of computer simulations both in the design phase and in the operating phase of chemical plants such as refineries, ethylene or gas processing plants or air separators. These examples give some of the applications the authors have been involved in.

The most important steps in these processes are transformation of the components by chemical reactions and separation of mixtures to obtain certain purities. The effects of the processes are based on different physical and chemical properties such as volatility, dew and bubble point, density and solubility of the components in mixtures.

The key tool for modeling a chemical engineering plant is the *flowsheet*, an abstract, graph-oriented scheme of a planned or existing chemical plant. It contains in its nodes the relevant single process steps, so called *unit operations*, which are modelled independently. A unit operation comprises either an apparatus like a pump, a valve, a distillation column, a heat exchanger or a reactor as a whole or only one or more distinct parts of a more complex process step. The key function of a unit operation is to transform the incoming material and information streams to give new output streams. The directed edges of the graph represent the ideal flow of material, energy and information among the unit operations. The flow of material in pipelines is modeled by individual unit operation only if transportation deadtime and the effects of turbulent/laminar flow is to be considered.

Figure 1 gives an example for a flowsheet of an ethylene plant. The central role of flowsheets has made the term *flowsheeting* a synonym for the simulation of chemical engineering plants.

Most of the large scale industrial plants operate in a steady-state mode. The feedstock properties are constant in quantity and composition and - in the ideal case - the plant continuously yields products at a constant rate and quality. During the *design* or *process synthesis* of such plants an intense feedback from the customer is indispensable to achieve the goals that he sets. In recent years customer's requirements increased drastically concerning purity of products, minimization of feedstock, energy consumption, equipment costs and build time. Additionally, legal constraints for environmental impacts have to be strictly obeyed. With all major constraints in mind the necessary basic process steps are determined in a first design phase, called "basic engineering". Then, the flowsheet is refined gradually in the phase of "detailed engineering", leading to the exact dimensions of all unit operations, e.g. the areas of heat exchangers, volumes of tanks or number of trays in distillation columns. Also, different

Figure 1: Flowsheet of an ethylene plant

operating scenarios (summer/winter, day/night, different feedstock qualities) are investigated. Stationary flowsheeting is an indispensable tool for all of these tasks. In any cases mentioned, the mathematical problem is to solve a large system of nonlinear discontinuous equations with a sparse Jacobian matrix.

A more sophisticated approach to meet the above mentioned goals in a systematic way makes use of *mathematical methods of optimization*. Clearly, the objective function typically represents feedstock or energy consumption or the costs for the construction of the plant. The equality constraints are the equations that describe the flowsheet. Finally, inequality constraints typically force the purity of the products to obey certain limiting requirements of the customer or guarantee the satisfaction of legal environmental restrictions. The simultaneous optimization of various continuous process parameters together with integer quantities such as column feed or side draw points calls for mixed-integer nonlinear programming methods. These techniques allow even the systematic optimization of structural alternatives in the flowsheet network.

The *operation of an actual plant* poses a set of new tasks to be solved by optimization methods: First, the plant model has to be fitted to actual plant measurements by adjusting free parameters using data collected over an extended period of stable operation ("parameter tuning"). Second, a regression of measured data on the basis of this improved model gives reliable information on the exact state of the plant ("data reconciliation"). Third, the new target setpoints of the control system are determined by mathematical optimization on the basis of the previously reconciled plant model ("setpoint optimization").

All three steps are repeated in a cyclic manner. The first two tasks are inverse problems and use suitable versions of the least-squares function as objective functions. The objective functions for the latter task usually include maximum current profit and production yield as well as minimum feedstock and utilities consumption. Equality constraints arise from the equations describing the flowsheet. Inequality constraints typically force the purity of the products to obey certain minimum requirements of the customer or guarantee the satisfaction of legal environmental restrictions. An efficient application of all three steps requires the *on-line* operation of the control system and the plant model. On-line operation in this case comprises data transfer in both directions - from the process to the plant model and vice versa - at unique user controlled points in time. The simultaneous optimization of various continuous process parameters together with integer quantities such as column feed or side-draw points calls for mixed-integer nonlinear programming methods in the off-line analysis.

In recent years *dynamic simulation* has become an increasingly important tool in the process industries. It is used in a variety of *off-line* applications, i.e. with no direct interactions with the actual plant, like operability studies or safety and risk analyses. It allows the investigation of start-up and shut-down procedures and, together with optimal control techniques, even their systematic optimization. Most important, dynamic flowsheeting is the basis for the design of standard and advanced control strategies and a platform for the preliminary implementation of the control systems. This is decisive for a stable and efficient operation of a plant.

Furthermore, dynamic simulation models form the heart of *operator training systems*. These simulation programs are used long time before the completion of the plant to implement distributed control systems and to train future operator personnel in monitoring and controlling the design states of operation as well as emergency situations. These applications make tough demands on the efficiency and robustness of the integration algorithms. To represent the behavior of the actual plant the model must be solved in real-time or faster. Data exchanged with the *on-line* operating control system and operator input account for permanent discontinuities in input values.

Dynamic simulation is increasingly used for dynamic on-line optimization and identification of process information not directly available through measurements. Both applications are an integral part of modern model-based control systems. Mathematically, dynamic process simulation is governed by large systems of differential-algebraic equations with discontinuities.

Principally, simulators for the above mentioned tasks follow two different concepts in setting up and solving the plant simulation model:

- the sequential modular approach and
- the equation-oriented approach.

Both strategies are compared and discussed e.g. in [2, 17, 26, 32, 43]. Practi-

cal experience during the last decade has proven equation-oriented simulation technique superior to the sequential modular approach. It strongly relies on efficient numerical methods for large, sparse algebraic and differential-algebraic systems of equations.

The present paper focuses on these mathematical aspects of stationary and dynamic flowsheeting. Practical experience of the authors in these fields is strongly influenced by the development of the equation-oriented simulator OPTISIM<sup>1</sup>, an in-house development of Linde AG [6]. If not stated otherwise, model equations and numerical solution methods in the subsequent sections follow the approaches in this software product. It combines state-of-the-art mathematical methods with a huge amount of experience originating from a day-to-day use of various simulators in different areas of flowsheeting. The paper includes also results from individual research of the authors.

Section 2 presents the mathematical models of the most important units. For the simulation of a plant, these units must be combined to a flowsheet. This process is described in Section 3 with the help of a simple, yet instructive example. Section 4 focuses on stationary flowsheeting. It describes the numerical solution of the steady state equations, especially the treatment of the large, sparse systems and remedies for typical numerical difficulties. This section also surveys techniques for steady state optimization. Section 5 describes properties of differential-algebraic equations arising in dynamic flowsheeting. Special attention is given to the index of the model equations for typical unit operations and to the treatment of discontinuities.

## 2 Mathematical models of unit operations

This section introduces dynamic simulation models of the most important process steps found in chemical and refinery plants. All models are based on conservation laws for mass, energy and momentum. Since time constants related to mass and energy holdups are several magnitudes larger than those found in momentum transport phenomena and dominate the dynamic behavior, mass and energy balances are formulated dynamically whereas momentum balances are usually applied in a simplified and quasi-stationary form. Trivially, for directly calculating a steady state solution with the model equations discussed, accumulation terms on the left hand sides of the balance equations have to be neglected.

### 2.1 The flash unit

A basic operation in process engineering is the single-stage thermal separation of a multicomponent mixture (see Figure 2). The mixture is heated to the bubble-point. The ascending vapor is richer in the low boiling, i.e. volatile,

---

<sup>1</sup>OPTISIM<sup>®</sup> is a registered trademark of Linde AG.

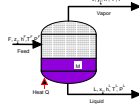


Figure 2: Flash unit

components than the liquid. As the vapor is removed continuously, the original mixture gets poorer in the light, more volatile components. The residue in the flash drum becomes richer in the heavy, less volatile components.

A single-stage equilibrium separation with neglectable holdup in the vapor phase is described by the following equations:

$$\frac{d}{dt}M = F - V - L, \quad (2.1a)$$

$$\frac{d}{dt}(M x_j) = F z_j - V y_j - L x_j \quad \text{for } j = 1, \dots, n_c - 1, \quad (2.1b)$$

$$\sum_{j=1}^{n_c} x_j = 1, \quad (2.1c)$$

$$\sum_{j=1}^{n_c} y_j = 1, \quad (2.1d)$$

$$y_j = K_j(T, P, x, y) x_j \quad \text{for } j = 1, \dots, n_c, \quad (2.1e)$$

$$L = \Phi(M), \quad (2.1f)$$

$$\frac{d}{dt}(M h^L) = F h^F - V h^V - L h^L + Q, \quad (2.1g)$$

$$h^L = h^L(T^L, P^L, x), \quad h^V = h^V(T^V, P^V, y), \quad (2.1h)$$

$$P^V = P, \quad P^L = P, \quad T^V = T, \quad T^L = T. \quad (2.1i)$$

These equations determine the following state variables:

$M$ : molar liquid holdup in the flash drum,

$L, V$ : molar liquid and vapor flowrates,  
 $x_j$ : mole fraction of component  $j = 1 \dots n_c$  in the liquid,  
 $y_j$ : mole fraction of component  $j = 1 \dots n_c$  in the vapor,  
 $h^L, h^V$ : molar enthalpies of liquid and vapor,  
 $T, T^L, T^V$ : temperatures in the drum and in the products,  
 $P^L, P^V$ : pressures in the products.

The pressure  $P$  in the drum, the added heat  $Q$ , the molar feed flowrate  $F$  as well as its mole fractions  $z_j$  and its molar enthalpy  $h^F$  are given. The number of components is denoted by  $n_c$ . The row vectors  $x$  and  $y$  collect the mole fractions of the liquid and the vapor, respectively:

$$x = (x_1, \dots, x_{n_c}), \quad y = (y_1, \dots, y_{n_c}). \quad (2.2)$$

Equation (2.1a) is the all over material balance, while (2.1b) are individual material balances for the first  $n_c - 1$  components. The last component is determined by (2.1c). This equation as well as (2.1d) expresses the fact, that the mole fractions in liquid and vapor sum up to 1. Thermodynamic equilibrium is ensured by (2.1d) and (2.1e). The latter equations describe the relation between the mole fractions in the vapor phase and in the liquid phase. The so-called  $K$ -values  $K_j$  depend on pressure, temperature and mole fractions of the liquid and vapor phase. They are determined by a physical property system based on given  $T, P, x$ , and  $y$ .

It is assumed, that the flowrate  $L$  of the liquid leaving the drum is a known function  $\Phi$  of the holdup  $M$  in the drum (2.1f). Equation (2.1g) is a total enthalpy balance, with the liquid and vapor molar enthalpies defined by (2.1h). Both values are also evaluated in a physical property system from  $T, P$ , and  $x$  or  $y$ , respectively. Equations (2.1i) equate pressures and temperatures in the product flows to those in the drum. The system consists of  $2n_c + 10$  equations for the same number of unknowns. For variants of the flash unit and its applications, see [41].

## 2.2 The distillation column

The extent of separation in a single-stage flash unit is limited to the composition difference between liquid and vapor phase at a given pressure  $P$  and boiling temperature. To increase the effect of separation and to achieve higher purities numerous equilibrium stages at different pressures and temperatures are stacked to give a tray distillation column (Figure 3). Pressure and temperature decrease from bottom to top. The simulation model of a distillation column is easily derived from the flash unit model described in the preceding section.

$$\frac{d}{dt}M_i = L_{i+1} + V_{i-1} - L_i - V_i + F_i, \quad (2.3a)$$

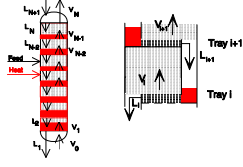


Figure 3: Distillation column

$$\frac{d}{dt}(M_i x_{i,j}) = L_{i+1} x_{i+1,j} + V_{i-1} y_{i-1,j} - L_i x_{i,j} - V_i y_{i,j} + F_i z_{i,j} \quad (2.3b)$$

for  $j = 1, \dots, n_c - 1$ ,

$$\sum_{j=1}^{n_c} x_{i,j} = 1, \quad (2.3c)$$

$$\sum_{j=1}^{n_c} y_{i,j} = 1, \quad (2.3d)$$

$$y_{i,j} = K_{i,j}(T_i, P_i, x_i, y_i) x_{i,j} \quad \text{for } j = 1, \dots, n_c, \quad (2.3e)$$

$$L_i = \Phi(M_i), \quad (2.3f)$$

$$\frac{d}{dt}(M_i h_i^L) = L_{i+1} h_{i+1}^L + V_{i-1} h_{i-1}^V - L_i h_i^L - V_i h_i^V + F_i h_i^F + Q_i, \quad (2.3g)$$

$$h_i^L = h_i^L(T_i, P_i, x_i), \quad h_i^V = h_i^V(T_i, P_i, y_i), \quad (2.3h)$$

$$P_i - P_{i+1} = \Psi(M_{i+1}, V_i). \quad (2.3i)$$

In this system,  $i = 1, \dots, N$  denotes the number of the tray, while  $j = 1, \dots, n_c$  is the number of the component in the mixture. Similarly to (2.2), the row

vectors  $x_i$  and  $y_i$  are defined as

$$x_i = (x_{i,1}, \dots, x_{i,n_c}), \quad y_i = (y_{i,1}, \dots, y_{i,n_c}). \quad (2.4)$$

The  $N(2n_c + 7)$  state variables are

- $M_i$ : molar liquid holdup on stage  $i$ ,
- $L_i$ : molar liquid flowrates descending from stage  $i$  to stage  $i - 1$ ,
- $V_i$ : molar vapor flowrates ascending from stage  $i$  to stage  $i + 1$ ,
- $x_{i,j}$ : mole fractions of component  $j$  in the liquid at stage  $i$ ,
- $y_{i,j}$ : mole fraction of component  $j$  in the vapor at stage  $i$ ,
- $h_i^L, h_i^V$ : molar enthalpies of liquid and vapor at stage  $i$ ,
- $T_i$ : temperature at stage  $i$ ,
- $P_i$ : pressure at stage  $i$ .

The external heat input  $Q_i$  to stage  $i$ , the feed flowrate  $F_i$  to stage  $i$  as well as its mole fractions  $z_{i,j}$  and its molar enthalpy  $h_i^F$  are given. The system (2.3a)–(2.3i) consists of  $N(2n_c + 7)$  equations. The following specifications complete the system: the reflux flowrate  $L_{N+1}$  to the top tray together with its mole fractions  $x_{N+1,j}$  and its molar enthalpy  $h_{N+1}^L$ ; the vapor flowrate  $V_0$  to the bottom tray as well as its composition  $y_{0,j}$  and the molar enthalpy  $h_0^V$ ; the external pressure at the column top  $P_{N+1}$ .

The rationale for these equations is the same as that for a single stage separation (see Sec. 2.1). Only the material and enthalpy balances have to be modified adequately. Equation (2.3i) expresses the pressure drop between two consecutive stages in terms of resistance incurred by the vapor flowing through the holes or valves in the tray, and also through the static height of liquid on the tray. The exact forms of the liquid flow function  $\Phi$  and vapor flow function  $\Psi$  depend on the type and geometry of the trays used. The pressure drop equation for the top tray  $i = N$  is usually formulated in a different way than the pressure drop across an internal tray since there is no external holdup  $M_{N+1}$ . Usually, the number  $N$  of trays is within  $10 \leq N \leq 100$ , and the number  $n_c$  of components is restricted to  $n_c \leq 50$ . For a more detailed discussion, see [18] and [31].

### 2.3 The heat exchanger

The main purpose of heat exchangers is to transfer energy among two or more fluid stream separated through the heat exchanger material. Hot streams leave the heat exchanger at a lower temperature as they enter the apparatus. The transferred energy is used to heat up cold streams on their way through the equipment. Depending on the relative flow direction in the passages of the exchanger there exist three principal modes of operation: cocurrent, counter-current, and crosscurrent. The decision for a certain mode of operation depends

on the number of passages, the maximum energy transfer rate, or constructive limitations. In low temperature processes, such as air separators or parts of an ethylene plant, countercurrent heat exchangers are most efficient and are therefore used predominantly. For demonstration purposes, Fig. 4 and the following model equations are restricted to two streams. The generalisation to more streams is straightforward.

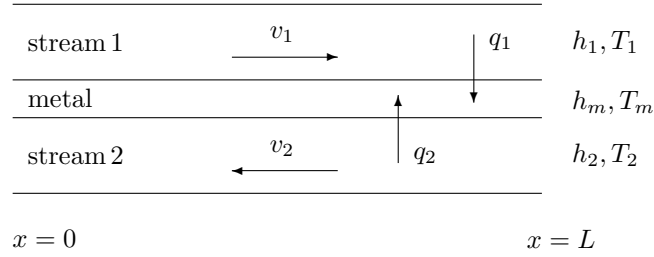


Figure 4: Heat exchanger

It is assumed that all streams exchange heat with the heat exchanger metal. The flow velocities  $v_1$  and  $v_2$  are taken constant. The enthalpy flowrates  $h_1(x, t)$ ,  $h_2(x, t)$  as well as the temperatures  $T_1(x, t)$ ,  $T_2(x, t)$  of the two streams depend on the spatial variable  $x$  and on time  $t$ . The state of the metal is described by the enthalpy  $h_m(x, t)$  per unit length and by the temperature  $T_m(x, t)$ . The energy transferred from a fluid to the metal is given by the heat transfer equation defining the energy flux per unit length

$$q_i = \alpha_i \Omega_i (T_i - T_m) \quad (i = 1, 2).$$

Here,  $\alpha_i = \alpha_i(T_i, P_i, z_i)$  is the heat transfer coefficient between the  $i$ -th stream and the heat exchanger wall. It depends on the flow regime and on physical properties and hence on temperature  $T_i$ , pressure  $P_i$  and composition  $z_i = (z_{i,1}, \dots, z_{i,n_c})$  of the stream. The quantity  $\Omega_i$  denotes the heat transfer area per unit length for the  $i$ -th stream.

The effect of heat transport in axial direction due to conduction and dispersion is usually much smaller than that of convection and is therefore neglected here. Consequently, the model does not contain diffusive terms and is of hyperbolic type:

$$\frac{1}{v_1} \frac{\partial h_1}{\partial t} = -\frac{\partial h_1}{\partial x} - \alpha_1 \Omega_1 (T_1 - T_m), \quad (2.5a)$$

$$\frac{1}{v_2} \frac{\partial h_2}{\partial t} = \frac{\partial h_2}{\partial x} - \alpha_2 \Omega_2 (T_2 - T_m), \quad (2.5b)$$

$$\frac{\partial h_m}{\partial t} = \alpha_1 \Omega_1 (T_1 - T_m) + \alpha_2 \Omega_2 (T_2 - T_m) , \quad (2.5c)$$

$$0 = h_{\text{stream}}(T_1, P_1, z_1) - h_1 , \quad (2.5d)$$

$$0 = h_{\text{stream}}(T_2, P_2, z_2) - h_2 , \quad (2.5e)$$

$$0 = h_{\text{metal}}(T_m) - h_m . \quad (2.5f)$$

The physical property functions  $h_{\text{stream}}$  and  $h_{\text{metal}}$  depend on the temperature and in the case of a stream also on pressure and composition – as indicated in (2.5d)–(2.5f). The concentrations are assumed constant along each heat exchanger tube, the pressure is assumed constant or varying linearly with the exchanger length. The system is completed by the initial conditions  $h_i(x, 0)$ ,  $h_m(x, 0)$  and the boundary conditions  $h_1(0, t)$ ,  $h_2(L, t)$ .

## 2.4 The ideal continuous flow stirred tank reactor

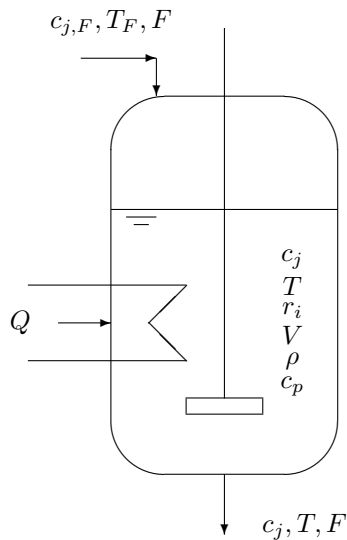


Figure 5: A continuous flow stirred tank reactor

The continuous flow stirred tank reactor consists of a vessel of constant liquid volume  $V$  furnished with inlet and outlet nozzles, a cooling water jacket and a stirrer assuring a uniform concentration and temperature distribution in the reactor volume. Inlet and outlet flow rate are assumed constant at rate  $F$ . Let  $A_j, j = 1, \dots, n_c$  denote the chemical species which are taking part in a system of  $M$  reactions. The equations for these reactions, which are assumed to be

independent, may be written as

$$\sum_{j=1}^{n_c} \alpha_{i,j} A_j = 0, \quad i = 1, \dots, M. \quad (2.6)$$

Here,  $\alpha_{i,j}$  is the stoichiometric coefficient of component  $A_j$  in the  $i$ -th reaction. Stoichiometric coefficients are taken positive for products and negative for reactants.

Let  $c_{j,F}$  be the molar concentration of  $A_j$  in the feed to the reactor, and  $c_j$  the molar concentration in the reactor and hence in the outflowing product. Then a mass balance for  $A_j$  gives

$$V \frac{dc_j}{dt} = F(c_{j,F} - c_j) + V \sum_{i=1}^M \alpha_{i,j} r_i, \quad j = 1, \dots, n_c. \quad (2.7)$$

Here,  $r_i$  is the total rate of reaction for the  $i$ -th reaction. For many technical reaction systems it is given by the power law rate equation

$$r_i = r_i(c_1, \dots, c_{n_c}, T) = k_i \prod_{j=1}^{n_c} c_j^{\beta_{i,j}} - k'_i \prod_{j=1}^{n_c} c_j^{\gamma_{i,j}}. \quad (2.8)$$

The Arrhenius rate constants  $k_i$  and  $k'_i$  of the  $i$ -th reaction in forward and backward direction obey the *Arrhenius equation*:

$$k = k_0 \exp\left(-\frac{E}{\mathcal{R}T}\right), \quad k' = k'_0 \exp\left(-\frac{E'}{\mathcal{R}T}\right), \quad (2.9)$$

where  $T$  is the temperature in the reactor. The constants  $k_0, k'_0$  and  $E, E'$  are frequency factors and activation energies, respectively. As usual,  $\mathcal{R}$  is the gas law constant. The exponents  $\beta_{i,j}$  and  $\gamma_{i,j}$  denote the order of the  $i$ -th reaction with respect to  $A_j$  in forward and backward direction, respectively. They obey the relation

$$\gamma_{i,j} - \beta_{i,j} = \alpha_{i,j}.$$

For a reversible reaction each direction must be considered separately, whereas for an irreversible reaction only the forward direction is relevant, hence  $k'_i = \gamma_{i,j} = 0$ .

Let  $T_F$  denote the temperature in the feed. Then, under reasonable assumptions (e.g. constant pressure) an enthalpy balance for the reactor gives

$$V \rho c_p \frac{dT}{dt} = F \rho c_p (T_F - T) + V \sum_{i=1}^M (-H_{\Delta,i}) r_i + Q. \quad (2.10)$$

Here,  $c_p$  is the constant molar heat capacity of the reaction mixture,  $\rho$  is the constant density,  $H_{\Delta,i}$  is the heat of the  $i$ -th reaction, and  $Q$  is the rate of heat entering the reactor. For more details, the reader is referred to [1, 37].

## 2.5 Physical property data

The model equations require physical property data, e.g. density, enthalpy or volume. They are given by explicit functions or as solutions of highly nonlinear implicit equations (thermodynamic state equations). In order to obtain the physical property data asked for by a unit model, these equations must be evaluated or solved.

The thermodynamic state equations implicitly relate temperature, pressure and volume. Often these equations, like the cubic state equation, have multiple solutions for certain variables. So, the choice of the solution to return depends on the thermodynamic state of the system. The physical property equations are solved iteratively in the physical property data system each time a unit operation model is evaluated. This approach is used in almost all steady state and dynamic simulation systems.

## 3 A flowsheet example

In this section the steps on the way to the simulation model of an actual process or plant are shown. The small example process in Fig. 6 is first explained and structured in a flowsheet. The model equations of the unit operations in the flowsheet are then combined to give the complete simulation model of the process.

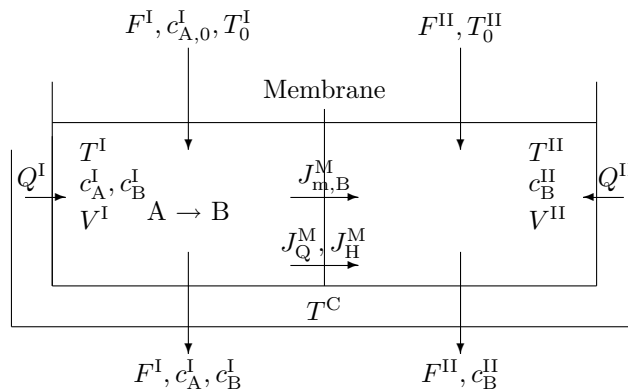
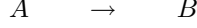


Figure 6: Two phase process

Fig. 6 shows the schematic of a tank reactor to produce component B from component A. The interior of the tank is separated in a reaction phase (I) and a product phase (II). The exothermic first order irreversible reaction in phase I

is



with heat of reaction  $H_{\Delta,R} < 0$ .

Both phases are separated by a selective membrane which is permeable only for component B. Heat is exchanged between both phases across the membrane.

Component A is fed into phase I in an aqueous solution by the volume stream  $F^I$ , with feed concentration  $c_{A0}^I$  and feed temperature  $T_0^I$ . The outlet from the reaction phase is given by volume stream  $F^I$  with concentration  $c_A^I$  and  $c_B^I$  of components A and B and temperature  $T^I$  of the reaction volume  $V^I$ . The pure washwater stream  $F^{II}$  with the temperature  $T_0^{II}$  is fed to volume  $V^{II}$  of the product phase. Its outlet temperature is  $T^{II}$  and the concentration of component B is  $c_B^{II}$ .

Both phases exchange heat with the cooling water through heat streams  $Q^I$  and  $Q^{II}$ .

For reaction systems some additional modeling assumptions are valid [37]:

- Densities and heat capacities are constant and equal to those of the pure solvent.
- No phase changes occur; all phases are liquid.
- Heat and mass transfer coefficients are constant.
- The temperature  $T^C$  of the cooling water is constant.

To set up a simulation model the process is first divided into smaller units to give the flowsheet of Fig. 7. The central unit operations in the flowsheet are a reactor and a tank unit with dynamic material and energy balances modeling the reaction phase (3.1) and product phase (3.2). The rate of reaction is modeled by the Arrhenius equation (3.1d). The membrane is assumed to have no storage capacities (3.5). The mass transfer stream (3.5b) is modeled proportional to the concentration difference in both phases. The energy exchange comprises two phenomena. The conduction heat flow  $J_Q^M$  (3.5a) is proportional to the temperature difference of both phases. The enthalpy transport  $J_H^M$  (3.5c) is caused by mass transfer through the membrane. It is proportional to the mass flow  $J_{m,B}^M$  and the temperature difference across the membrane.

The heat exchanged with the cooling water is modeled for both phases in separate elementary exchanger units assuming the heat transfer to be proportional to the temperature difference between the phase temperatures and the cooling water (3.3,3.4).

$$V^I \frac{dc_A^I}{dt} = F^I(c_{A0}^I - c_A^I) - r_0 V^I \quad (3.1a)$$

$$V^I \frac{dc_B^I}{dt} = -F^I c_B^I + r_0 V^I - J_{m,B}^M \quad (3.1b)$$

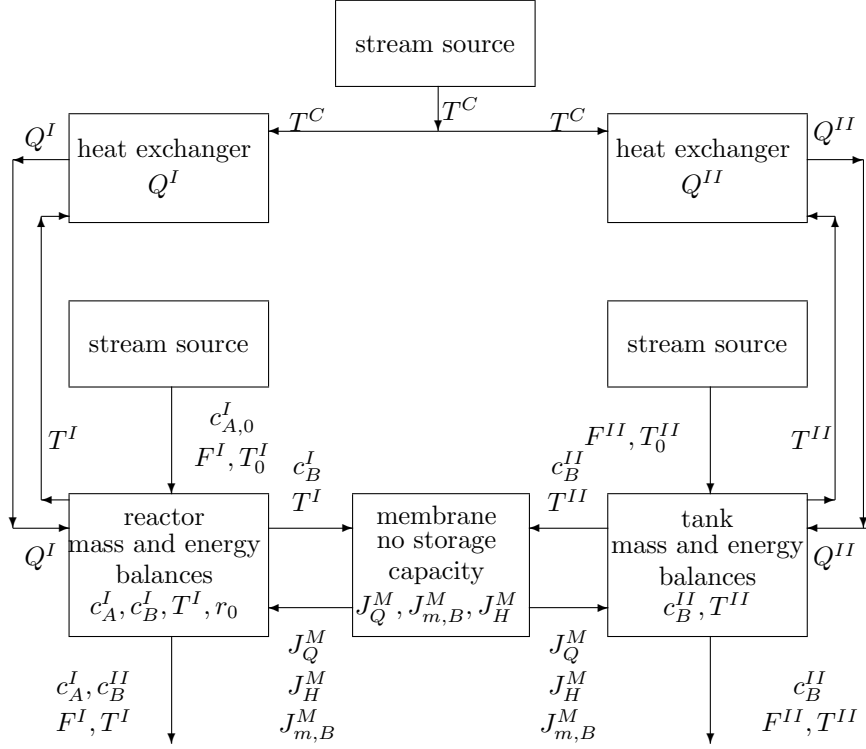


Figure 7: Flowsheet of the two phase process

$$V^I \rho c_p \frac{dT^I}{dt} = F^I \rho c_p (T_0^I - T^I) + Q^I - J_Q^M - J_H^M - r_0 V^I H_{\Delta,R} \quad (3.1c)$$

$$0 = r_0 - c_A^I k_0 e^{-\frac{E}{RT^I}} \quad (3.1d)$$

$$V^{II} \frac{dc_B^{II}}{dt} = -F^{II} c_B^{II} + J_{m,B}^M \quad (3.2a)$$

$$V^{II} \rho c_p \frac{dT^{II}}{dt} = F^{II} \rho c_p (T_0^{II} - T^{II}) + Q^{II} + J_Q^M + J_H^M \quad (3.2b)$$

$$0 = Q^I - k^I A^I (T^C - T^I) \quad (3.3)$$

$$0 = Q^{II} - k^{II} A^{II} (T^C - T^{II}) \quad (3.4)$$

$$0 = J_Q^M - k^M A^M (T^I - T^{II}) \quad (3.5a)$$

$$0 = J_{m,B}^M - \beta A^M (c_B^I - c_B^{II}) \quad (3.5b)$$

$$0 = J_H^M - c_p (T^I - T^{II}) J_{m,B}^M \quad (3.5c)$$

Source units for information and material streams are included to define required input information.

The procedure to analyze a plant and to set up the flowsheet is similar in almost all stationary and dynamic flowsheet oriented simulation packages [6, 20, 24, 30]. However, they differ significantly in the solution strategies and hence in the way the system of equations is generated.

In OPTISIM the model equations with their partial derivatives are available in executable form in the model library. Prior to a simulation, the flowsheet information is analyzed, input and output variables of the unit models are assigned internally and a calling sequence to the model equation programs is generated. If a numerical solution algorithm asks for a function evaluation of the simultaneously solved plant model the sequence is interpreted for calls to the equation set of the individual unit operations. The individual function evaluations are combined to give the entire plant information. Similar strategies are implemented in DIVA [20].

The resulting state vector associated with the simulation model (3.1)–(3.5) is

$$x = (c_A^I, c_B^I, T^I, r_0, c_B^{II}, T^{II}, Q^I, Q^{II}, J_Q^M, J_{m,B}^M, J_H^M)^T .$$

Input or control quantities

$$u = (c_{A0}^I, T_0^I, F^I, T_0^{II}, F^{II}, T^C)^T$$

of the system are provided by input unit operations.

In SpeedUp [30] model equations are provided and analyzed symbolically. The program code of the equation system is generated and compiled as a whole before a simulation can be run. Somewhat different simulation strategies are pursued in HYSYS [24] and PROTISS<sup>2</sup>.

## 4 Stationary flowsheeting

**Spatial discretization of the heat exchanger equations.** The spatial discretization of the heat exchanger equations is described here for the dynamic version already. Trivially, this includes the stationary case. The set of coupled partial differential and algebraic equations (2.5) is transformed into a

---

<sup>2</sup>PROTISS is a product of SIMSCI, UK.

differential-algebraic system of the type (5.1) by the *method of lines*. Here, finite element approximations

$$h_i(x, t) = \sum_{k=1}^N \eta_{i,k}(t) \phi_k(x) \quad (4.1)$$

with basis functions  $\phi_k(x)$  and time dependent coefficients  $\eta_{i,k}(t)$  are used for the spatial discretization of the partial differential equations (2.5a)–(2.5c). Stability requires *upwinding* within the frame of a Petrov–Galerkin method. This technique takes test functions  $\psi_k(x)$ , which are different from the basis functions, and leads to upwinded, nonsymmetric Galerkin equations (see e.g. [40]).

The first derivative of the physical property function  $h$  (enthalpy) is discontinuous at the temperatures  $T = T_{BP}$  (Bubble–Point) and  $T = T_{DP}$  (Dew–Point) due to phase changes. Consequently, the solutions of the system (2.5) show discontinuities at those locations where the phases change. Naturally, their positions in the heat exchanger depend on time  $t$ . A discretization, that does not destroy the order of the approximation, requires the knowledge of these positions  $x_{BP}(t)$  and  $x_{DP}(t)$ . They provide valuable information to the process engineer, too. Therefore, finite elements with *moving nodes* [28] are used. The principle is outlined here for a single phase change location  $x_{BP}(t)$ . The key is a moving coordinate system that tracks  $x_{BP}(t)$ . For example, a time dependent, piecewise linear coordinate transformation  $x = x(\xi; x_{BP}(t))$  with the properties

$$x(0; x_{BP}(t)) = 0, \quad x\left(\frac{L}{2}; x_{BP}(t)\right) = x_{BP}(t), \quad x(L; x_{BP}(t)) = L$$

defines a new coordinate system with the variable  $\xi$ , in which the phase change takes place at the constant position  $\xi = L/2$  (cf. [36]). Assuming that the phase change takes place in stream 1, the additional variable  $x_{BP}$  is determined by the equation

$$T_1(x_{BP}(t), t) - T_{BP} = 0. \quad (4.2)$$

#### 4.1 Steady state simulation

Large scale chemical plants deliver products at a constant rate with constant quality, i.e. concentration, temperature, etc., based on feeds available at uniform conditions. In other words, the plants operate in a steady state mode. The corresponding mathematical task is the solution of a set of nonlinear equations

$$F(x, p, \text{sign } q) = 0, \quad (4.3a)$$

$$q = q(x, p) \quad (4.3b)$$

where  $F \in \mathfrak{R}^{n_x}$  are the model equations with all time derivatives set to zero. The vector  $x$  collects state variables such as concentrations, material and energy

flow rates, temperatures, pressures, or material and energy holdups. The vector  $p$  summarizes the given parameters, e.g. exchanger areas or vessel volumes. The so-called *switching functions*  $q$  are used to switch among different states of the system described by different subsets of equations. Changing from one set of equations to another causes discontinuities in the solution of (4.3a). Typical reasons for discontinuities in the steady state case are control devices, the appearance or disappearance of phases, piecewise continuous physical property correlations or limitations in rate equations.

This section shows how to obtain a steady state solution of a large system of nonlinear, discontinuous equations by variants of Newton's method taking into account the special structure and properties, e.g. sparsity, of the problem. Approaches to obtain optimal steady states are discussed thereafter.

Eq. (4.3a) is solved using an inexact Newton method

$$\frac{\partial F}{\partial x}(x^{(i)})\Delta x^{(i)} = -F(x^{(i)}, p), \quad x^{(i+1)} = x^{(i)} + \alpha\Delta x^{(i)}, \quad (4.4)$$

where  $\alpha \in (0, 1]$  is the step length [38]. The system of linear equations in (4.4) is solved iteratively, see below.

**Solution of sparse linear equations.** A simulation model for the on-line optimization of an ethylene plant consists typically of about 40.000 equations, the Jacobian has about 200.000 non-zero entries. Because of stability and computing time reasons, the high dimension of the systems requires sparse matrix methods.

Figure 8: A typical sparsity structure

Fig. 8 gives a typical structure of non-zero elements in the Jacobian. Unfortunately, the matrices do not exhibit properties that would facilitate their decomposition, like bandedness, symmetry, positive definiteness or diagonal dominance. Rather, general sparse matrix algorithms have to be used for their factorization [10]. These methods are combined with iterative methods for linear equations in order to minimize effort and to increase accuracy. The latter feature is decisive because the matrices are often ill-conditioned (see e.g. [38]).

Principally, direct methods for sparse systems differ from the standard Gaussian algorithms in their pivot selection strategy. Sparse matrix methods do not use only stability criteria, but also criteria to minimize additional fill-in during the decomposition phase. In general, the solution of a sparse linear system is structured in three phases:

1. Analyze phase: The structure of non-zeroes is analyzed and a pivot sequence is determined.
2. Decomposition phase: The  $LU$ -decomposition is computed based on the previously selected pivot-sequence.
3. Solution phase: For a given right hand side a solution is determined by forward-backward substitution.

The computational effort decreases from step 1 to step 3.

In OPTISIM the computation of the first two steps is avoided as far as possible by iteratively solving the linear equations (4.4). The equations are preconditioned with a previously decomposed matrix, so that the factorization of the current Jacobian need not be computed. Only in the case of poor convergence, a change in the structure of the Jacobian, or discontinuities the solution procedure is restarted with phase 1 or phase 2.

The use of sparse matrix methods in OPTISIM is particularly facilitated because all unit models provide analytical derivatives. Consequently, the sparsity structure is explicitly known and no numerical tests with finite differences are necessary to decide if an element is 0 or not.

For the solution of the linear system (4.4) iterative methods GMRES (generalized **minimal residual**) [34, 42], "biconjugate gradient" [33] or "iterative refinement" together with the above mentioned preconditioning are applied. For a comparative survey, see [35].

**Discontinuities.** Discontinuities occur because the structure of equations or parameter values depend discontinuously on values of state variables. If this is the case, discontinuities are inevitable during the solution of the nonlinear system (4.3a) because the final solution and hence the structure of model equations and certain parameter values are unknown initially.

A discontinuity occurs whenever a switching function (4.3b) changes sign. Different impacts on the equation system must be handled:

- a:** Changes in the equation with no effect on the Jacobian,
- b:** Changes in the values of the Jacobian elements while keeping the sparsity structure unchanged,
- c:** Changes in the dependencies of the equations leading to a different sparsity structure of the Jacobian,
- d:** Changes in the number of equations and variables.

Each of these discontinuities require different actions w.r.t. convergence control (**a**), recomputation, reanalysis and decomposition of the Jacobian (**b**, **c**) or even reinitialization of the simulation model (**d**). The OPTISIM program keeps track of all discontinuities of the equations.

## 4.2 Steady state optimization

As discussed in the introduction, flowsheeting in practice often requires the solution of optimization problems:

$$\Phi(x, p) \stackrel{!}{=} \min_{x, p}, \quad (4.5a)$$

$$F(x, p) = 0, \quad (4.5b)$$

$$x_{\min} \leq x \leq x_{\max}, \quad p_{\min} \leq p \leq p_{\max}. \quad (4.5c)$$

In the optimization problem (4.5), let the vector

$$h(x, p) = \left( (x - x_{\min})^T, (x_{\max} - x)^T, (p - p_{\min})^T, (p_{\max} - p)^T \right)^T$$

summarize the inequality conditions and  $s = \left( x^T, p^T \right)^T$  the optimization variables. Then the system

$$\Phi(s) \stackrel{!}{=} \min_s, \quad (4.6a)$$

$$F(s) = 0, \quad (4.6b)$$

$$h(s) \geq 0 \quad (4.6c)$$

has to be solved. This is done by the iteration  $s^{(i+1)} = s^{(i)} + \Delta s^{(i)}$  starting from initial values  $s^{(0)}$ . The main difference among optimization methods is the way the search direction  $\Delta s^{(i)}$  is computed.

Sequential-linear programming (SLP) is popular in the chemical engineering community because of its robustness. Here, the objective function and the constraints are linearized around the current point  $s^{(i)}$ . Then the solution  $\Delta s^{(i)}$

of the linearized problem

$$\nabla\phi(s^{(i)}) \cdot \Delta s^{(i)} = \min_{\Delta s^{(i)}} \quad (4.7a)$$

$$F(s^{(i)}) + \frac{\partial F}{\partial s}(s^{(i)}) \cdot \Delta s^{(i)} = 0 \quad (4.7b)$$

$$h(s^{(i)}) + \frac{\partial h}{\partial s}(s^{(i)}) \cdot \Delta s^{(i)} \geq 0 \quad (4.7c)$$

is computed. Because this problem may be unbounded, an additional constraint on the step length is added

$$\|\Delta s^{(i)}\| \leq \alpha^{(i)}. \quad (4.8)$$

In order to ensure that the linearization is a good approximation to the nonlinear problem  $\alpha^{(i)}$  is controlled. For details about this "trust region" approach, see [14].

In contrast to the SLP method, the objective function in sequential quadratic programming (SQP) is approximated quadratically. Starting from a positive definite matrix, the Hessian is approximated successively using the BFGS-formula [12, 14].

**"Feasible path" and "infeasible path" methods.** Formulation (4.5) suggests the interpretation of the flowsheet equations (4.5b) as optimization constraints and to solve them as part of the optimization problem. Therefore, optimization, i.e. the determination of  $p$ , and the solution of flowsheet equations, i.e. the determination of  $x$  for given  $p$ , are performed simultaneously in the "infeasible path" approach. This can be done efficiently only if the sparsity structure of the Jacobian  $\left(\frac{\partial F}{\partial x}, \frac{\partial F}{\partial p}\right)$  and of the constraints is exploited. Efficient software for this task is still under development. This is currently the main reason not to use the above "infeasible path" approach, but a "feasible path" method:  $x = x(p^{(i)})$  is computed from the flowsheet equations outside the optimization procedure. The optimization solver can "see" only the optimization variables  $p$ . In general, feasible path methods require more effort, but they have the additional advantage that in every optimization iteration a valid solution of the flowsheet is available. This is especially advantageous if the optimization is not iterated until convergence.

For the feasible path approach, sensitivities  $\frac{\partial x}{\partial p}$  are necessary to compute derivatives of the objective function and constraints w.r.t.  $p$ , e.g.

$$\frac{d\Phi}{dp} = \frac{\partial\phi}{\partial x} \frac{\partial x}{\partial p} + \frac{\partial\phi}{\partial p}.$$

They can be obtained solving the linear system of equations

$$\frac{\partial F}{\partial x} \frac{\partial x}{\partial p} + \frac{\partial F}{\partial p} = 0.$$

Numerical results for the example of the two phase process can be found in the Appendix.

## 5 Dynamic flowsheeting

As can be seen from the examples in Sec. 2 and 4 dynamic models of chemical processes result in large systems of linearly implicit differential-algebraic equations (DAEs) of the form

$$\begin{pmatrix} A_1(y, z, p, u) & A_2(y, z, p, u) \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \dot{y} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} f(y, z, p, u) \\ g(y, z, p, u) \end{pmatrix}. \quad (5.1)$$

Let  $x = \begin{pmatrix} y \\ z \end{pmatrix}$  be the state variables with  $x \in \mathbb{R}^{n_x}$ ,  $y, f \in \mathbb{R}^{n_y}$ ,  $z, g \in \mathbb{R}^{n_x - n_y}$ , and  $A_1 \in \mathbb{R}^{n_y \times n_y}$  nonsingular. Parameters  $p \in \mathbb{R}^{n_p}$  are known constants whereas controls  $u \in \mathbb{R}^{n_u}$  are given functions in time.

Besides pure dynamic simulation, control and optimal control problems, frequency analysis and real-time training simulation are important tasks to be solved. The main focus of the following section is on the properties of the DAE system (5.1) and on aspects of the numerical solution of DAE systems not covered by standard DAE solution algorithms.

### 5.1 Typical properties of differential-algebraic equations occurring in Chemical Engineering

In practical applications, flowsheet DAE systems which consist of thousands of equations are common. For example, a rigorous model of an air separation plant involves about 1.000 differential and 3.000 algebraic equations, whereas ethylene plant models consist of some 30.000 equations.

The various unit operations show a wide span of time constants which leads to a high degree of stiffness in the model equations. Fast components are controllers, reactors or machinery like compressors. In contrast, distillation columns as well as storage tanks with large holdups account for large time constants. Stiffness also results from the spatial discretization of partial differential equations in the dynamic simulation of heat exchanger models. The discretization of a steady state heat exchanger model with streams of widely different flowrates leads to stiffness w.r.t. to space.

An even more crucial property than stiffness for measuring difficulties in the theoretical and numerical treatment of differential-algebraic equations, is their index. Roughly spoken, the index is the number of times all or parts of the equations must be differentiated totally w.r.t. time to obtain a set of ordinary differential equations (5.2a) for all variables [3, 7, 13]

$$\dot{x} = \Phi(x, p, u) \quad (5.2a)$$

$$0 = \Psi(x, p, u) \quad (5.2b)$$

In addition to (5.2a) a set of constraint equations (5.2b) is obtained defining the solution manifold. Both sub-systems (5.2) are called the extended system of the DAE system (5.1). A different approach to DAEs is found in [16] leading to the perturbation index.

DAEs in chemical engineering may show an arbitrarily high index. In case of the index exceeding 2, they cannot be treated directly and reliably with modern standard integration methods [3]. To reduce the index to 1 or 0 the DAE system must be analyzed to determine the number of total differentiations of each equation.

For large scale nonlinear simulation models it is often not practicable to analyze the DAEs analytically and to reduce the index symbolically. Structural analysis algorithms have been proposed in [5, 29, 39] to determine bounds on the index, the number of dynamic degrees of freedom and the number of total differentiations of each equation to reduce the index to 1 or 0. The actual index reduction is then carried out by automatic differentiation [15] or numerical differentiation [22].

Both, automatic and numerical differentiation of equations with the aim to reduce the index of DAE cause a loss of stability in the numerical solution. This can be avoided using a projection method, see [11]. Another remedy is an appropriate reformulation of the model.

The model equations of Sec. 2 and 3 have been analyzed for their index:

**The flash unit** model is of index 2. The difficulty is to determine a differential equation for the vapor stream  $V$ . To avoid an index 2 problem the model could be reformulated. If the pressure  $P$  in the drum and the vapor pressure  $P^V$  are used as state variables and the first equality in (2.1i) is substituted by a valve equation relating the internal and external pressure to the vapor stream  $V$

$$V = \bar{\Psi}(P, P^V)$$

an index 1 DAE system is obtained.

**The distillation column** model is an index 1 DAE system. However, if different specifications are assumed in the model higher index problems are obtained [31].

**The heat exchanger** model has after discretization also index 1, both with and without moving nodes.

**The two phase process** from Sec. 3 implemented as a dynamic simulation problem with given input values has index 1 because each of the algebraic equations can be solved for the corresponding variable. For some associated dynamic design cases where output variables are specified and input variables are released, Table 1 gives the corresponding indices.

given time function	unknown time function	Index
$c_A^I$	$c_{A0}^I$	2
$T^{II}$	$F^I$	3
$T^I$	$c_{A0}^I$	3
$c_B^{II}$	$T_C$	4

Table 1: Design cases for two phase process and their index

**Example:**

Another example illustrating the index problem in chemical engineering are two coupled tanks as shown in Fig. 9.

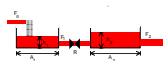


Figure 9: Communicating tanks

Both tanks are connected through a valve with flow resistance  $R$ . The dynamic behaviour is governed by

$$\begin{aligned}
 A_1 \dot{h}_1 &= F_0 - F_1 \\
 A_2 \dot{h}_2 &= F_1 - F_2 \\
 F_1 R &= h_1 - h_2 \\
 F_2 &= f(h_2) .
 \end{aligned}$$

If  $R$  and the input stream  $F_0$  are given, the unknowns, i.e. the liquid heights  $h_1, h_2$  and the flows  $F_1, F_2$  can be computed. The areas  $A_1, A_2$  are assumed to be given. For  $R \neq 0$  this is an index 1 system. For  $R = \infty$ , i.e. the valve is closed completely, the flow  $F_1$  becomes zero and the index remains 1. If the flow resistance  $R$  is decreased, the stiffness of the system increases. For  $R = 0$ , the levels in both tanks adjust instantaneously and the index of the system is 2. It is important to see that a direct algebraic coupling of balanced quantities increases the index of the system.

Fig. 10 gives the simulation results for

$$R(t) = \begin{cases} R_0(\frac{1}{2} + \frac{1}{2} \cos(\pi \frac{t}{t_0})) & t < t_0 \\ 0 & t \geq t_0 \end{cases},$$

with initial values  $h_1(0) = 1, h_2(0) = 1, f(h_2) = h_2, t = 8, F_0 = 2$ .

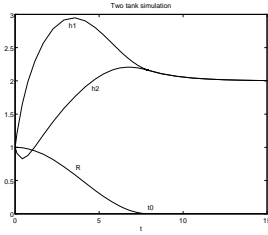


Figure 10: Dynamic simulation of two communicating tanks

If a design problem is to be solved the index increases. Two design cases are investigated:

**Case a:** Determine  $R$  such that  $h_2 = const.$

$$\begin{aligned} \Rightarrow \dot{h}_2 = 0 &= F_1 - F_2 = \frac{h_1 - h_2}{R} - F_2, \\ \Leftrightarrow R &= \frac{h_1 - h_2}{F_2}. \end{aligned}$$

Thus, the index is 2.

**Case b:** Determine  $F_0$  such that  $F_2 = const.$

$$\Rightarrow \frac{\partial f}{\partial h_2} \dot{h}_2 = 0 = \frac{\partial f}{\partial h_2} \frac{F_1 - F_2}{A_2},$$

Two cases must be distinguished:

- $R \neq 0$ :

$$0 = \frac{\partial f}{\partial h_2} \frac{h_1 - h_2}{R} - F_2.$$

Differentiation leads to an equation involving  $\dot{h}_1$  and thus  $F_0$ . Hence, this is an index 3 system.

- $R = 0$ :

$$0 = \frac{\partial f}{\partial h_2} \frac{F_1 - F_2}{A_2}.$$

This is an index 2 system.



Figure 11:  $N$  Communicating tanks

The situation may even become worse if more tanks are coupled. The dynamic behavior is governed by the system of equations

$$\begin{aligned} A_1 \dot{h}_1 &= F_0 - F_1 = F_0 + \frac{h_2 - h_1}{R_1}, \\ A_2 \dot{h}_2 &= F_1 - F_2 = -\frac{h_2 - h_1}{R_1} + \frac{h_3 - h_2}{R_2}, \\ &\vdots \\ A_i \dot{h}_i &= F_{i-1} - F_i = -\frac{h_i - h_{i-1}}{R_{i-1}} + \frac{h_{i+1} - h_i}{R_i}, \\ &\vdots \\ A_N \dot{h}_N &= F_{N-1} - F_N = -\frac{h_N - h_{N-1}}{R_{N-1}} - \underbrace{f(h_N)}_{:=ch_N} \end{aligned}$$

together with a specification: Determine  $F_0$  such that  $f(h_n) = \text{const}$ . This leads to a system of Hessenberg form of size  $N + 1$  and thus the index is  $N + 1$ .

$$0 = \begin{pmatrix} A_1 & & & & & \\ & A_2 & & & & \\ & & \ddots & & & \\ & & & A_N & & \\ 0 & \dots & & 0 & 0 & \end{pmatrix} \begin{pmatrix} \dot{h}_1 \\ \dot{h}_2 \\ \vdots \\ \dot{h}_N \\ \dot{F}_0 \end{pmatrix}$$

$$- \begin{pmatrix} -\frac{1}{R_1} & \frac{1}{R_1} & 0 & \dots & & 0 & 1 \\ \frac{1}{R_1} & -\frac{1}{R_1} - \frac{1}{R_2} & \frac{1}{R_2} & & & & 0 \\ 0 & \ddots & \ddots & \ddots & & & \\ 0 & \dots & & 0 & -\frac{1}{R_{N-1}} & -\frac{1}{R_{N-1}} - c & 0 \\ 0 & \dots & & & 0 & c & 0 \end{pmatrix} \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_N \\ F_0 \end{pmatrix}.$$

Tanks are only a special case of holdup devices with the output being an integral of the input. If the input to the first holdup is to satisfy a specification for the last output, the index is always one more than the number of storages in series. Some unit models such as distillation columns and flowsheets in general may contain numerous coupled holdups. Consequently, the index in a design task may be arbitrarily high.

## 5.2 Integration algorithms for DAE systems

A wide range of general purpose integration algorithms for the direct solution of DAEs with index 1 and 2 have been published in recent years. Implementations of the different approaches are:

- Backward differential formula methods DASSL[3] and DAESOL[11],
- Runge-Kutta methods RADAU5 [16],
- Extrapolation methods LIMEX [9] and MEXX [23].

Engineering approaches to solving higher index DAEs are found in [8, 19, 27]. The method of choice in OPTISIM is an implementation of a variable stepsize, variable order BDF method exploiting specific structures of the model equations available in the OPTISIM model library.

## 5.3 Treatment of discontinuities in the dynamic case

Compared to the previously discussed sources of discontinuities in steady state simulation, dynamic simulation shows even more cases for discontinuities: interactive user input, discrete measurements or controls, model switching due to different operation conditions and delays in the model equations. The latter depend not only on actual values of  $x^T = (y, z)$  but also on past values. Delays are used to model transport deadtimes in pipelines.

Discontinuities in dynamic simulation are also treated efficiently and reliably using switching functions. Rewriting (5.1) with switching function results in

$$\begin{pmatrix} A_1 & A_2 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \dot{y} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} f(y, z, p, u, \text{sign } q) \\ g(y, z, p, u, \text{sign } q) \end{pmatrix} \quad (5.3a)$$

$$q = q(y, z, p, u) \quad (5.3b)$$

with  $A_i = A_i(y, z, p, u, \text{sign } q)$ ,  $i = 1, 2$ . A point where a discontinuity occurs is described as a zero of a switching function (5.3b).

The main steps of a discontinuity algorithm in dynamic simulation are:

1. After every integration step the occurrence of a discontinuity is checked by testing the sign of the switching function.
2. If no discontinuity has occurred, the integration is continued. Otherwise the switching point is determined iteratively.
3. The right hand side of the equations are changed according to the new set of signs of the switching functions.
4. The discretization is adapted. In general, this involves a restart of the integration. For discontinuities in higher derivatives it might often be sufficient to reduce the order of the integration scheme.

For details of a switching algorithm, see e.g. [11].

For the iterative determination of the switching point, values of the switching function and thus of the values of the state variables  $x$  are necessary not only at the discretization points but also at intermediate points. They are computed using a continuous error-controlled representation  $P^x(t)$  of the solution in the last integration step which can be obtained from the integration scheme (here: the BDF-polynomial) [11]. In the case of an ODE, the switching point  $\hat{t}$  is given as solution of<sup>3</sup>  $q(P^x(\hat{t})) = 0$ . However, in the DAE case the situation is more complicated: a root  $\hat{t}$  of the switching function

$$q(P^y(\hat{t}), P^z(\hat{t})) = 0$$

does in general not imply that the algebraic equations are fulfilled at  $\hat{t}$ . Thus, the values are not consistent initial values for the next interval. It has been proposed<sup>4</sup> to obtain  $\hat{t}$  and the algebraic variable  $\hat{z}$  from

$$\begin{aligned} g(P^y(t), z) &= 0 \\ q(P^y(t), z) &= 0, \end{aligned}$$

which leads to consistent values. However, this method is restricted to semi-explicit DAE systems with index 1. For DAEs with index  $\geq 2$  we suggest to treat the integration step size (or the right point of the integration interval  $t$ ) as a variable. Assume for example an implicit Euler discretization for the system (5.3) (with  $A_1 = I, A_2 = 0$  for simplicity), the system

$$\begin{aligned} y &= y_n + (t - t_n)f(y, z) \\ 0 &= g(y, z) \\ 0 &= q(y, z) \end{aligned}$$

---

<sup>3</sup>Parameters  $p$  and controls  $u(t)$  are dropped for ease of notation.

<sup>4</sup>P.G. Thomsen, talk at the conference "Differential-algebraic equations: Theory and applications in technical simulation", Oberwolfach June 13-19, 1993.

can be solved for  $(\hat{y}, \hat{z}, \hat{t})$ . This is an  $(n_y + n_z + 1) \times (n_y + n_z + 1)$  system in contrast to the one dimensional root finding which has to be carried out in the ODE case using interpolants. In order to limit computational costs a good initial guess for  $\hat{y}, \hat{z}, \hat{t}$  is obtained using the interpolant.

## 5.4 Consistent initialization

Standard integration methods with order and stepsize control require consistent initial values  $(y_0, z_0), (\dot{y}_0, \dot{z}_0)$ , i.e. the initial values must satisfy not only the original DAE (5.1) but also the extended system (5.2). If initial values are inconsistent, e.g. at the very beginning of a new simulation experiment or after a discontinuity has occurred, the solution contains an error portion that cannot be controlled by conventional error control strategies. For the initialization of DAEs general approaches have been published [21, 22, 29].

To avoid inconsistent values in the solution after a discontinuity, smoothed functions, e.g. a ramp or a half cosine wave, are often used instead of step changes in parameters or input variables. However, in the DAE context non-differentiable solutions may occur even though the input function is continuously differentiable. The reinitialization of DAEs after discontinuities has been studied in [4, 25].

## References

- [1] R. Aris. *Elementary chemical reactor analysis*. Butterworth Publishers, Boston, 1989.
- [2] L. T. Biegler. Chemical process simulation. *Chemical Engineering Progress*, 85/10, 1989.
- [3] K.E. Brennan, S.L. Campbell, and L.R. Petzold. *The numerical solution of initial value problems in ordinary differential-algebraic equations*. North Holland Publishing Co., 1989.
- [4] L. Brüll and U. Pallaske. On consistent initialization of differential-algebraic equations with discontinuities. In *Proceedings of the Fourth European Conference on Mathematics in Industry*, pages 213–217, Stuttgart, 1991. B.G. Teubner.
- [5] P. Bujakiewicz and P.P.J. van den Bosch. Determination of perturbation index of a DAE with maximum weighted matching algorithm. In *Proc. of IEEE/IFAC Symposium CACSD '94*, Tucson, AZ, 1994.
- [6] P. Burr. The design of optimal air separation and liquefaction processes with the OPTISIM equation oriented simulator and its application to on-line and off-line plant optimization. In *Proc. AIChE Spring National Meeting*, April 1991, Houston, TX, 1991.

- [7] S.L. Campbell and C.W. Gear. The index of general nonlinear DAEs. *Numer. Math.*, 72:173–196, 1995.
- [8] Y. Chung and A. W. Westerberg. Solving a class of near index problems as perturbations of index problems. *Ind. Eng. Chem. Res.*, 31:2593–2603, 1992.
- [9] P. Deuffhard, E. Hairer, and J. Zugck. One-step and extrapolation methods for differential-algebraic systems. *Numer. Math.*, 51:501–516, 1987.
- [10] I. S. Duff, A. M. Erisman, and J. K. Reid. *Direct methods for sparse matrices*. Clarendon Press, Oxford, 1986.
- [11] E. Eich. *Projizierende Mehrschrittverfahren zur numerischen Lösung der Bewegungsgleichungen technischer Mehrkörpersysteme mit Zwangsbedingungen und Unstetigkeiten*. PhD thesis, Institut für Mathematik, Universität Augsburg, 1991. "VDI-Fortschrittsberichte", Reihe 18, Nr. 109, VDI-Verlag, Düsseldorf, 1992.
- [12] R. Fletcher. *Practical methods of optimization*. John Wiley & Sons, 2nd edition, 1987.
- [13] C. W. Gear. Differential-algebraic equation index transformation. *SIAM J. Sci. Stat. Comp.*, 7:39–47, 1988.
- [14] P.E. Gill, W. Murray, and M.H. Wright. *Practical optimization*. Addison Wesley, 1981.
- [15] A. Griewank and G.F. Corliss, editors. *Automatic differentiation of algorithms: Theory, implementation and application*, Philadelphia, 1991. SIAM.
- [16] E. Hairer and G. Wanner. *Solving ordinary differential equations II*. Springer, 1991.
- [17] M. Hillestad and T. Hertzberg. Dynamic simulation of chemical engineering systems by the sequential modular approach. *Comput. chem. Engng.*, 10:377–388, 1986.
- [18] C.D. Holland and A.I. Liapis. *Computer methods for solving dynamic separation problems*. McGraw-Hill, New York, 1983.
- [19] R.B. Jarvis and C.C. Pantelides. A differentiation-free algorithm for solving high-index DAE systems. In *Proc. AIChE 1992 Annual Meeting*, Miami Beach, 1.-6. November 1992 1992. American Institute of Chemical Engineers. Paper 146g.

- [20] A. Kröner, P. Holl, W. Marquardt, and E.-D. Gilles. DIVA – An open architecture for dynamic simulation. *Comput. chem. Engng.*, 14:1289–1295, 1990.
- [21] A. Kröner, W. Marquardt, and E.D. Gilles. Computing consistent initial conditions for differential-algebraic equations. Proc. ESCAPE-1, May 1992, Elsinore, Denmark. *Comput. chem. Engng.*, 16, Suppl.:131–138, 1992.
- [22] B. Leimkuhler, L. R. Petzold, and C. W. Gear. Approximation methods for the consistent initialization of differential-algebraic equations. *SIAM J. Numer. Anal.*, 28:205–226, 1991.
- [23] Ch. Lubich.  $h^2$  extrapolation methods for differential-algebraic systems of index 2. *Impact Comp. Sci. Eng.*, 1:260–268, 1990.
- [24] D.P. Mahoney and O. Santollani. HYSYS – an integrated system for process engineering and control. Hyprotech, Ltd., Calgary, Alberta, 1994.
- [25] C. Majer, W. Marquardt, and D. Gilles. Reinitialization of DAEs after discontinuities. Proc. ESCAPE-5, June 1995, Bled, Slovenia. *Comput. chem. Engng.*, 19, Suppl.:507–512, 1995.
- [26] W. Marquardt. Dynamic process simulation - Recent progress and further challenges. In *Proc. Fourth International Conference on Chemical Process Control*, pages 131–180, Padre Island, Texas, 1991.
- [27] S. E. Mattsson and G. Söderlind. Index reduction in differential-algebraic equations using dummy derivatives. *SIAM J. Sci. Stat. Comput.*, 14:677–692, 1993.
- [28] K. Miller and R.N. Miller. Moving finite elements, Part I. *SIAM J. Numer. Anal.*, 18:1019–1032, 1981.
- [29] C. C. Pantelides. The consistent initialisation of differential-algebraic systems. *SIAM J. Sci. Stat. Comput.*, 9:213–231, 1988.
- [30] C. C. Pantelides. SpeedUp - Recent advances in process simulation. *Comput. chem. Engng.*, 12:745–755, 1988.
- [31] C.C Pantelides, D. Gritsis, K.R. Morison, and R.W.H. Sargent. The mathematical modelling of transient systems using differential-algebraic equations. *Comput. chem. Engng.*, 12:449–454, 1988.
- [32] J.D. Perkins. Equation-oriented flowsheeting. In A.W. Westerberg and H.H. Chien (eds.): *Foundations of computer aided chemical process design. CACHE publications*, Austin, pp. 309-367, 1983.
- [33] W.T H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes*. Cambridge University Press, 1992.

- [34] Y. Saad and M. H. Schultz. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7:856–869, 1986.
- [35] W. Schmid, M.Paffrath, and R.H.W. Hoppe. Application of iterative methods for solving nonsymmetric linear systems in the simulation of semiconductor processing. *Surv. Math. Ind.*, 5:1–26, 1995.
- [36] N. L. Schryer. Designing software for one-dimensional partial differential equations. *ACM Trans. Math. Software*, 16:72–85, 1990.
- [37] J. M. Smith. *Chemical engineering kinetics*. McGraw-Hill, Singapore, third edition, 1981.
- [38] J. Stoer and R. Bulirsch. *Introduction to numerical analysis*. Springer, New York, 2nd edition, 1993.
- [39] J. Unger, A. Kröner, and W. Marquardt. Structural analysis of differential-algebraic equation systems – Theory and applications. *Comput. chem. Engng.*, 19:867–882, 1995.
- [40] R. Wait and A. R. Mitchell. *Finite element analysis and applications*. John Wiley & Sons, Chichester, 1985.
- [41] S.M. Walas. *Phase equilibria in chemical engineering*. Butterworth Publishers, Boston, 1985.
- [42] H. F. Walker. Implementation of the GMRES method using Householder transformations. *SIAM J. Sci. Stat. Comput.*, 9:152–163, 1988.
- [43] S. E. Zitney and M. A. Stadtherr. Computational experiments in equation-based chemical process flowsheeting. *Comput. chem. Engng.*, 12, 1988.

**Authors’ addresses:**

Prof. Dr. Edda Eich-Soellner, Fachbereich Informatik/Mathematik, Fachhochschule München, Lothstr. 34, D–80323 München, Germany.

Prof. Dr. Peter Lory, Institut für Wirtschaftsinformatik, Universität Regensburg, D–93040 Regensburg, Germany.

Dr. Peter Burr and Dipl.-Ing. Andreas Kröner, Linde AG, Process Engineering and Contracting Division, Dr.-Carl-von-Linde-Str. 6-14, D–82049 Höllriegelskreuth, Germany

## A Numerical data for the two phase problem

### A.1 The parameter set

The constant parameters and input variables of the two phase process are:

cooling area	$A^I = 10^{-1}$	$m^2$
cooling area	$A^{II} = 10^{-1}$	$m^2$
membrane area	$A^M = 10^{-1}$	$m^2$
mass transfer coefficient	$\beta = 10^{-2}$	$\frac{m}{s}$
feed concentration of A	$c_{A0}^I = 0.2$	$\frac{kmol}{m^3}$
molar heat capacity	$c_p = 83.2$	$\frac{kJ}{kg \cdot K}$
activation energy	$E = 12000$	$\frac{kJ}{kmol}$
heat of reaction	$H_{\Delta,R} = -1.93 \cdot 10^5$	$\frac{kJ}{kmol}$
flowrate through phase I	$F^I = 2 \cdot 10^{-4}$	$\frac{m^3}{s}$
flowrate through phase II	$F^{II} = 1 \cdot 10^{-4}$	$\frac{m^3}{s}$
frequency factor	$k_0 = 1.8 \cdot 10^5$	$\frac{l}{s}$
heat transfer coefficient	$k^I = 10$	$\frac{kW}{m^3 \cdot K}$
heat transfer coefficient	$k^{II} = 10$	$\frac{kW}{m^3 \cdot K}$
heat transfer coefficient	$k^M = 10.0$	$\frac{kW}{m^3 \cdot K}$
Gas law constant	$\mathcal{R} = 8.314$	$\frac{kJ}{kmol \cdot K}$
molar density	$\rho = 51.0$	$\frac{kg}{m^3}$
feed temperature	$T_0^I = 293$	K
feed temperature	$T_0^{II} = 293$	K
cooling water temperature	$T^C = 293$	K
volume	$V^I = 10^{-2}$	$m^3$
volume	$V^{II} = 10^{-2}$	$m^3$

## A.2 The steady state solution

To compute a steady state solution of this process equations (3.1d),(3.3)-(3.5) together with the steady state versions of equations (3.1a),(3.1b),(3.1c), (3.2) have to be solved. The following solution is obtained:

$c_A^I = 2.9059 \cdot 10^{-6}$	$\frac{kmol}{m^3}$
$c_B^I = 0.1375$	$\frac{kmol}{m^3}$
$c_B^{II} = 0.125$	$\frac{kmol}{m^3}$
$T^I = 296.17$	K
$T^{II} = 294.31$	K
$J_{m,B}^M = 1.25 \cdot 10^{-5}$	$\frac{kmol}{s}$
$Q^I = -3.168$	kW
$Q^{II} = -1.308$	kW
$J_Q^M = 1.8607$	kW
$J_H^M = 1.93 \cdot 10^{-3}$	kW
$r_0 = 4 \cdot 10^{-3}$	$\frac{kmol}{m^3 \cdot s}$

### A.3 The stationary optimization solution

The two phase process is optimized w.r.t. costs. Optimization parameters are the cooling temperature  $T^C$ , exchange areas  $A^M, A^I, A^{II}$ , volumes  $V^I, V^{II}$ , and the input concentration  $c_{A0}^I$  (i.e.  $p = (A^M, V^I, V^{II}, A^I, A^{II}, c_{A0}^I)$ ). The difference between investment and feed costs, resp., and proceeds is minimized:

$$\phi(x, p) = 100(V^I + V^{II}) + A^I + A^{II} + A^M + 100c_{A0}^I - 1000c_B^{II}$$

with constraints

$$\begin{aligned} T^{II} &\leq 300, \\ T^{II} &\leq 300, \\ 10^{-5} &\leq V^I \leq 10^{-1}, \\ 10^{-5} &\leq V^{II} \leq 10^{-1}, \\ 10^{-5} &\leq A^I \leq 10, \\ 10^{-5} &\leq A^{II} \leq 10. \end{aligned}$$

The solution of this optimization problem is  $A^M = 10^{-1} \text{ m}^2$ ,  $V^I = 4 \cdot 10^{-4} \text{ m}^3$ ,  $V^{II} = 10^{-5} \text{ m}^3$ ,  $A^I = 1.54 \text{ m}^2$ ,  $A^{II} = 10^{-5} \text{ m}^2$ ,  $c_{A0}^I = 0.3 \frac{\text{kmol}}{\text{m}^3}$ .