OULU 2010
C 377

ACTA UNIVERSITATIS OULUENSIS

Ilkka Malinen

IMPROVING THE ROBUSTNESS WITH MODIFIED BOUNDED HOMOTOPIES AND PROBLEM-TAILORED SOLVING PROCEDURES

UNIVERSITY OF OULU,
FACULTY OF TECHNOLOGY,
DEPARTMENT OF PROCESS AND ENVIRONMENTAL ENGINEERING,
CHEMICAL PROCESS ENGINEERING LABORATORY

ACTA UNIVERSITATIS OULUENSIS
C Technica 377

ILKKA MALINEN

IMPROVING THE ROBUSTNESS WITH MODIFIED BOUNDED HOMOTOPIES AND PROBLEM-TAILORED SOLVING PROCEDURES

Academic dissertation to be presented with the assent of the Faculty of Technology of the University of Oulu for public defence in Kuusamonsali (Auditorium YB210), Linnanmaa, on 21 January 2011, at 12 noon

UNIVERSITY OF OULU, OULU 2010

Acta Univ. Oul. C 377, 2010

Supervised by
Professor Juha Tanskanen

Reviewed by
Professor Ville Alopaeus
Professor Eric S. Fraga

ISBN 978-951-42-9337-5 (Paperback)
ISBN 978-951-42-9338-2 (PDF)
http://herkules.oulu.fi/isbn9789514293382/
ISSN 0355-3213 (Printed)
ISSN 1796-2226 (Online)
http://herkules.oulu.fi/issn03553213/

Cover Design
Raimo Ahonen

JUVENES PRINT
TAMPERE 2010

Malinen, Ilkka, Improving the robustness with modified bounded homotopies and problem-tailored solving procedures
University of Oulu, Faculty of Technology, Department of Process and Environmental Engineering, Chemical Process Engineering Laboratory, P.O.Box 4300, FI-90014 University of Oulu, Finland
Acta Univ. Oul. C 377, 2010
Oulu, Finland

Abstract

The aim of this work is to improve the overall robustness in equation-oriented chemical engineering simulation work. Because the performance of locally convergent solving methods is strongly dependent on a favourable initial guess, bounded homotopy methods were investigated as a way to enlarge the domain of convergence. Bounded homotopies make it possible to keep the homotopy path inside a feasible problem domain. Thus the fatal errors possibly caused by unfeasible variable values in thermodynamic subroutines can be avoided.

To enable the utilization of a narrow bounding zone, modifications were proposed for bounded homotopies. The performance of the modifications was studied with simple test problems and several types of distillation systems in the MATLAB environment.

The findings illustrate that modified bounded homotopies with variables mapping make it possible to bound the homotopy path strictly to run inside a feasible problem domain. The homotopy path can be tracked accurately and flexibly also inside a narrow bounding zone.

It was also noticed that by utilizing the concept of bounding the homotopy path with respect to the homotopy parameter, the possibility of approaching starting point and solution multiplicities is increased in cases where the traditional problem-independent homotopy method fails. The concept aims to connect separate homotopy path branches thus offering a trackable path with real space arithmetic.

Even though the modified bounded homotopies were found to overcome several challenges often encountered with traditional problem-independent homotopy continuation methods, alone they are not enough to guarantee that the solution is approached from an arbitrary starting point. Therefore, problem-tailored solving procedures were implemented in the consideration of complex column configurations. Problem-tailored solving procedures aim to offer feasible consecutive sub-problems and thus direct the solving towards the state distribution that fulfils exact product purity specifications.

As a whole, the modified bounded homotopies and problem-tailored solving procedures were found to improve the overall robustness of an equation-oriented solving approach. Thus the threshold for designing and implementing complex process systems such as complex distillation configurations for practical use could be lowered.

Keywords: bounded homotopies, chemical engineering, distillation, homotopy methods, MESH equations, path tracking, process modelling, simulation, solving methods

Preface

This study was carried out in the Department of Process and Environmental Engineering at the University of Oulu during the period 2003–2010.

I would like to thank my supervisor Prof. Juha Tanskanen for his encouragement to study homotopy continuation methods in chemical engineering model solving. The creative discussions and advice he gave me especially in the critical phases of publishing the research results greatly contributed to moving the work along.

I would also like to offer my gratitude to Prof. Ville Alopaeus at Aalto University and Prof. Eric S Fraga at University College London (UCL) who reviewed the manuscript of this thesis. Sue Pearson and Mike Jones from Pelc Southbank Languages are acknowledged for the linguistic corrections made to this thesis and several papers before.

I want to express my sincere thanks to all the past and present 'junior' and 'senior' members of the Chemical Process Engineering Laboratory. The coffee room debates and discussions not only cheered me up but also widened my understanding of all kinds of things from science and philosophy to more practical things, such as subatomic particles and the universe! My special thanks go to my colleagues Dr Juha Ahola and Mr. Jani Kangas for the colourful discussions and helpful comments during the years.

Finally, I warmly thank my parents, Seppo and Saara, as well as my brothers Mikko and Juha. Your steady support during my study and research has been invaluable.

The majority of this thesis work has been funded by the postgraduate program Graduate School in Chemical Engineering (GSCE). Financial support given by the Tauno Tönning and the Emil Aaltonen Foundations is also gratefully acknowledged.

Oulu, November 2010 Ilkka Malinen

List of symbols and abbreviations

a	Parameter
A	Coefficient
$\mathbf{A}$	Weighting matrix
b	Domain boundary
$\mathbf{b}$	Vector of domain boundary
B	Coefficient
$\mathbf{B}$	Jacobian matrix approximation
$\dot{B}$	Bottom product flow, mol/s
C	Coefficient
D	Coefficient
$\dot{D}$	Distillate flow, mol/s
$\mathbf{e}$	Vector where every element has value one
E	Coefficient
f	Function
$\mathbf{f}$	Set of problem equations
$\mathbf{f}'$	Jacobian matrix of $\mathbf{f}$
F	Coefficient
$\mathbf{g}$	Auxiliary function
h	Molar enthalpy of liquid flow, J/mol
$\mathbf{h}$	Homotopy function
H	Molar enthalpy of vapour flow, J/mol
$\mathbf{I}$	Diagonal identity matrix
K	Phase equilibrium value
$\mathbf{K}$	Occurrence matrix
l	Lower inner boundary
$\mathbf{l}$	Vector of lower inner boundary
$\dot{L}$	Liquid flow, mol/s
M	Scaling parameter
n	Dimension
nc	Number of components
N	Number of corrector iterations
p	Pressure, Pa
$\mathbf{p}$	Newton step
$\mathbf{q}$	Auxiliary vector
$\dot{Q}$	Heat flow, J/s

R	Reflux ratio or universal gas constant
s	Arc length
S	Reboil ratio
T	Temperature, K
u	Upper inner boundary
$\mathbf{u}$	Unit tangent vector or vector of upper inner boundary
$\dot{V}$	Vapour flow, mol/s
$\mathbf{W}$	Diagonal weighting matrix
x	Problem variable or liquid phase mole fraction
$\mathbf{x}$	Vector of variables
$\hat{\mathbf{x}}$	Mapped vector of $\mathbf{x}$
y	Vapour phase mole fraction
$\mathbf{z}$	Tangent vector

Greek Letters

α	Parameter
γ	Activity coefficient
δ	Relative measure for bounding zone width
$\partial\Omega$	Subset boundary
Δ	Increment or step length
ε	Error tolerance
θ	Homotopy parameter
λ	Step length or energy of interaction in the Wilson formulation
Λ	Binary interaction parameter
ν	Molar volume
ξ	Auxiliary variable
$\boldsymbol{\pi}$	Penalty function
$\boldsymbol{\Pi}$	Penalty matrix
ρ	Function
$\boldsymbol{\upsilon}$	Auxiliary function
ω	Non-negative scalar step factor
Ω	Subset

Subscripts

b	Bounded
i	ith component or ith variable
j	jth equilibrium stage or jth component

k	kth component
n	nth variable
N	Reboiler stage
opt	Optimum
t	tth element
unscaled	Unscaled
θ	Homotopy parameter

Superscripts

b	Bounded
F	Feed
i	ith corrector step
inf	Infinity
k	kth point on the homotopy path or kth iteration round
L	Liquid phase
max	Maximum
min	Minimum
mod	Modified
p	Number of sub-problems inside the homotopy parameter interval [0 1] in discrete homotopy path tracking
sat	Saturated
S	Side draw
T	Transpose
V	Vapour phase
0	Starting point
*	Solution
'	Boundary

Abbreviations

ASPENPlus	AspenPlus is a process modelling tool in AspenTech's aspenONE® Process Engineering applications
BP	Bubble Point Method
CFD	Computational Fluid Dynamics
CHEMCAD	CHEMCAD is a chemical process engineering software by Chemstations™
EQ	Equilibrium

HYSYS	AspenHYSYS is a process modelling tool in AspenTech's aspenONE® Engineering applications
IO	Inside-Out Method
IVP	Initial Value Problem
MATLAB	MATLAB® is a commercial product package for computing launched by MathWorks™
MESH	Set of equations used to mathematically describe the equilibrium stage
NAE	Non-Linear Algebraic Equation
NEQ	Non-Equilibrium
ODE	Ordinary Differential Equation
PDE	Partial Differential Equation
PRO/II	PRO/II® is a process simulation software by Invensys™
SC	Simultaneous Convergence
SR	Sum Rates Method

List of original papers

This thesis is based on the following publications, which are referred to in the text by their Roman numerals:

I Malinen I & Tanskanen J (2008) Modified bounded homotopies to enable a narrow bounding zone. Chemical Engineering Science 63(13): 3419–3430

II Malinen I & Tanskanen J (2010) Homotopy parameter bounding in increasing the robustness of homotopy continuation methods in multiplicity studies. Computers & Chemical Engineering 34(11): 1761–1774

III Malinen I & Tanskanen J (2007) Modified bounded Newton homotopy method in solving sidestream column configurations. In: Plesu V & Agachi PS (Eds.) Proceeding of 17th European Symposium on Computer Aided Process Engineering (ESCAPE–17), May 27–30, Bucharest, Romania. CD-ROM

IV Malinen I & Tanskanen J (2007) A rigorous minimum energy calculation method for a fully thermally coupled distillation system. Chemical Engineering Research and Design 85(A4): 502–509

V Malinen I & Tanskanen J (2009) Thermally coupled side-column configurations enabling distillation boundary crossing. 1. An overview and a solving procedure. Industrial & Engineering Chemistry Research 48(13): 6387–6404

VI Malinen I & Tanskanen J (2009) Thermally coupled side-column configurations enabling distillation boundary crossing. 2. Effects of intermediate heat exchangers. Industrial & Engineering Chemistry Research 48(13): 6372–6386

The manuscripts for the publications were written by the author of this thesis.

Contents

1 Introduction

> *"There is a simple explanation for the dearth (and death might be a more appropriate word here) of purely algorithmic articles: The methods that have been developed and that now are widely available in simulators are capable of solving the great majority of simulation problems. In addition, computers are significantly faster than they were and devising methods to save tiny fractions of a second no longer should be an adequate reason for developing new computer-based solution methods. Reliability (actually getting an answer) is far more important."* (Taylor 2007)

Process design can be understood as an iterative activity that aims to create and optimize both process structure and state distribution so that the criteria set for process performance can be met. In addition to product purity requirements, the process must fulfil the requirements set for safety, health and environmental issues, as well as being as cost-effective as possible.

Basically, alternative process structures can be generated with a combination of creativity and accumulated knowledge (heuristic guidelines). However, the state distribution for the process cannot be determined without expensive and time-consuming pilot testing, or preferably examinations based on mathematical process models. Since process design is strongly iterative by its basic nature, it is highly justified to utilize process simulation instead of piloting when examining the state distribution of the process. In addition, because of the tendency to tighten time schedules and decrease total costs, interest in exploiting process modelling and simulation is certain to increase, rather than decrease, in the future.

Frequently, two questions are faced that encapsulate almost everything that is challenging in process simulation:

- How should the system be specified to fulfil the performance criteria set?
- How can the solution actually be obtained?

The first of these questions is highly dependent on the experience and knowledge of the engineer faced with the simulation task. However, the characteristic of the second is more or less numerical, and even though the solving can be substantially alleviated with proper specifications and the selection of an appropriate solving strategy, obtaining the solution may still be a challenging and tedious task.

Even though the solving algorithms in commercial process simulation packages are very versatile nowadays, enabling the carrying out of a large variety of tasks, simulation itself is still very much the same as 30 years ago. Locally convergent methods are still widely utilized, enabling convergence from an initial guess towards the most attractive root at best. Therefore, not only is the determination of multiple solutions challenging, but approaching even one reasonable solution may also be problematic.

The challenges in solving could largely be tackled by utilizing homotopy continuation methods, which have a significantly larger convergence domain compared to locally convergent solving methods. (e.g. Wayburn & Seader 1987, Lin *et al*. 1987, Gritton *et al*. 2001) However, homotopy continuation methods have not been adopted into wider use. Obviously, homotopy continuation methods are not known so well and they are also thought to be complex and troublesome. In addition, the *robustness* of traditional homotopy continuation methods is not superior enough (Christiansen *et al*. 1996, Paloschi 1995, 1996, Wayburn & Seader 1987).

1.1 Robustness in process simulation

As noted for example in Bogle (1983), an algorithm that is utilized in chemical engineering should be *robust*, that is, capable of finding a solution from a wide range of initial guesses, and *efficient*, both in execution time and storage. Algorithm performance is usually judged based on efficiency or how much central processor time is required for convergence. However, poor robustness is a more likely reason for total failure.

Bogle & Perkins (1988) evaluated solving methods and aspects of their implementation with robustness as the first priority, and efficiency, the speed with which the solution is found, second. It was concluded that in engineering applications it is important that a solution is found (the code is robust) while efficiency may be only a bonus. According to Paloschi (1998), the robustness of the solver is especially important in the early stages of the simulation when a good initial guess is not available.

Robustness is often simply seen as a property of the solving method. However, it may be impossible to establish a solving method that would be absolutely robust, i.e. always succeeding in approaching the solution from an arbitrary initial guess. Therefore, it is unfair to focus solely on the robustness of

the solving method and it is well justified to consider robustness more widely as an aggregate of several aspects.

The general view of robustness adopted in this thesis is illustrated in Figure 1. In addition to the robustness of the *solving method*, the robustness of the applied *solving strategy*, that is, how the solution is sought through the solving methods, is thought to be a solid element of overall robustness. In fact, by utilizing a feasible solving strategy the poor robustness of problem-independent solving methods can be at least partly compensated. Problem-tailored solving procedures are highlighted in this thesis as a way to improve the overall robustness in process simulation. In addition, since the selected *solving approach* has an effect on robustness and on feasible solving strategy alternatives, it is justified to include it also in the overall view of robustness.

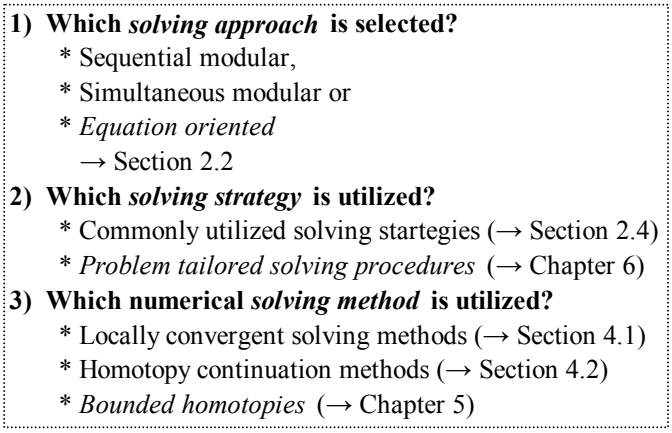

Fig. 1. The elements affecting overall robustness in process simulation. References to the Chapters and Sections where these issues are considered are marked by '→'.

As the result of the above reasoning, the following definition has been adopted in this thesis:

> *Robustness is a fundamental property that describes the ability to determine one or more solutions for a non-linear equation set representing the state model of a process. In addition to the robustness of the solving method, the robustness of the utilized solving strategy must be taken into account when considering the overall robustness of a solving approach.*

It is worth noting that all the aspects presented in Figure 1 can basically be assembled in the form of a solving algorithm. The solving algorithm is either

robust or not, i.e. the algorithm either obtains the solution for the equation set from an initial guess or it does not. By offering various starting points the level of robustness could be quantified. However, this kind of statistical approach would basically require an infinite number of starting points to be examined. In simple cases, the level of robustness between algorithms could be compared by analytically determining the extents of their convergence domains. However, this approach cannot be easily extended into wider systems.

In addition to the overall robustness, the overall efficiency of the algorithm is admittedly important. Numerical tools and routines reducing the number of numerical operations and computing machine memory requirement, for example by exploiting the problem sparsity, are essential when aiming to improve the efficiency of the solving algorithm.

1.2 Purpose of the work

Briefly stated, the purpose of this thesis is to consider alternatives for improving the overall robustness of an equation-oriented solving approach. In addition to advanced numerical problem-independent solving methods in the form of modified bounded homotopies, it is considered that overall robustness can be enhanced by applying problem-tailored solving procedures. Since the interest in chemical engineering seems to be in implementing process systems that have more and more complex characteristics, the utilization of advanced solving methods and problem-tailored solving procedures are thus considered highly necessary. This is relevant for example when examining complex distillation configurations.

At the beginning of the thesis, the role of simulation in chemical engineering is considered in general in Chapter 2. In particular, various strategies for solving process configurations are represented. To give background for the distillation systems examined in Papers I–VI, an overview of distillation modelling based on MESH equations and the properties of various distillation configurations is given in Chapter 3.

In Chapter 4, problem-independent solving methods applicable in an equation-oriented solving approach are presented. In addition to locally convergent Newton-Raphson based solving methods, homotopy continuation methods possessing a wider domain of convergence are examined.

Because of the insufficient robustness of traditional homotopy continuation methods, bounded homotopies are considered in Chapter 5 as a way to improve

robustness. The modifications presented in Papers I and II are especially highlighted in order to make the bounded homotopies originally proposed by Paloschi (1995, 1996, 1998) more applicable in chemical engineering.

In Chapter 6, problem-tailored solving procedures are presented as a way to improve overall robustness in comparison with more traditional solving strategies. The problem-tailored solving procedures presented herein have been developed in stages in Papers III–VI. They have been applied in minimum energy examinations of a fully thermally coupled distillation system (Paper IV) and thermally coupled side-column configurations (Papers V–VI). The observations and results gathered from Papers I–VI are discussed in Chapter 7. Finally, in Chapter 8, conclusions are drawn and objectives are suggested for further studies.

2 Process simulation in chemical engineering

"The trend in computer-aided chemical process design and in many other important applications is to embed numerical methods in large, user-friendly programs or, possibly, expert systems. Failure of a non-linear equation solver buried deeply within such a system would be catastrophic because of the waste of costly calculations completed before the failure occurred and because of the waste of even more costly manpower as the user waited for expert help to arrive. For such a system, we need a sequence of methods of monotonically increasing reliability (and operating cost) culminating in a single method (or family of methods) capable of solving all but the most pathological problems. Whenever more economical methods fail, the final method in the sequence could be employed as a method of last resort. We propose that homotopy continuation methods be considered for this 'method of last resort'." (Wayburn & Seader 1987)

2.1 Chemical engineering models and problem solving

Models in chemical engineering typically consist of material and energy balances, equations for phase equilibria, transport equations, and chemical kinetic expressions (Seider *et al.* 1991). These equations may include linear algebraic, non-linear algebraic (NAE) and transcendental, ordinary differential (ODE), partial differential (PDE), integral equations, and combinations of them (Seader *et al.* 1990). Often, both temporal and spatial derivatives exist. The spatial derivatives are usually approximated with finite differences or trial functions in the form of polynomials. When the process is considered as steady-state, temporal derivatives are set to zero. (Seider *et al.* 1991) Thus, the process model is reduced to a set of non-linear equations:

$$\mathbf{f}(\mathbf{x}) = \mathbf{0}. \tag{1}$$

In addition to steady-state examinations, solving this equation set is required in the initialization of dynamic simulation (Paloschi 1998).

As Shacham has listed in Westerberg & Chien (1984), solving a non-linear equation set is often difficult because:

- some of the functions are undefined for certain values of the variables,
- there are several solutions to the system, not all of them physically feasible,

- the functions are very non-linear and badly scaled, and
- the system is very large and sparse.

Although mass balance relations are often linear, energy balances and phase equilibria equations require thermo-physical property correlations, which are in general non-linear. In addition to nonlinearity, thermo-physical property correlations usually involve terms with discontinuities in the functions and their derivatives. (Bogle 1983) In some cases the solving of Eq. (1) may be facilitated at least somewhat by reducing functions to a simpler form, for example by eliminating variables from the denominators thus making the functions continuous and better behaving (Gritton *et al.* 2001).

Since process models in chemical engineering usually involve at least one non-linear equation, the whole equation system becomes non-linear. In this case, multiple solutions may exist. However, some of the solutions may not be physically realistic. For example, a solution with a negative mole fraction or flow rate value is not feasible.

In addition to the actual solution, unfeasible variable values may also be encountered during solving. In this case, thermodynamic subroutines easily fail if properties outside their validity range are evaluated. Therefore, it would be important to constrain variables within the valid domain. However, problems may still be encountered when variables lie on the domain boundaries or close to them. (Bogle 1983)

Equation systems are often both large and sparse. Sparsity means that each equation involves only a small fraction of the problem variables of the overall equation set. (Bogle 1983) On the whole, it is evident that solving mathematical relations that describe chemical processes and even single process units cannot be done without a computer.

The computers being launched on the markets are continually smaller, faster, with larger memory facilities, but still cheaper than previously. The huge development in computer technology has strongly affected working practices. Computer aided tools have become a standard feature in almost all aspects of chemical engineering, and methods for both steady-state and dynamic simulation have been evolved (Christiansen *et al.* 1996). Several commercial process simulation packages as well as proprietary (in-house) programs have been developed for process design purposes.

2.2 Approaches to solving flowsheeting problems

The use of computer aids to perform steady-state heat and material balancing, sizing and costing calculations for a chemical process is called *flowsheeting* (Westerberg 1979). A *flowsheeting package* can be understood as a comprehensive and flexible tool for modelling a system of process units. In general, a flowsheeting package has libraries for commonly used process units and subroutines for calculating thermo-physical properties for common substances. If these libraries are not comprehensive enough, the users may add their own unit models and information of non-standard components. (Bogle 1983)

To solve flowsheeting problems, three kinds of approaches exist (Bogle 1983, Perkins in Westerberg & Chien 1984):

- sequential modular,
- simultaneous modular, and
- equation-oriented.

A sequential modular approach must have subroutines for each unit, and each unit may be solved only if all inputs and parameters are known. The calculation is iterative and the order of calculation is the same as the process flow. Managing recycle streams that do not have calculated numerical values requires the user to guess the values. If the values after calling each subroutine in flow turn are not the same as the values guessed, the guessed values are updated. The iteration is continued until the required error tolerance is fulfilled. (Bogle 1983)

The equation-oriented approach, which is also known as simultaneous, equation-based, and even global strategy (Westerberg & Chien 1984), is the major competing approach to the sequential modular approach. All the equations describing the process are gathered together and solved simultaneously with a general-purpose multi-dimensional root finding algorithm (Barton 2000). When the process model includes heat integration and recycle flows, as is often the case for example in distillation systems, the equation-oriented solving approach is more flexible than the sequential modular approach.

The simultaneous modular approach is some kind of combination of the two main approaches. Subroutines of the sequential modular approach are meant to be utilized as in the equation-oriented approach.

2.3 Role of simulation in chemical engineering

Process simulation can be utilized in searching and testing new process ideas, and comparing process alternatives. Simulation can also be utilized in examining and optimizing existing processes. Simulation packages can accelerate the completion of process design projects and possibly offer more rigorous results (Seider *et al.* 2004). These packages are also relatively easy to use.

Since testing with laboratory- and pilot-scale equipment is not only expensive and time-consuming, but may also include safety risks, simulation is an excellent alternative for developing and maturing novel process ideas to the level of implementation. Currently, the trend is to utilize the flexibility offered by simulation as much as possible and to resort to pilot tests only when it is absolutely necessary.

The advanced models utilized nowadays to describe the chemical and thermo-physical properties of components accurately make it possible to exploit cumulated knowledge more innovatively in process design. The reduction of raw material and energy consumption and the minimizing or prevention of the formation of hazardous by-products and emissions regularly require new kinds of processes, which are often more complex than the current ones. However, structurally as well as operationally complex systems with detailed thermodynamics make process simulation a challenging task.

Generally, problem solving should be fast and, because of the high cost of human contribution, require as little manual simulation activity as possible. Simulation is, however, seldom a routine kind of work, and more often demands creativity and a problem-solving-oriented attitude. Therefore, it is not viable to completely replace humans. However, it is worth aiming to minimize manual work with the aid of solving algorithms, which exploit sophisticated solving methods and approved solving strategies. Thus the human contribution could be focused on tasks that cannot be carried out by a machine, and which definitely require experience and knowledge-based innovativeness.

2.4 Various solving strategies

Basically, the aim is to solve process systems with exact product purity specifications. The solving strategy illustrated in Figure 2 is extremely useful from the process design and simulation aspects. It is, however, numerically

extremely challenging to obtain a converged result directly when the structure of the system or the component mixture under consideration is complex.

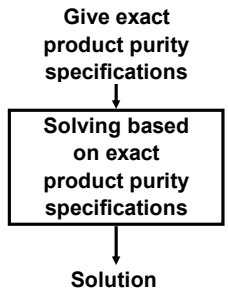

Fig. 2. Solving strategy where the system is solved directly based on product purity specifications.

In order to facilitate solving, in distillation simulation the solution is often sought based on the strategy illustrated in Figure 3. In this case, the solution is sought based on sub-problems, which help to approach the solution in a step-by-step manner. Flow ratios can be utilized as simple specifications in the sub-problems. The technique where one solution is used as the initial value for the subsequent calculation is sometimes known as 'sneaking up on an answer' (Kister 1992).

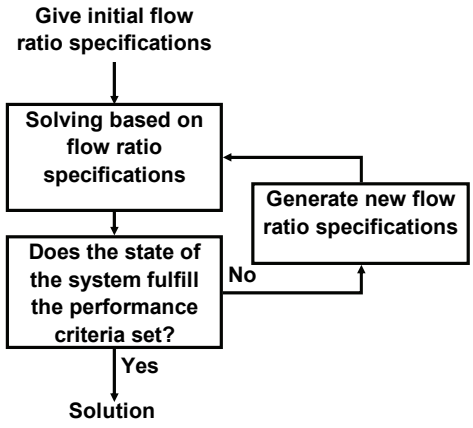

Fig. 3. Solving strategy where a sequence of sub-problems with flow ratio specifications is solved to approach the required product flow purities.

The problem is that the strategy illustrated in Figure 3 does not have a well-constructed procedural base. It may take several rounds to approach the purity criteria and thus require a considerable amount of human contribution and time.

In addition, this kind of solving strategy is highly dependent on the knowledge and experience of the engineer carrying out the simulation task.

To approach the desired product flow purities more straightforwardly, the strategy illustrated in Figure 4 is often applied. In this strategy, the system is first solved with flow ratio specifications. The idea is to generate an initial state distribution for the actual solving phase, where exact product purity specifications are utilized. If solving with exact purity specifications is not successful, the system is solved with new flow ratio specifications in order to obtain a more favourable initial state distribution.

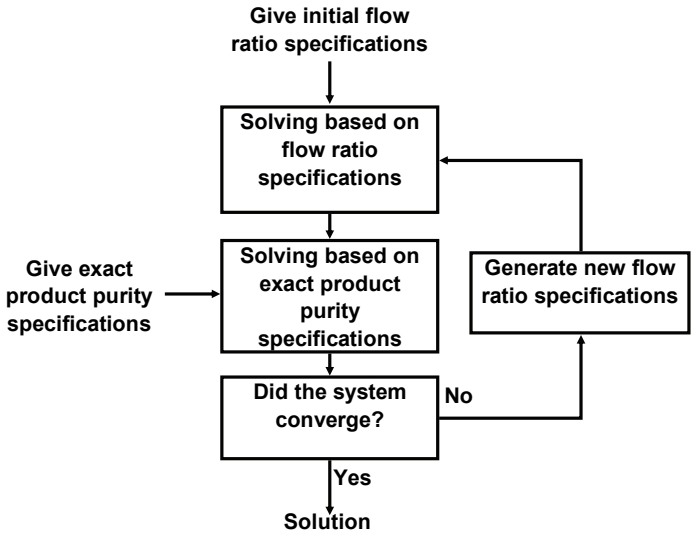

Fig. 4. Solving strategy, where the system is solved first with flow ratio specifications in order to obtain a favourable initial state distribution for the actual solving phase with exact product purity specifications. If a converged result is not obtained, a new initial state distribution is generated.

The strategy in Figure 4 results essentially in a state distribution that fulfils the purity criteria set for the process exactly. Therefore, in this thesis, the solving strategy illustrated in Figure 4 is preferred to the strategy illustrated in Figure 3.

3 Modelling and solving aspects of distillation

> *"The design of complex distillation column configurations is an industrially relevant problem. Theoretical work under the assumption of minimum reflux has shown that these configurations can save up to 50% more utility than conventional column arrays. However, the complex column configurations that can potentially produce larger energy savings, such as the fully thermally coupled columns (Petlyuk), are not commonly used in industrial practice, largely because of control concerns. Another reason complex columns are not widely used is the fact that the design of such systems involves strong interactions, and requires detailed simulation models that are often difficult to converge."* (Yeomans & Grossmann 2000)

Distillation is usually modelled with what are termed MESH equations. MESH equations are a mathematical formulation of material and heat balances as well as the phase equilibria prevailing in a single equilibrium stage. The equation set for a distillation column is obtained by aggregating equation sets describing single distillation column stages. In the same way, models for distillation column sequences and various column configurations may be formulated based on MESH equations.

In addition to MESH equations, thermodynamic correlations for individual components as well as component mixtures are required in distillation modelling. The variety of thermodynamic models and their field of application is wide. The selection between the alternatives is made based on the properties of the component mixture under consideration. Simple and idealized correlations may give reasonable results for zeotropic systems. However, azeotropic systems require thermodynamic models, which are capable of describing the non-ideal behaviour of the component system.

In general, equation sets describing distillation configurations are large and sparse. As a result of the close material and heat interactions between column stages, as well as the thermal couplings between columns, equation sets describing distillation systems can seldom be decomposed into smaller equation sets. In addition, both the complex structure of column configurations and the complex thermodynamic behaviour of component mixtures may induce multiple solutions. The complex characteristics of distillation systems make solving an interesting, but challenging task. As a whole, distillation systems offer an

excellent domain for examining the various solving methods and strategies that have been developed for chemical engineering purposes.

3.1 Distillation modelling based on MESH equations

According to the notation of Figure 5, the MESH equations for a homogeneous unreactive equilibrium stage applicable for multi-component separation can be formulated as follows:

The overall material balance:

$$\dot{L}_{j-1} + \dot{V}_{j+1} - (\dot{L}_j + \dot{L}_j^S) - (\dot{V}_j + \dot{V}_j^S) + \dot{L}_j^F + \dot{V}_j^F = 0\,. \tag{2}$$

The material balances for the components ($i = 1 \ldots nc-1$):

$$\begin{aligned} &x_{i,j-1}\dot{L}_{j-1} + y_{i,j+1}\dot{V}_{j+1} - x_{i,j}(\dot{L}_j + \dot{L}_j^S) - y_{i,j}(\dot{V}_j + \dot{V}_j^S) \\ &+ x_{i,j}^F \dot{L}_j^F + y_{i,j}^F \dot{V}_j^F = 0. \end{aligned} \tag{3}$$

The phase equilibrium relations ($i = 1 \ldots nc$):

$$y_{i,j} - K_{i,j}x_{i,j} = 0\,. \tag{4}$$

Summation equations for mole fractions:

$$\sum_{i=1}^{nc} x_{i,j} - 1 = 0 \tag{5}$$

$$\sum_{i=1}^{nc} y_{i,j} - 1 = 0\,. \tag{6}$$

The heat balance:

$$\begin{aligned} &\dot{L}_{j-1}h_{j-1} + \dot{V}_{j+1}H_{j+1} - (\dot{L}_j + \dot{L}_j^S)h_j - (\dot{V}_j + \dot{V}_j^S)H_j \\ &+ \dot{L}_j^F h_j^F + \dot{V}_j^F H_j^F + \dot{Q}_j = 0. \end{aligned} \tag{7}$$

Alternatively, instead of two summation equations for mole fractions and $nc-1$ material balances for components, a summation equation for mole fractions:

$$\sum_{i=1}^{nc} x_{i,j} - \sum_{i=1}^{nc} y_{i,j} = 0 \tag{8}$$

and nc material balances for components can be utilized. Doing this does not change the total number of MESH equations.

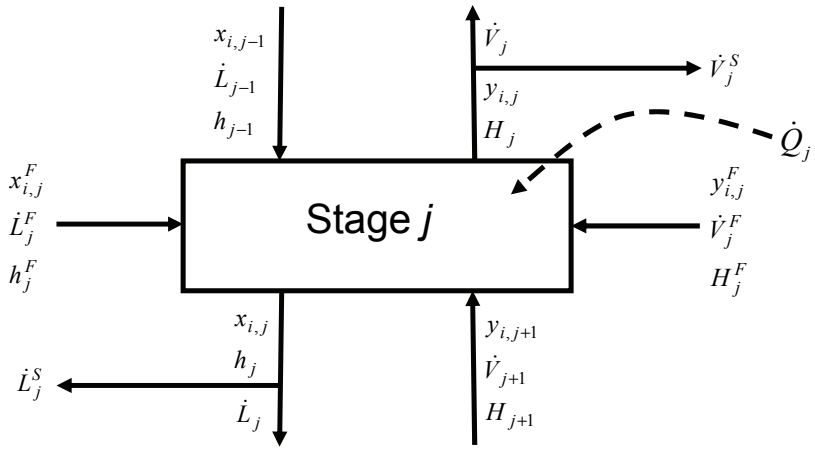

Fig. 5. A homogeneous unreactive equilibrium stage.

The MESH equations can also be utilized when modelling reboilers and condensers. Reboilers and condensers are often considered in terms of reflux and reboil ratios. The relations can be formulated as follows:

Relation for the reflux ratio R:

$$\dot{L}_1 - (\dot{D}_1^L + \dot{D}_1^V)\,R = 0\,. \tag{9}$$

Relation for the reboil ratio S:

$$\dot{V}_N - (\dot{B}_N^L)\,S = 0\,. \tag{10}$$

Depending on the simulation case in question, the condenser (stage 1) is usually considered as either a total condenser or a partial condenser. Thus, either the dew point vapour $\dot{D}_1^V$ or the bubble point liquid $\dot{D}_1^L$ distillate flow is set to zero.

All material balances and summation equations include only problem variables. In heat balance and phase equilibrium relations there are also terms, which are not actual variables of the problem. These are the phase equilibrium value $K_{i,j}$, and the enthalpies for liquid h_j and vapour H_j. These terms are obtained from thermodynamic correlations, where mole fractions, pressure and temperature may exist as variables. These correlations are usually non-linear, thus posing a challenge to the solving of the MESH equation system.

The MESH equations represented above are in simple form and represent the column stage at steady-state. In this case the vapour and liquid streams leaving

the stage are assumed to be in equilibrium (EQ). Even though it is possible to model distillation processes with non-equilibrium (NEQ) or rate-based models, or even detailed computational fluid dynamics (CFD) (Taylor 2007), the equations presented above are suitable for both conceptual and detailed distillation design.

3.2 Complexity of distillation models

Complexity appears in distillation in several ways. Distillation columns and column configurations may be complex in structure. In addition, non-ideal thermodynamics complicate distillation design and model solving.

3.2.1 Complexity caused by the structure of distillation system

As illustrated in Figure 6a, a simple distillation column is a conventional column having only one feed, two product flows, one condenser and one reboiler. The distillate is obtained from the top and the bottom product from the bottom of the column. All the other types of distillation columns are referred to as complex columns. Complex columns may have more than one feed, or have one or more side product flows, intermediate reboilers, intermediate condensers or pumparounds. (Doherty & Malone 2001, Kister 1992)

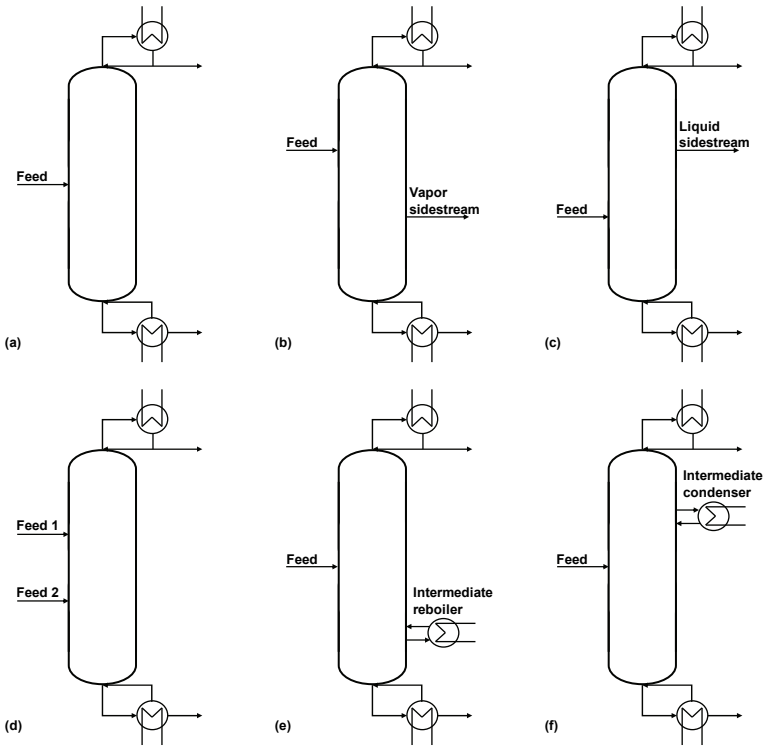

Fig. 6. A simple column and some complex distillation columns. (a) A simple column, (b) a vapour sidestream column, (c) a liquid sidestream column, (d) a column with two feed flows, (e) a column having an intermediate reboiler below the feed and (f) a column having an intermediate condenser above the feed.

Column sequences are created by linking single distillation columns. This is done by routing the product flow of one distillation column into the feed stage of another column. This is how the conventional direct and indirect column sequences illustrated in Figure 7 are comprised. The prefractionator scheme illustrated in Figure 7c represents a more novel alternative, where both the distillate and bottom products of a distillation column are routed as one-way connections to the feed stages of a side draw column (e.g. Doherty & Malone 2001, King 1980).

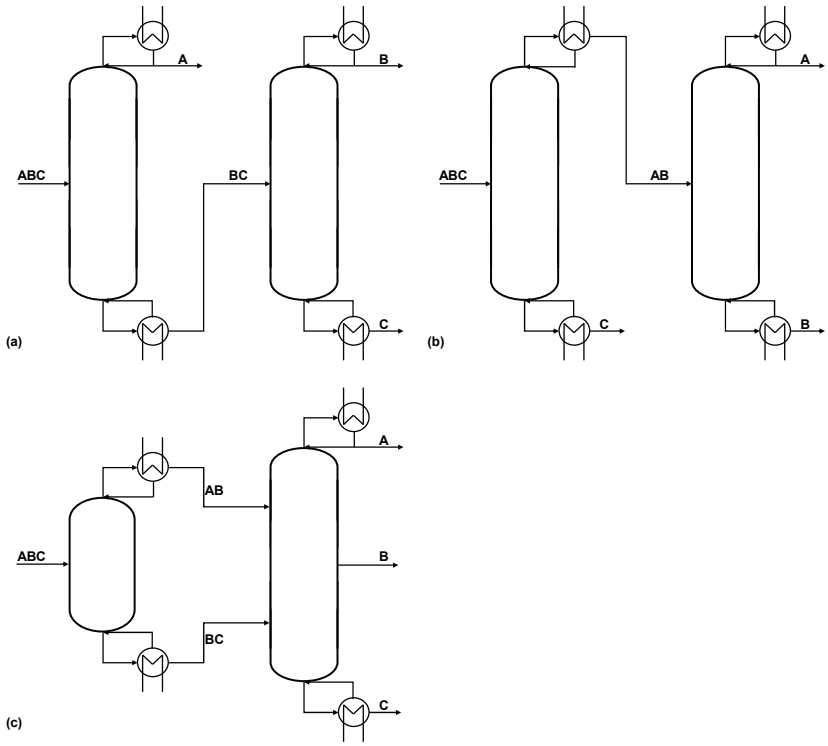

Fig. 7. Conventional column schemes for ternary distillation. (a) Direct, (b) indirect and (c) prefractionator schemes.

The column sequences having only one-way connections and no recycle flows from subsequent columns back to the previous ones are workable for the separation of zeotropic mixtures. However, when the complexity of the component mixtures increases, recycle flows become necessary. Classical extractive distillation with a heavy entrainer introducing no new azeotropes and a distillation sequence exploiting a curved distillation boundary are examples of homogeneous azeotropic distillations, where recycle flows are essential (Figure 8) (e.g. Doherty & Malone 2001, Rooks *et al*. 1996, Wahnschafft *et al*. 1992, 1993).

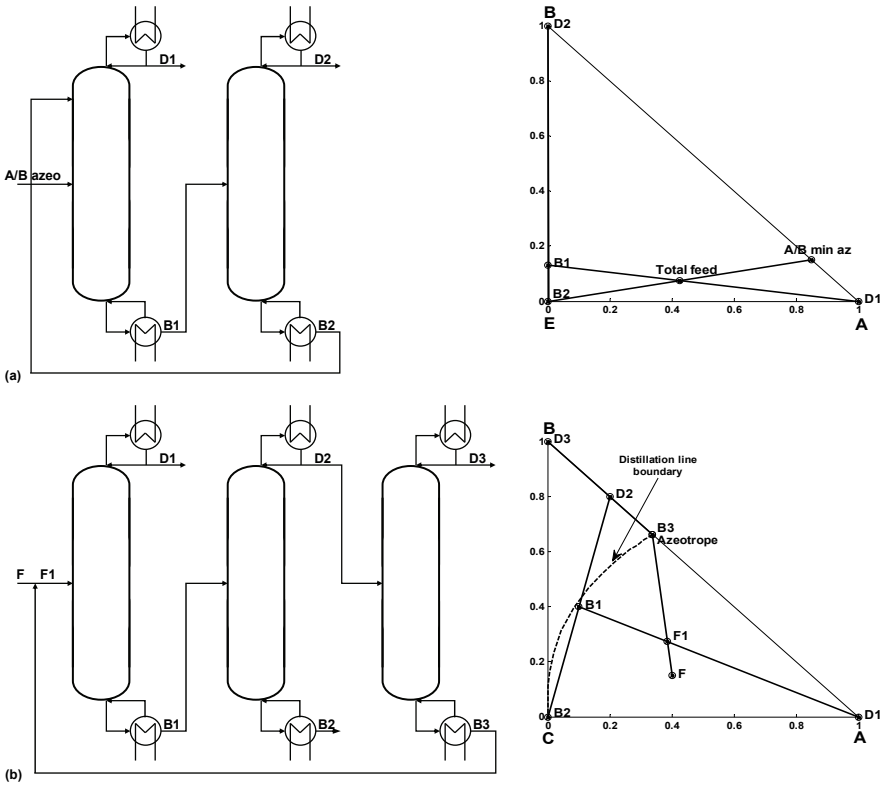

Fig. 8. Examples of homogeneous azeotropic distillations having a recycle flow. (a) Extractive distillation with a heavy entrainer and (b) a distillation sequence exploiting a curved distillation boundary.

Recycle flows complicate not only the structure and operation of distillation configurations, but also design of such systems. Recycle flows also make the solving of the column system model more challenging. In this case, the modular solving approach is no longer as competitive as it is when solving conventional schemes.

The unsuitability of the modular solving approach becomes apparent when there are two-way connections in the column configuration. Two-way connections are characteristic of the thermally coupled column configurations shown in Figure 9 (see extensive list of references in Papers IV–VI). The strong interaction between distillation columns favours the equation-oriented solving approach, in which case the equations describing the column configuration are solved as an aggregate set.

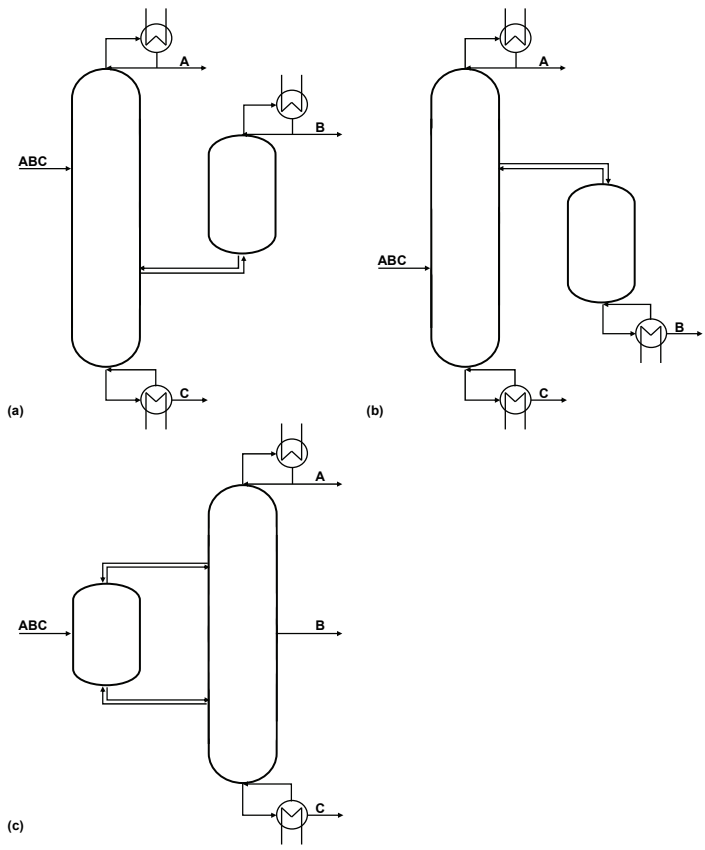

Fig. 9. Thermally coupled distillation column configurations for ternary distillation. Thermally coupled (a) side-rectifier and (b) side-stripper, and (c) fully thermally coupled (the Petlyuk) column configuration.

On the whole, the possibility of obtaining high product flow purities with diminished energy consumption makes thermally coupled and heat integrated distillation systems very attractive. However, the complexity of these systems makes the solving more demanding. From the research and development aspect as well as the perspective of detailed process design, robust solving methods and strategies are therefore desirable.

3.2.2 Complexity caused by thermodynamics

The mathematical correlations describing the thermodynamic properties of single components as well as component systems are usually non-linear. Nonlinearity is

essential, making it possible to describe the complex thermodynamic properties of component systems accurately.

One example of nonlinearity is the modified Antoine equation, which represents the dependence of a pure component-saturated vapour pressure p^{sat} on the temperature as follows:

$$\ln\left(p^{sat}\right) = A + \frac{B}{T+C} + D\ln(T) + E\left[T^{F}\right]. \tag{11}$$

The coefficients *A*, *B*, *C*, *D*, *E* and *F* are obtained by regressing experimental data. (HYSYS manual)

Another example of the strong non-linear character of thermodynamic relations is the Wilson formulation of liquid phase activity coefficient γ for component *i*:

$$\ln \gamma_i^L = 1 - \ln\left(\sum_{j=1}^{C} x_j \Lambda_{ij}\right) - \sum_{k=1}^{C}\left(\frac{x_k \Lambda_{ki}}{\sum_{j=1}^{C} x_j \Lambda_{kj}}\right), \tag{12}$$

where

$$\Lambda_{ij} = \frac{v_j^L}{v_i^L}\exp\left(-\frac{(\lambda_{ij} - \lambda_{ii})}{RT}\right)$$

$$\lambda_{ji} = \lambda_{ij}$$

$$\Lambda_{ji} \neq \Lambda_{ij}\,.$$

The term $(\lambda_{ij} - \lambda_{ii})$ is an empirically determined energy term, and v_i^L describes the molar volume of component *i* in the liquid phase. (Holland 1981, Poling *et al*. 2001)

In general, variable values substituted into thermodynamic correlations must be positive. If negative values are substituted into the logarithm or square root functions, complex numbers are obtained. Even though complex numbers are mathematically feasible, they are regularly not tolerated in the thermodynamic subroutines utilized in process simulation packages. In addition, the substitution of the pure zero value into a logarithm function poses a fatal error, in which case the solving is interrupted.

Complex thermodynamics also poses challenges in the form of azeotropes and distillation boundaries. Distillation regions divided by distillation boundaries (separatrices) is an issue that must be taken into account in process design and simulation (e.g. Fidkowski *et al*. 1993, Krolikowski 2006, Stichlmair & Herguijuela 1992, Thong & Jobson 2001, Wahnschafft *et al*. 1992, 1993).

3.3 Distillation multiplicities

Since the thermodynamic behaviour of component systems is frequently non-linear, the existence of multiple steady-states is highly probable in distillation (see extensive list of references in Paper II). In addition, complex column configurations can result in multiple steady-states (e.g. Chavez *et al*. 1986, Lin *et al*. 1987).

In order to succeed in process operation, and to prevent inappropriate process design and the non-consideration of attractive process alternatives, it is important to determine multiple solutions when they exist. This is not an easy task with locally convergent solving methods, which at best are able to find only one solution from an initial guess. Thus, the existence of multiple solutions is easily missed without several simulation runs altering the initial guess. Even then, since the region of attraction of solutions may be small, a random search may not locate every solution. (e.g. Bekiaris & Morari 1996, Dalal & Malik 2003, Seider *et al*. 1991, Vadapalli & Seader 2001)

∞/∞ analysis has been utilized to generate favourable initial profiles for attaining multiple steady-state solutions. Also, dynamic simulation with stabilizing control has been utilized to overcome the shortcomings of conventional solvers. (e.g. Bekiaris & Morari 1996, Kannan *et al*. 2005)

Since homotopy continuation methods are basically able to approach several solutions from a single starting point, they are therefore appreciated in multiplicity studies.

3.4 Various alternatives for solving distillation models

Several classes of methods for numerically solving MESH equations can be identified (Perry 2008):

1. Tearing methods
2. Inside-out (IO) methods

3. Simultaneous convergence (SC) methods
4. Relaxation methods
5. Continuation methods
6. Collocation methods
7. Optimization methods

The most widely utilized methods for solving MESH equations fall into categories 1–3 and 5. The inside-out (IO) methods have replaced the older tearing methods as the standard solving method. The inside-out methods and simultaneous convergence (SC) methods are available in many commercial process simulators such as ASPENPlus, HYSYS, PRO/II and CHEMCAD. Homotopy methods are finding increasing use in commercial software. (Seader & Henley 2006, Perry 2008)

Tearing methods

In *tearing methods,* a system of MESH equations is broken into small groups and each group is solved in a series of steps. When solving any subset of equations, only a corresponding number of variables, the so-called 'tear' variables, can be determined. To start the calculation, values for the remaining variables must be assumed. The 'torn' set of equations is then solved for the 'tear' variables. Consecutive groups of equations and variables are torn from the full set of equations and variables until all the variables have been updated. The procedure is repeated until all the equations are satisfied simultaneously. (Kister 1992, Perry 2008)

Several tearing (or decoupling) methods can be found in the literature, e.g. the bubble point (BP) method, the sum rates (SR) method, the θ method of convergence and the 2N Newton-Method. Gaussian elimination, also known as the Thomas algorithm, has broadly been utilized to solve a tridiagonal matrix system in these applications. (Henley & Seader1981, Holland 1981, Kister 1992, Perry 2008, Seader & Henley 2006)

Some tearing methods have a relatively limited range of application. For example, the BP methods are more successful for distillation and the SR methods for absorption and stripping (Henley & Seader 1981). One possible drawback of the tearing methods is that the number of times that physical properties must be evaluated per outer loop iteration may be high, thus increasing the solving time. Iteration loops may also be hard to converge. (Perry 2008) In addition, tearing

methods usually have a very limited set of specifications which can be utilized (Henley & Seader 1981, Seader & Henley 2006).

Inside-out methods

Even though *inside-out* (IO) methods actually belong to the group of tearing methods, they are nowadays so widely utilized in commercial simulation programs that it is worth presenting IO methods as a separate group of methods. In inside loop iterations, a simplified K-value and enthalpy correlations are utilized, where the iteration variables T, $\dot{V}$, x and y are relatively free of interactions with each other. Thus the inside loop works well for a wide range of mixtures and is little affected by the non-ideality of mixtures. The simplified thermodynamic correlations are frequently updated from rigorous thermodynamic expressions in the outside loop. By reducing the number of complex thermodynamic calculations, the simulation time is reduced. (Kister 1992, Perry 2008, Seader & Henley 2006)

Simultaneous correction methods

In *simultaneous correction* (SC) (or simultaneous convergence) methods, an attempt is made to solve all of the MESH equations simultaneously. Thus, separate inner and outer loops do not exist. Simultaneous convergence procedures are generally fast. They are also less sensitive to difficulties with non-ideal solutions than tearing methods are. (Henley & Seader 1981, Kister 1992, Perry 2008, Seader & Henley 2006) However, as mentioned in Section 4.1, the Newton-Raphson methods or variants require an initial guess close to the final solution to converge successfully.

Relaxation methods

The relaxation method uses at least one set of MESH equations (energy and component balances) in an unsteady-state form. The equations are integrated numerically from some initial state. At each time step, the equations are solved with the phase equilibria to obtain changes in stage temperatures, flow rates, and compositions. Even though the relaxation method is stable by always converging to the solution, the rate of convergence is slow and slows even further when the steady-state solution is approached. One option to speed up solving is to use the

relaxation method to bring the column state distribution near the solution and use Newton's method to obtain the final solution. (Henley & Seader 1981, Kister 1992, Perry 2008)

Homotopy continuation methods

The *homotopy continuation methods* can be classified in two basic categories: mathematical and physical (or parametric) homotopies. Mathematical homotopies place the MESH equations into a homotopy equation of a purely mathematical origin. In physical homotopies, the nature of MESH equations is exploited. (Kister 1992, Perry 2008) The homotopy continuation methods are considered in more depth in Section 4.2.

Design specifications

Distillation subroutines frequently offer a limited set of specifications to be utilized. A reflux ratio (or reflux rate) and product rate specifications are offered primarily. While these are the easiest specifications to solve, designers often prefer to set other specifications, such as purity specifications or recovery of some component. To accomplish this, simulators usually have a *design specification* facility. The variables and parameters that it is permitted to specify by means of a subroutine are utilized as floating variables to achieve the desired values for the desired column specifications. During the simulation calculations, a control subroutine compares the calculated values with the desired specifications and aims to manipulate floating variables so that the desired specifications are met. (Kister 1992, Seider *et al*. 2004)

Initial profile

Especially in cases where the distillation system is structurally complex and/or when component mixtures are highly non-ideal, the initial flow rates and temperature profiles have a significant role in contributing to the solving. If the *initial profile* is not favourable, the converged result may not be achieved.

Generally, programs permit the user to initialize the internal flow rates and temperatures. On the basis of a few initialized values, the program calculates the others by linear interpolation. In some systems, for example columns with purity

specifications and highly non-ideal systems, the temperature profile should be near the expected results. (Kister 1992)

4 Numerical methods for an equation-oriented solving approach

> *"It is ironic, particularly in light of the effort devoted to this problem in the 70's and 80's, that at the moment, the leading commercial products rely on a slight modification of a 17th century algorithm."* (Barton 2000)

Since all problem variables are treated equally in the equation-oriented solving approach, any problem variable may be utilized as a design specification. This feature makes the equation-oriented solving approach flexible and useful.

In general, the equation set $\mathbf{f}(\mathbf{x})$ must be square, that is, the number of elements in the vector $\mathbf{x}$ must be equal with the number of equations. This is identical with the requirement of degrees of freedom being zero. The equation system must also be non-singular and well posed. In the case of a badly posed problem the specifications cannot be satisfied by the acceptable set of unknown variables. (Bogle 1983, Bogle & Perkins 1988)

Since equation sets are often large and sparse, it is reasonable to decompose a large equation system into smaller equation sets. This is called partitioning (Westerberg 1979), and can be done by permuting the occurrence matrix until it is block triangular corresponding to the reordering of equations and variables (Bogle 1983). Methods applicable for approaching the block triangular form can be found for example in Duff *et al.* (1986).

Traditionally, Newton-Raphson based *locally convergent* methods are utilized in non-linear equation set solving. Local convergence means that for successful solving, an initial guess must be offered that is sufficiently close to the root. Even though the usability of locally convergent solving methods can be improved with a good initialization by utilizing a succession of approximate models to provide a guess in the vicinity of the solution (Seider *et al.* 1991), the poor robustness of locally convergent solving methods cannot be fully overcome. To increase the robustness, homotopy continuation methods with the property of more *global convergence* can be adopted.

4.1 Locally convergent solving methods

Locally convergent solving methods are well known, quite simple in implementation, and often work satisfactorily to obtain a root relatively quickly. In addition, they are very efficient when sufficient knowledge regarding the

location of a root is available (Gritton *et al.* 2001). Locally convergent methods are popular and widely utilized in various circumstances where non-linear equation sets are solved.

Even though the convergence domain may be enlarged somewhat, the generation of a suitable initial point can be as difficult as obtaining the actual root, especially when the functions are highly non-linear (Gritton *et al.* 2001). In addition, locally convergent solving methods are able to find only one root from one initial guess at best. Therefore, several initial guesses need to be generated when utilizing locally convergent solving methods in multiplicity studies.

Consideration of the locally convergent solving methods given below is based on Bogle (1983), Bogle & Perkins (1988), Dennis & Schnabel (1983), Kelley (1995), Westerberg (1979) and Shacham in Westerberg & Chien (1984).

4.1.1 The Newton-Raphson method

Newton-Raphson based solving methods rely on the sequence of iterations according to:

$$\mathbf{x}^{k+1} = \mathbf{x}^{k} - \left[\mathbf{f}'(\mathbf{x}^{k})\right]^{-1}\mathbf{f}(\mathbf{x}^{k}), \tag{13}$$

where $\mathbf{f}'(\mathbf{x}^{k})$ is known as a Jacobian matrix evaluated at $\mathbf{x}^{k}$.

The non-linear equation systems are often large and sparse. In addition, they are often unstructured. This means that the calculation of the next iterate $\mathbf{x}^{k+1}$ by utilizing the inverse Jacobian according to Eq. (13) becomes impractical. Since the inverse Jacobian tends to be full and calculation of the inverse requires a significant amount of computation work, the next iterate should be calculated based on a two-phase strategy:

$$\mathbf{f}'(\mathbf{x}^{k})\mathbf{p}^{k} = -\mathbf{f}(\mathbf{x}^{k}), \tag{14}$$

$$\mathbf{x}^{k+1} = \mathbf{x}^{k} + \alpha\mathbf{p}^{k}. \tag{15}$$

First, the Newton step $\mathbf{p}^{k}$ is obtained from Eq. (14) based on Gaussian elimination. Then, the next iterate $\mathbf{x}^{k+1}$ is calculated based on Eq. (15).

α in Eq. (15) is a suitable parameter $\alpha > 0$ that is utilized to improve the convergence property. A common practice is to try a full Newton step ($\alpha = 1$) first. If this step does not reduce the residual of the functions, the numerical value of parameter α is diminished until reduction is attained. This practice is a backtracking form of the line search method. Since the Newton step is a descent

direction for $\mathbf{f}$, the described practice guarantees a decrease in absolute function values.

4.1.2 Jacobian matrix determination

Fundamentally, the Jacobian matrix is defined with partial derivatives as:

$$\mathbf{f}'(\mathbf{x}) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & & \vdots \\ \vdots & & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}. \tag{16}$$

Each row contains a gradient vector of the corresponding function. When the Jacobian matrix is determined on the basis of analytical derivatives, the Newton method has a quadratic convergence rate near the solution. Since analytical Jacobian matrix elements are seldom easily available, it is the established practice to approximate the Jacobian matrix numerically, based on finite differences such as forward or central differences.

Since the sparsity pattern of the Jacobian matrix is equal to the occurrence (or incidence) matrix, the location of zero and non-zero elements in the Jacobian matrix may be known in advance. The sparsity pattern of the Jacobian can also be gained from the Jacobian matrix approximation carried out in the first iteration round. Since only non-zero elements need to be stored, the storage demand decreases. In addition, the number of function evaluations in the numerical Jacobian matrix approximation can be reduced and the computation rationalized by taking the Jacobian matrix sparsity pattern into account. The use of sparse matrix techniques is therefore highly recommended.

4.1.3 Quasi-Newton methods

Even though the Newton-Raphson method is well known and has a quadratic convergence property close to the solution, the need for approximating the Jacobian matrix in every iteration round makes the method computationally tedious. This is a fact regardless of whether the Jacobian matrix is determined analytically or numerically.

In order to reduce the work that is spent in Jacobian matrix approximation, the quasi-Newton methods have been developed. In quasi-Newton methods the Jacobian matrix approximation obtained from the previous iteration round is updated based on the function evaluations of the present round. Thus, continual Jacobian matrix evaluation is avoided and, in an ideal case, the Jacobian matrix needs to be determined only in the first iteration round.

Broyden's method is the most popular quasi-Newton method. It is based on the secant updating strategy, where the Jacobian matrix approximation $\mathbf{B}$ is updated based on the following updating formula:

$$\mathbf{B}^{k+1} = \mathbf{B}^{k} + \frac{((\mathbf{f}^{k+1} - \mathbf{f}^{k}) - \mathbf{B}^{k}\mathbf{p}^{k})(\mathbf{p}^{k})^{T}}{(\mathbf{p}^{k})^{T}\mathbf{p}^{k}}. \tag{17}$$

Broyden's method, like secant updating methods in general, has a superlinear convergence rate close to the root.

Broyden's method is not suitable for sparse problems since the sparsity pattern of the Jacobian matrix is not preserved in updating. To preserve the original sparsity, Schubert's method has been developed, where only non-zero Jacobian matrix elements are stored and updated.

Because the quasi-Newton methods only update the Jacobian, they admittedly result in inaccuracy in the solving, thus decreasing the robustness. This weakness can be partly tackled by approximating (initiating) the Jacobian matrix in iteration rounds where a decrease in function values is not achieved. The new approximated Jacobian matrix is utilized in updating from then on.

4.2 Homotopy continuation methods

As recently as in the early 1980s, computer technology was not mature enough for the large-scale utilization of homotopy continuation methods in chemical engineering. However, the fast increase in the availability and power of computers has made homotopy continuation methods usable in chemical engineering.

Homotopy continuation methods are also referred to by other names such as embedding methods, continuation methods and homotopy methods (Allgower & Georg 2003, Wayburn and Seader in Westerberg & Chien 1984). These methods are known as *globally convergent* methods and even methods of 'last resort', which can almost be guaranteed to converge from an arbitrary starting point to a

solution (Wayburn & Seader 1987). Homotopy continuation methods also make it possible to approach several solutions from a single starting point. Therefore, they are usable in multiplicity studies.

Computationally, homotopy methods are substantially heavier than locally convergent solving methods. Therefore, the solving times of homotopy methods are certainly longer. However, the clear benefit of homotopy continuation methods in terms of better robustness becomes advantageous in cases where the other methods fail to obtain a solution.

It is worth noting that both problem-dependent and problem-independent homotopy methods exist. The division is made based on the role of the continuation parameter. In the case of problem-dependent homotopy, the continuation parameter has a physical meaning. In the case of problem-independent homotopy, the continuation parameter is an artificial parameter without physical meaning.

All the non-linear equation set solving methods considered in this thesis are problem-independent. Therefore, only homotopy methods with respect to an arbitrary homotopy parameter are considered herein.

4.2.1 Homotopy methods

A homotopy $\mathbf{h}(\mathbf{x},\theta)$ is a continuous blending of two functions, $\mathbf{f}(\mathbf{x})$ and $\mathbf{g}(\mathbf{x})$, by means of a homotopy parameter θ (Wayburn & Seader 1987). $\mathbf{f}(\mathbf{x})$ describes the equation set that is being solved and $\mathbf{g}(\mathbf{x})$ is an auxiliary set of equations, whose solution is known or can easily be found. Homotopy equations constructed of $\mathbf{f}(\mathbf{x})$ and $\mathbf{g}(\mathbf{x})$ must satisfy the condition (Kovach 1987)

$$\mathbf{h}(\mathbf{x},\theta) = \mathit{func}\left[\mathbf{f}(\mathbf{x}),\mathbf{g}(\mathbf{x}),\theta\right] = \mathbf{0} \tag{18}$$

such that

$$\mathbf{h}(\mathbf{x}^0,0) = \mathbf{g}(\mathbf{x}^0) = \mathbf{0} \text{ (simple equation system)}$$
$$\mathbf{h}(\mathbf{x}^*,1) = \mathbf{f}(\mathbf{x}^*) = \mathbf{0} \text{ (difficult (actual) equation system).}$$

The basic idea of homotopy methods is to provide an implicitly defined curve from a starting point $(\mathbf{x}^0,0)$ to a solution $(\mathbf{x}^*,1)$. If this succeeds, the solution $\mathbf{x}^*$ of the original equation set $\mathbf{f}(\mathbf{x})$ can be obtained. (Allgower & Georg 2003)

Homotopy methods introduce nonlinearities gradually (Seider *et al.* 1991). In fact, *solving a sequence of equations with a diminishing degree of simplifications until finally the actual equation set is solved is called homotopy* (Seydel 1988).

The smooth continua of solutions defined by the homotopy equation are called *branches*, and the diagram formed as a *branching diagram* (Seydel 1988). The basic idea and some general terms appearing in homotopy methods are illustrated in Figure 10.

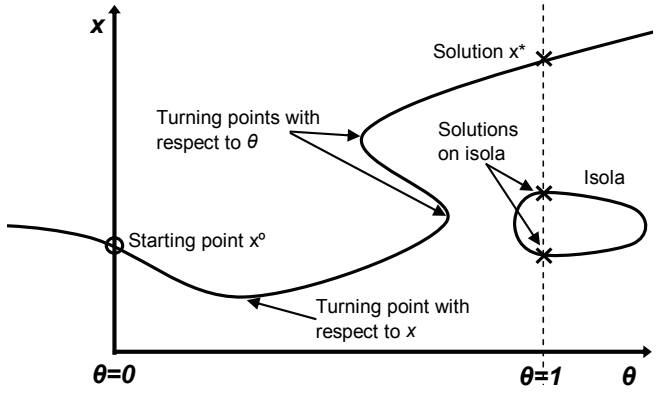

Fig. 10. Two branches and some general terms appearing in homotopy methods.

In general, homotopy methods are based on the homotopy function defined by

$$\mathbf{h}(\mathbf{x},\theta) = \theta\,\mathbf{f}(\mathbf{x}) + (1-\theta)\,\mathbf{g}(\mathbf{x}) = \mathbf{0}\,, \tag{19}$$

where $\mathbf{h}(\mathbf{x},\theta)$ is a convex combination of $\mathbf{f}(\mathbf{x})$ and $\mathbf{g}(\mathbf{x})$ (Choi *et al.* 1996). Depending on the selection of $\mathbf{g}(\mathbf{x})$, different homotopies are obtained. The following three homotopies are presented e.g. in Wayburn & Seader (1987):

$$\text{Newton (or global) homotopy: } \mathbf{g}(\mathbf{x}) = (\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{x}^0)) \tag{20}$$

$$\text{Fixed-point homotopy: } \mathbf{g}(\mathbf{x}) = (\mathbf{x} - \mathbf{x}^0) \tag{21}$$

$$\text{Affine homotopy: } \mathbf{g}(\mathbf{x}) = \mathbf{A}(\mathbf{x} - \mathbf{x}^0). \tag{22}$$

The matrix $\mathbf{A}$ in Eq. (22) denotes a proper weighting matrix that is utilized to scale the elements of a difference vector $(\mathbf{x} - \mathbf{x}^0)$ properly to fit with the equations of the original equation set $\mathbf{f}(\mathbf{x})$. A typical choice is the Jacobian matrix $\mathbf{f}'(\mathbf{x})$ defined at the starting point $\mathbf{x}^0$. This selection guarantees scale invariance with respect to the variables. The Newton homotopy is also scale-invariant.

It is worth noting that different homotopies have different properties. In the Newton homotopy, numerical values of the elements in the vector $\mathbf{f}(\mathbf{x})$ decrease

linearly from their initial values at $\theta = 0$ to zero at $\theta = 1$ (Kovach 1987). In addition, the Newton homotopy has the advantage that closed paths for $\mathbf{x}$ on θ are possible within a finite domain. However, since the auxiliary function $\mathbf{g}(\mathbf{x}) = \mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{x}^0)$ may have several solutions, the Newton homotopy may have multiple roots at $\theta = 0$. This means that more than one branch may exist at $\theta = 0$. (Seader *et al.* 1990)

The starting point multiplicity problem of the Newton homotopy can be avoided by using fixed-point or affine homotopy. This is because in these homotopies the auxiliary function $\mathbf{g}(\mathbf{x})$ is satisfied by only one root at $\theta = 0$.

4.2.2 Continuation principles for homotopy path tracking

Continuation can be considered as a way to track the curve defined by n *scalar equations in the* $(n+1)$*-dimensional* $(\mathbf{x}, \theta)$ *space* (Seydel & Hlavacek 1987). Despite the *branch tracking* (or *path following*) method that is utilized, a chain of solution points is formed that is expected to lead from the starting point $(\mathbf{x}^0, 0)$ to the actual solution $(\mathbf{x}^*, 1)$.

Tracking can be carried out with the *predictor-corrector* strategy illustrated in Figure 11. The next point on the homotopy path is predicted $(\bar{\mathbf{x}}^{(k+1)}, \bar{\theta}^{(k+1)})$ based on the previous point $(\mathbf{x}^{(k)}, \theta^{(k)})$. The predicted point is corrected until a point $(\mathbf{x}^{(k+1)}, \theta^{(k+1)})$ is approached that fulfils the criterion set for error tolerance ε. Corrector iterations can be carried out with a locally convergent solving method. The distance between points $(\mathbf{x}^{(k)}, \theta^{(k)})$ and $(\mathbf{x}^{(k+1)}, \theta^{(k+1)})$ is called *step size* or *step length*. (Seydel 1988)

A vast number of continuation methods can be constructed by combining the four basic elements of continuation methods differently (Seydel 1988):

a) predictor
b) parameterization strategy
c) corrector
d) step length control.

The first three of these can be chosen independently of each other. However, the step length control must correspond to the predictor, the underlying parameterization and the corrector. Different continuation methods have their merits and relatively simple principles work satisfactorily even for complicated problems. Therefore, no particular continuation method can be recommended exclusively. (Seydel 1988)

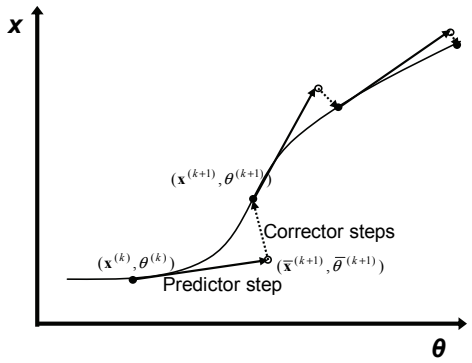

Fig. 11. The basic idea of predictor-corrector tracking strategy. Predicted points (○) and corrected points (●).

According to Richter (1983), there are two main methods of following a homotopy curve. These are the *discrete* and *continuous* path following methods. In the representation below, the discrete method is briefly reviewed first. Then, the numerical principles applied in continuous methods are considered more widely. Path tracking based on arc length parameterization is given particular consideration.

Discrete homotopy path tracking

In discrete homotopy path tracking, which is also referred to as the classical embedding method, the homotopy parameter interval [0,1] is divided to obtain a finite chain of problems:

$$\mathbf{h}(\mathbf{x}^{(k)}, \theta^{(k)}) = \mathbf{0}, \qquad 0 = \theta^{(0)} < \theta^{(1)} < \ldots < \theta^{(p)} = 1. \tag{23}$$

As illustrated in Figure 12, the next solution point $\mathbf{x}^{(k+1)}$ at $\theta^{(k+1)}$ is computed using the previous solution point $\mathbf{x}^{(k)}$ at $\theta^{(k)}$ as an initial guess for the corrective iterations. The procedure is continued until the actual solution at $\theta = 1$ is achieved.

To succeed in tracking, a sufficiently small increment $\Delta\theta$ must be chosen. Thus, a starting point $\mathbf{x}^{(k)}$ close enough to the next solution point $\mathbf{x}^{(k+1)}$ can be offered. To enable accurate tracking, the number of necessary steps with respect to θ may become high. (Richter 1983)

A clear drawback of the discrete path tracking method is that it fails when turning points are encountered with respect to the homotopy parameter θ. To

increase the robustness in path tracking, it is necessary to track the path closely rather than only use increments with respect to the homotopy parameter. Therefore, path-tracking methods based on arc length are preferred to the discrete method. (Allgower & Georg 2003)

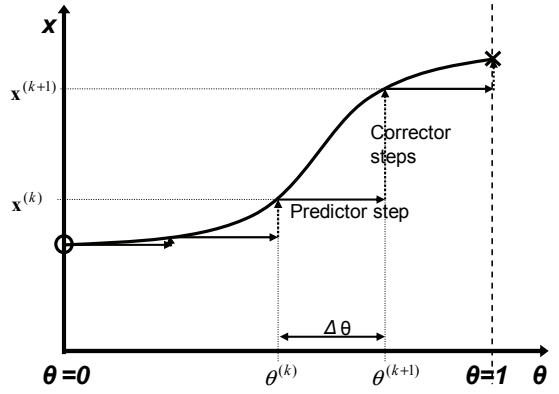

Fig. 12. The basic idea of discrete homotopy path tracking.

Parameterization in continuous homotopy path tracking

Parameterization can be understood as a measure along the homotopy path – *a mathematical way of identifying each solution on the branch* (Seydel 1988). Parameterization plays an important role in continuous path tracking. Various parameterization strategies are presented in Figure 13.

When parameterization is carried out with respect to the homotopy parameter θ, the predictor step length is determined based on the progress in relation to the homotopy parameter. In the corrector phase, the predicted step is corrected orthogonally to the homotopy parameter. This parameterization strategy does not work at turning points, where the path turns with respect to the homotopy parameter. To tackle this, a local parameterization strategy can be applied.

In the local parameterization strategy, the variable utilized in parameterization is altered during path tracking. Parameter switching on a local basis is referred to as reparameterization (Wayburn & Seader 1987). At a particular point on the homotopy path, the selection may be made by picking a variable, which has maximal change perpendicular to the homotopy parameter. The predictor step length is determined based on the increment in the parameterization variable Δx_i, and the correction is made orthogonally to the parameterization variable.

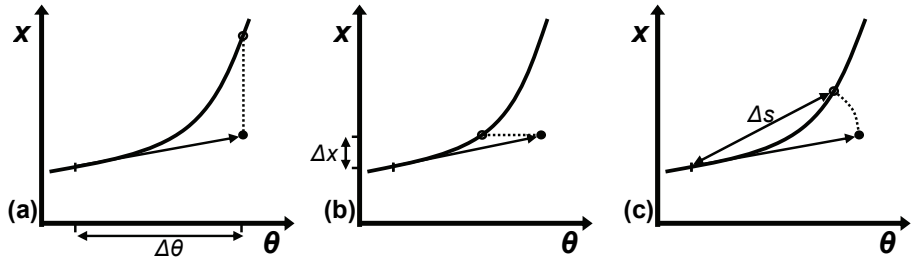

Fig. 13. The three parameterization strategies utilized in continuous path tracking. (a) Parameterization with respect to the homotopy parameter, (b) local parameterization, and (c) arc length parameterization.

Of the three parameterization strategies illustrated in Figure 13, arc length parameterization is the most sophisticated. In arc length parameterization, the predicted point $(\bar{\mathbf{x}}^{(k+1)}, \bar{\theta}^{(k+1)})$ and the corrected point $(\mathbf{x}^{(k+1)}, \theta^{(k+1)})$ lie equidistant from the previous point $(\mathbf{x}^{(k)}, \theta^{(k)})$.

Prediction of direction in continuous homotopy path tracking

In order to deal with turning points, $\mathbf{x}$ and θ can be considered as functions of arc length s. Differentiating the homotopy function $\mathbf{h}(\mathbf{x}, \theta)$ with respect to the arc length, results in (Wayburn & Seader 1987):

$$\frac{d\mathbf{h}}{ds} = \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \frac{d\mathbf{x}}{ds} + \frac{\partial \mathbf{h}}{\partial \theta} \frac{d\theta}{ds} = \mathbf{0} . \tag{24}$$

By taking into account the arc length relation

$$(dx_1/ds)^2 + \cdots + (dx_n/ds)^2 + (d\theta/ds)^2 = 1 \tag{25}$$

and the initial condition $\mathbf{h}(\mathbf{x}^0, 0) = \mathbf{0}$, an initial value problem (IVP) is obtained. It is worth noting that the system presented in Eq. (24) is linear and the arc length relation Eq. (25) non-linear (Seydel 1988). Therefore, the matrix formulation resulted by combining Eqs. (24) and (25)

$$\begin{bmatrix} \left(\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right)^{(k)} & \left(\frac{\partial \mathbf{h}}{\partial \theta}\right)^{(k)} \\ \left(\frac{d\mathbf{x}}{ds}\right)^{(k)} & \left(\frac{d\theta}{ds}\right)^{(k)} \end{bmatrix} \begin{bmatrix} \frac{d\mathbf{x}}{ds} \\ \frac{d\theta}{ds} \end{bmatrix}^{(k)} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} \tag{26}$$

is non-linear.

In order to determine the tangent vector

$$\left[(dx_1/ds)^{(k)},(dx_2/ds)^{(k)},\ldots,(dx_n/ds)^{(k)},(d\theta/ds)^{(k)}\right]^T$$

at the point $(\mathbf{x}^{(k)},\theta^{(k)})$ without tedious non-linear equation set solving, the normalization

$$\mathbf{e}_t^T\left[(dx_1/ds)^{(k)},(dx_2/ds)^{(k)},\ldots,(dx_n/ds)^{(k)},(d\theta/ds)^{(k)}\right]^T=1 \qquad (27)$$

can be utilized. All the elements in $(n+1)$-dimensional column vector $\mathbf{e}_t$ equal zero except the t th element, which equals unity. As a result, the tangent vector $\mathbf{z}^{(k)}$ can be determined based on the linear equation system

$$\begin{bmatrix}\left(\dfrac{\partial\mathbf{h}}{\partial\mathbf{x}}\right)^{(k)} & \left(\dfrac{\partial\mathbf{h}}{\partial\theta}\right)^{(k)}\\ \mathbf{e}_t^T & \end{bmatrix}\mathbf{z}^{(k)}=\begin{bmatrix}\mathbf{0}\\ 1\end{bmatrix}. \qquad (28)$$

Eq. (28) is also feasible for the determination of a tangent vector $\mathbf{z}^{(k)}$ at the turning points. However, t in Eq. (27) must be selected so that it does not correspond to the variable for which the turning point is encountered. (Seydel 1988)

Since the tangent vector $\mathbf{z}^{(k)}$ is normalized with respect to one parameter, it is basically not a unit tangent vector having the length equal to one. In order to obtain the *unit tangent vector* $\mathbf{u}^{(k)}$, the elements of $\mathbf{z}^{(k)}$ are divided by the Euclidian norm of $\mathbf{z}^{(k)}$

$$\mathbf{u}^{(k)}=\frac{1}{\left\|\mathbf{z}^{(k)}\right\|_2}\mathbf{z}^{(k)}. \qquad (29)$$

In principle, there are two directions for the unit tangent vector $\mathbf{u}^{(k)}$ that fulfil the relation presented in Eq. (26). These directions are the reverse for each other, that is $\mathbf{u}^{(k)}$ and $-\mathbf{u}^{(k)}$. The two directions are often called ‘positive’ and ‘negative’ directions with respect to the homotopy parameter.

Arc length based predictor-corrector strategy in continuous homotopy path tracking

After determining the tangent at a point $(\mathbf{x}^{(k)}, \theta^{(k)})$, the next point can be predicted based on the first-order Euler method according to

$$\begin{bmatrix} \bar{\mathbf{x}} \\ \bar{\theta} \end{bmatrix}^{(k+1)} = \begin{bmatrix} \mathbf{x} \\ \theta \end{bmatrix}^{(k)} + \lambda^{(k)} \mathbf{u}^{(k)} . \tag{30}$$

The distance between the predicted point $(\bar{\mathbf{x}}^{(k+1)}, \bar{\theta}^{(k+1)})$ and the previous point $(\mathbf{x}^{(k)}, \theta^{(k)})$ is λ^k, which is the step length. Instead of the ODE predictor described by Eq. (30), the predictor step may be based on some polynomial extrapolation method. In the first order secant method, the next point is predicted based on two previous corrected points. (Seydel 1988)

Since all prediction methods simply approximate the course of the path, a corrector phase must follow the predictor phase in order to return the predicted point more closely to the actual homotopy path. Corrective iterations can be carried out with some local solving method, the Newton-Raphson method for instance.

One option is to select the direction for the correction steps from the hyperplane that is orthogonal to the unit tangent vector $\mathbf{u}^{(k)}$. By iteratively correcting the predicted point, the arc length parameterization strategy illustrated in Figure 13c is imitated. In this case the direction for correction iterations is defined according to

$$\begin{bmatrix} \left(\frac{\partial \mathbf{h}}{\partial \bar{\mathbf{x}}}\right)^{(i)} & \left(\frac{\partial \mathbf{h}}{\partial \bar{\theta}}\right)^{(i)} \\ \mathbf{u}^{(k)} & \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ \Delta \theta \end{bmatrix}^{(i)} = \begin{bmatrix} -\mathbf{h}(\bar{\mathbf{x}}^{(i)}, \bar{\theta}^{(i)}) \\ 0 \end{bmatrix}, \tag{31}$$

where $(\Delta \mathbf{x}^{(i)}, \Delta \theta^{(i)})^T$ is the i th correction step. The sequence of correction iterations based on Eq. (31) and

$$\begin{bmatrix} \bar{\mathbf{x}} \\ \bar{\theta} \end{bmatrix}^{(i+1)} = \begin{bmatrix} \bar{\mathbf{x}} \\ \bar{\theta} \end{bmatrix}^{(i)} + \begin{bmatrix} \Delta \mathbf{x} \\ \Delta \theta \end{bmatrix}^{(i)} \tag{32}$$

is continued until a point $(\mathbf{x}^{(k+1)}, \theta^{(k+1)})$ close enough to the actual homotopy path is approached. (Seydel 1988, Wayburn & Seader 1987)

Step length control in continuous homotopy path tracking

A uniformly constant step length is not an efficient strategy to track a homotopy path. To increase efficiency, long steps are preferred. However, lengthening the predictor step generally increases the distance between the predicted point and the actual homotopy path. Thus, the number of corrective steps that are required in the corrector phase may also increase. This increases total computation time and decreases efficiency. Overlong steps also weaken robustness, for example by increasing the risk for segment jumping. In the worst case, the path tracking may even fail totally. (Allgower & Georg 2003, Choi *et al.* 1996)

Even though the efficiency and robustness of path tracking are to some extent contradictory to each other, by properly tuning the path tracking algorithm it is possible to attain a trade-off that offers good robustness and satisfactory efficiency. A step length control strategy is necessary in this.

Perhaps the simplest strategy to control step length adaptively has been presented in Seydel (1988) and Seydel & Hlavacek (1987). Their strategy is based on the fact that continuation is expensive for both short steps (many predictor steps) and long steps (slow or no convergence of correctors). Thus costs can be seen to be moderate with a certain 'medium' step length, which is related to some optimal number of corrector iterations. When the number of corrector iterations N is higher than the optimal number of corrector iterations N_{opt}, the step length is reduced. Likewise, the step length is increased in the case of $N < N_{opt}$. One option for updating the step length is to multiply it by the factor N_{opt} / N.

4.2.3 Fundamental causes of failure

There are several reasons why problem solving based on homotopy continuation methods may fail. Problems may appear because of

- shortcomings in path tracking, or
- failures caused by the course of the path of the homotopy method.

Loose path tracking easily fails. Failure occurs for example when a path makes sharp curves. In addition, loose path tracking may lead to segment jumping, where the path tracking routine jumps from one segment of the homotopy path to another. In this case the complete path may not be tracked and roots or other points of interest on the untracked segment are missed (Choi *et al.* 1996).

Even though many challenges in path tracking can be avoided simply by tracking the homotopy path more accurately, there are several fundamental causes of failure, which cannot be tackled by improving the performance of the path tracking algorithm alone. These failures are caused by the combined effect of the considered problem $\mathbf{f}(\mathbf{x})$ with its restrictions, and the homotopy equation $\mathbf{h}(\mathbf{x},\theta)$.

The following six causes of failure illustrated in Figure 14 have been compiled from Wayburn & Seader (1987) and Christiansen *et al.* (1996):

1. The Jacobian of $\mathbf{h}(\mathbf{x},\theta)$ becomes singular at turning points.
2. The homotopy path strikes an interior boundary of the domain of definition, i.e. boundary striking problem.
3. The homotopy path may exceed the domain on which the variables are defined.
4. The homotopy path becomes totally unbounded, i.e. runs towards infinity.
5. Multiple solutions exist for $\mathbf{g}(\mathbf{x})$, i.e. starting point multiplicity problem.
6. Occurrence of isolated solutions.

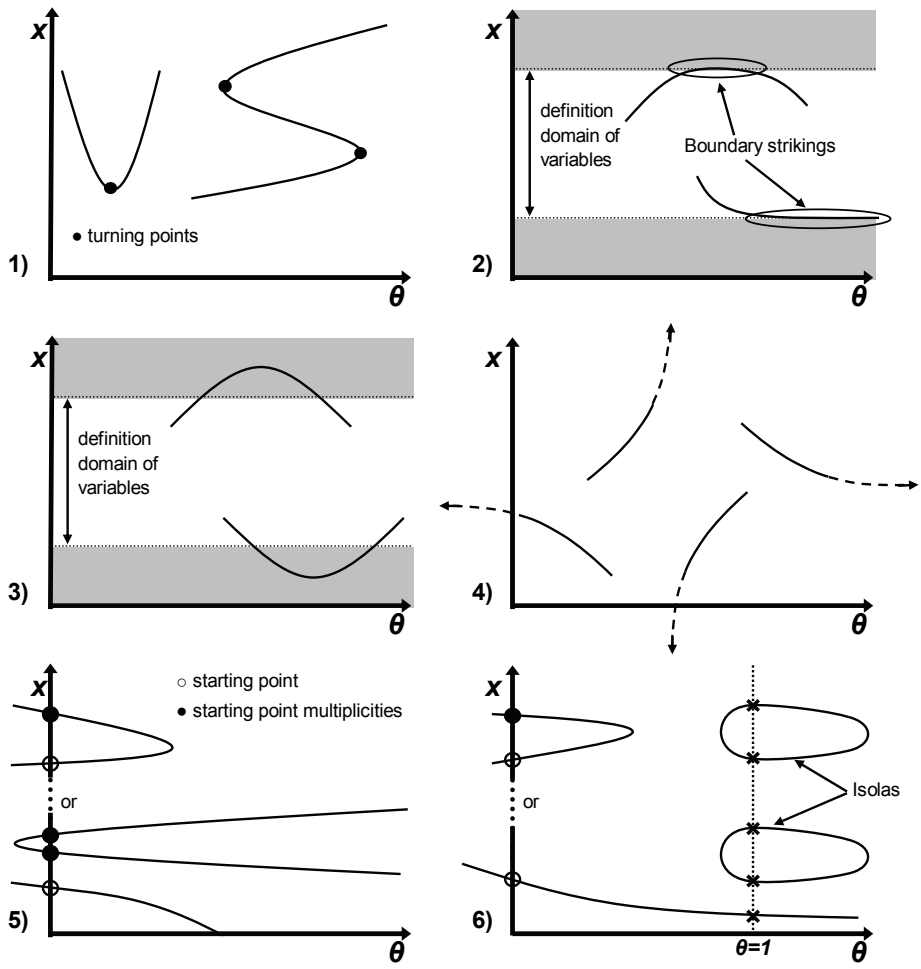

Fig. 14. Some fundamental causes of failure in homotopy continuation based problem solving.

Turning point singularity

The problem of turning point singularity is closely dependent on what kinds of principles the solving algorithm exploits in path tracking. Utilizing an *arc length* path tracking strategy means that the challenges encountered at turning points can be tackled.

Boundary striking

According to Wayburn & Seader (1987), one requirement for the homotopy continuation method to be able to approach the solution successfully from an arbitrary starting point is that the homotopy path does not strike the definition domain boundary of $\mathbf{f}(\mathbf{x})$, that is the homotopy $\mathbf{h}(\mathbf{x},\theta)$ has no solutions on the domain boundary. Since variables in chemical engineering, molar flows and mole fractions for instance, may have values very close to the problem domain boundary, it is not surprising that the homotopy path may strike the boundary. Wayburn & Seader (1987) calls this *interior boundary striking*.

Even if the actual homotopy path does not cross the problem domain boundary, boundary striking certainly constitutes a challenge in path tracking. For instance, a predictor step in predictor-corrector path tracking may cross the problem domain. Crossing may also occur in corrector steps. In addition, when a Jacobian matrix is approximated numerically close to the problem domain boundary with some finite difference method, it may happen that an unfeasible variable value is substituted into a thermodynamic calculation routine, in which case a fatal error may result.

Lin *et al.* (1987) have presented means to revise negative variable values into positive. One of these is based on variables mapping, where all steps to the negative variable space are mapped back to the positive space according to

$$\hat{x}_i = x_i e^{\left(\frac{\omega \Delta x_i}{x_i}\right)}, \tag{33}$$

where $\hat{x}_i$ is the mapped variable of x_i, ω is a non-negative scalar step factor and Δx_i is the step of variable x_i. Another strategy is to convert negative values to an arbitrary small positive value, e.g. 10^{-8}.

Variables exceeding the definition domain

In some cases the homotopy path runs partly or entirely outside the problem domain. Wayburn & Seader (1987) calls this *exterior boundary striking*. In these cases thermodynamic subroutines may generate a fatal error.

Wayburn and Seader (1987) suggested an absolute-value function $\mathbf{f}(|\mathbf{x}|)$ instead of function $\mathbf{f}(\mathbf{x})$. In this case the absolute value is regarded as part of the function and not part of $\mathbf{x}$. This is obtained by defining:

$$\hat{x}_i = |x_i| . \tag{34}$$

Since

$$\frac{d|x_i|}{dx_i} = \begin{cases} +1, & x_i \to 0^+ \\ -1, & x_i \to 0^- \end{cases}$$

Lin *et al.* (1987) suggested utilizing mapping

$$\hat{x}_i^2 = x_i \tag{35}$$

instead. This mapping does not set any constraints. The path is tracked in the $\hat{\mathbf{x}}$ domain instead of in the $\mathbf{x}$ domain. The functions are differentiable anywhere with respect to $\hat{\mathbf{x}}$.

It is clear that the mappings presented above do not prevent variables from having negative values, but only aim to tolerate negative variable values. These mappings are also useless in cases where variables encounter positive values outside their feasible definition domain. For example, mole fraction values above one are absurd and should not be accepted.

The unboundedness of the homotopy path

Occasionally, the homotopy path may run to infinity. This may happen both with respect to the homotopy parameter and problem variables. In these cases the solution may not be achieved at all or some of the multiple solutions may be missed. In addition, when the homotopy path runs to infinity with respect to problem variables $\mathbf{x}$, it inevitably runs outside the feasible problem domain.

Seader *et al.* (1990) suggested mapped continuation methods, called *toroidal* and *boomerang* mappings. In these methods, $\mathbf{x}$ and/or θ that extend to $\pm\infty$ are kept limited through a suitable transformation, where an infinite path is transformed into a finite domain. According to Seider *et al.* (1991), when fixed-point homotopy is utilized together with mapping functions, all solutions can usually be found by tracking a single path of finite length.

To prevent unboundedness in homotopy paths, Christiansen *et al.* (1996) suggested *branch-jumping* techniques. They used a simple *inverse mapping* function and showed how symmetry may be utilized to predict where solution branches connect across asymptotes. In the cases of single and linear asymptotes inverse mapping worked satisfactorily, but in situations where several asymptotes lie close in the variable space or in the cases of non-linear asymptotes the

situation is not that simple and the algorithm displayed poor convergence characteristics.

Starting point multiplicities

Starting point multiplicities are occasionally encountered when the Newton homotopy is utilized. This is because the auxiliary function $\mathbf{g}(\mathbf{x}) = \mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{x}^0)$ may have several roots.

One evident consequence is that the Newton homotopy path may double back to a second root at $\theta = 0$ without passing through the desired root at $\theta = 1$. Hence, instead of approaching the solution for the original problem $\mathbf{f}(\mathbf{x}) = \mathbf{0}$, another root for $\mathbf{g}(\mathbf{x}) = \mathbf{0}$ is approached (Christiansen *et al.* 1996, Wayburn & Seader 1987).

As illustrated in Figure 14, starting point multiplicities also provide the opportunity for separate homotopy path branches. In this case, a trackable real space path from the selected starting point at $\theta = 0$ to the solution at $\theta = 1$ may not exist.

Occurrence of isolated solutions

Even though homotopy methods have more global convergence compared to locally convergent solving methods, it may not always be possible to approach all solutions from a single starting point by tracking the homotopy path based on real space arithmetic alone. This happens when the roots of the equation set $\mathbf{f}(\mathbf{x}) = \mathbf{0}$ lie on a separate homotopy path branch than the starting point.

As illustrated in Figure 14, in some cases some of the roots may lie on an isola. An isola is a closed homotopy branch that is unreachable from the starting point based on real space arithmetic alone, but can be reached by computation in a complex domain. Once the first root on the isola is reached, the remaining isola roots can be obtained by calculations in the real space domain (Kuno & Seader 1988).

Basically, all variables in $\mathbf{x}$ and the homotopy parameter θ may be considered as complex numbers. Even though separate branches have trackable connections in complex space, the utilization of complex space arithmetic would make path tracking a challenging task. One challenge is how to determine the bifurcation points, where two or more branches intersect. The determination of intersection points is important because they offer a connection between branches.

5 Bounded homotopies

> *"One of the difficulties in using the available homotopies to solve sets of algebraic non-linear equations is the unboundedness of the continuation path. This is particularly important when physical properties must be calculated within some prescribed domain."* (Paloschi 1995)

Paloschi (1995, 1996, 1998) presented bounded homotopies to prevent the homotopy path from exceeding the boundaries of the problem domain. Bounded homotopies restrict the course of the homotopy path and thus the variables can be kept inside the feasible problem domain. This avoids the problem of fatal error generated by unfeasible variable values in thermodynamic subroutines. In addition, variable values outside the validity range of thermodynamic correlations can be avoided.

Despite the sophisticated basic principle, the bounded homotopies proposed by Paloschi do not work satisfactorily when simulating chemical engineering systems, such as distillation systems. To tackle these shortcomings, Papers I, II and IV introduce modifications, which aim to improve the usability of bounded homotopies in chemical engineering.

5.1 Bounded homotopies proposed by Paloschi

Bounded homotopies can be expressed with the general equation:

$$\mathbf{h}_b(\mathbf{x},\theta) = \pi(\mathbf{x})\mathbf{h}(\mathbf{x},\theta) + \upsilon(\mathbf{x}) - \upsilon(\mathbf{x}^b) = \mathbf{0}\,. \tag{36}$$

The bounded homotopy path defined by $\mathbf{h}_b(\mathbf{x},\theta)$ coincides with the path formed by the unbounded homotopy $\mathbf{h}(\mathbf{x},\theta)$ inside the subset Ω restricted by the box constraints

$$\Omega = \left\{x \in \Re^n : l_i \le x_i \le u_i,\ i = 1, \ldots, n\right\}, \tag{37}$$

and departs from it outside. As Figure 15 illustrates, the bounding zone is formed outside the subset boundary $\delta\Omega$. The width of the bounding zone is defined based on the selected positive constant δ that is a measure for the bounding zone width relative to the width of subset Ω. In fact, the basic idea of bounded homotopies is to force the homotopy path always to run inside subset Ω' composed by subset Ω and the bounding zone.

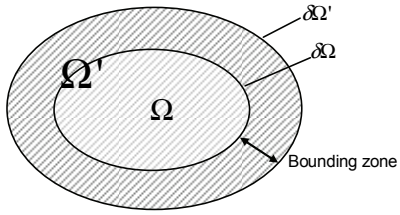

Fig. 15. A schematic representation of the formation of the bounding zone. Ω' describes an open and bounded subset that is composed of subset Ω and the bounding zone. The bounding zone is restricted by boundaries $\partial\Omega$ and $\partial\Omega'$.

One option for defining the auxiliary function $\mathbf{v}(\mathbf{x})$ in Eq. (36) is to specify it as an affine function:

$$\mathbf{v}(\mathbf{x}) = \mathbf{f}'(\mathbf{x}^0)(\mathbf{x} - \mathbf{x}^0) . \tag{38}$$

In this case the general Eq. (36) gets the form:

$$\mathbf{h}_b(\mathbf{x}, \theta) = \pi(\mathbf{x})\mathbf{h}(\mathbf{x}, \theta) + \mathbf{f}'(\mathbf{x}^0)(\mathbf{x} - \mathbf{x}^b) . \tag{39}$$

$\pi(\mathbf{x})$ is a penalty function that annihilates the magnitude of the unbounded homotopy $\mathbf{h}(\mathbf{x}, \theta)$ whenever the homotopy path runs outside Ω . The numerical value of the penalty function $\pi(\mathbf{x}) \in [0\ 1]$ depends on the course of the path inside the subset Ω'. Inside subset Ω the value is one and changes nonlinearly from one to zero when running from $\partial\Omega$ to $\partial\Omega'$. Correspondingly, the auxiliary function $\mathbf{f}'(\mathbf{x}^0)(\mathbf{x} - \mathbf{x}^b)$ compensates the annihilation. The definitions for penalty function $\pi(\mathbf{x})$ and vector $\mathbf{x}^b$ can be found in the original papers by Paloschi (1995, 1996, 1998).

In general, bounded homotopies have the following benefits:

- The homotopy path and thereby the problem variables can be kept inside a feasible domain. Thus, fatal errors generated by unfeasible variable values in thermodynamic subroutines are avoided. In addition, a homotopy path can be forced to run inside the domain where the thermodynamic correlations are valid.
- The bounded homotopy path is shorter than the unbounded one. The shorter path may reduce the number of steps required in path tracking.

Despite the benefits, there are also some shortcomings in bounded homotopies when applied as presented by Paloschi (1995, 1996, 1998). These shortcomings

have decreased the usability of bounded homotopies in chemical engineering. The following shortcomings have been compiled from Paper I:

- The way the bounding zone is defined is not beneficial. In the papers by Paloschi (1995, 1996, 1998), the strategy is utilized where a bounding zone is formed outside the predetermined constraints $\mathbf{l}$ and $\mathbf{u}$, based on the magnitude of the positive constant δ. In this case, it may be difficult to match the outermost values of the bounding zone with the real physical constraints.
- In the case of a narrow bounding zone, the numerical challenges in path tracking are considerable. This is illustrated in Figure 16. When the path is close to the narrow bounding zone, the predictor-corrector path tracking method easily predicts points outside the feasible problem domain. In this case, the predicted step must be shortened to adjust the predicted point inside the problem domain. However, because the narrow bounding zone forces the bounded path to bend abruptly after running inside the bounding zone, very short predictor steps need to be taken in order to enable accurate path tracking. In addition, corrector steps may easily fall outside the problem domain. The increasing number of predictor steps and continuous need to return the steps that have fallen outside the problem domain back into the feasible domain certainly increase the solving time.
- In the case of a narrow bounding zone, the numerical values of the elements in the auxiliary function vector $\mathbf{f}'(\mathbf{x}^0)(\mathbf{x}-\mathbf{x}^b)$ of Eq. (39) are particularly small. Thus the magnitude of compensation given by the auxiliary function against the annihilation effect of penalty function $\pi(\mathbf{x})$ is almost negligible. As a result, homotopy path tracking may fail and an erroneous solution inside the bounding zone might be attained rather than the correct one. Thus, the general equation of bounded homotopies (Eq. (39)) has a poor bounding effect and even a questionable capability of keeping the path strictly bounded.
- The penalty function $\pi(\mathbf{x})$ in Eq. (39) also focuses the annihilation effect on equations where bounded variables do not exist. In addition, in the case of a large and sparse equation set, the element of the diagonal penalty matrix $\mathbf{\Pi}(\mathbf{x})$ focuses the annihilation effect only on one equation in $\mathbf{h}(\mathbf{x},\theta)$. In this case, the annihilation effect is not focused on all the equations where bounded variables exist. This weakens the accuracy.
- The way to preserve sparseness in bounded homotopies applied to solving sparse problems is largely based on the utilization of zero-free diagonal

matrices (Paloschi 1998). Considering only the diagonal terms admittedly accelerates solving and gives savings in the required storage capacity. However, omitting the non-diagonal terms may result in inaccuracies in path tracking.

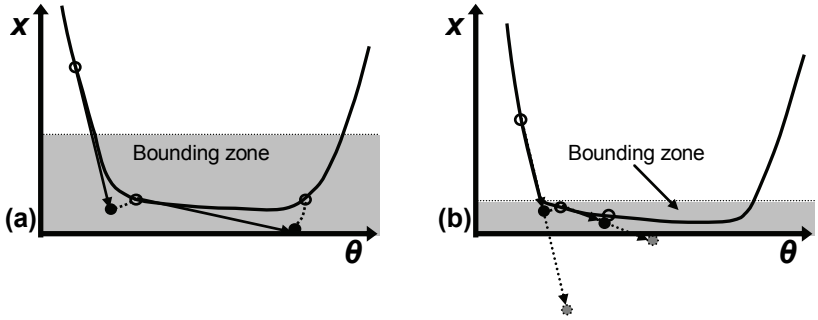

Fig. 16. The effect of bounding zone width on homotopy path tracking. Path tracking is (a) easy with a wide bounding zone, and (b) challenging with a narrow bounding zone.

5.2 Modified bounded homotopies

It is known that some variables, such as mass and mole fractions in distillation calculation, may often have reasonable values very close to the problem domain boundaries. Therefore, in order to avoid a situation where one or more of the actual solutions fall inside the bounding zone, the bounding zone must be kept sufficiently narrow.

To force the homotopy path to be tightly bounded and to enable flexible and accurate path tracking inside a narrow bounding zone, not only must the general equation of the bounded homotopies be modified, but also a suitable variables mapping strategy needs to be applied. When implementing these, it is important to preserve the problem sparsity and thus make it possible to keep the number of function evaluations reasonable in the numerical Jacobian matrix approximation.

With respect to the shortcomings and requirements presented above, modified bounded homotopies have been proposed in Papers I and IV. Modified bounded homotopies include formulations for both small- and large-scale problems. In addition, the concept of homotopy parameter bounding has been proposed in Paper II. The concept aims to establish a path in real space so that multiple

solutions, which would be unattainable with unbounded homotopy method, could be approached.

5.2.1 Variables mapping

In order to facilitate accurate and flexible path tracking close to the problem domain boundaries, variables mapping has been proposed in Papers I and IV. The proposed mapping scales a finite variable space into an infinite space, so that problem variables become more sensitive to changes close to the problem domain boundaries. Sensitivity is a useful property especially when utilizing a narrow bounding zone. In addition, the proposed mapping tackles the boundary striking problem that may appear both in problem-dependent and problem-independent homotopies.

The basic idea of variables mapping is to offer a way to track the homotopy path in an infinite space $]-\infty\ \infty[$ rather than a finite space $[\mathbf{b}^{\min}\ \mathbf{b}^{\max}]$. On the basis of maximum ($b_i^{\max}$) and minimum ($b_i^{\min}$) values that are set for every variable x_i, mapping from a finite space into an infinite space is carried out as

$$x_i^{\inf} = \log_{10}\left(\frac{2(x_i - b_i^{\min})}{b_i^{\max} - b_i^{\min}}\right) \quad \text{when } x_i < \frac{1}{2}(b_i^{\max} + b_i^{\min}), \tag{40}$$

$$x_i^{\inf} = \log_{10}\left(\frac{0.5(b_i^{\max} - b_i^{\min})}{b_i^{\max} - x_i}\right) \quad \text{when } x_i \geq \frac{1}{2}(b_i^{\max} + b_i^{\min}). \tag{41}$$

The maximum ($b_i^{\max}$) and minimum ($b_i^{\min}$) values can be realized as domain boundary values, either real physical restrictions or as artificial values. The mole fraction, for instance, has physical restrictions 0 and 1, which can be utilized as domain boundary values. Because molar flows and reflux and reboil ratios have a natural minimum value of 0 but not a self-evident maximum, an artificial value of 1000 can be selected. In addition, temperatures (in Kelvins) can be bounded by [250 450]. All the problem variables have been mapped as described above in the distillation cases examined in Papers I–VI.

Although the homotopy path is tracked in an infinite variable space, the variables must be mapped into a finite space before substituting them into the equations of the original equation set $\mathbf{f}(\mathbf{x})$. The mapping is carried out based on the following equations:

$$x_i = b_i^{\min} + \frac{1}{2}(b_i^{\max} - b_i^{\min})10^{x_i^{\inf}} \qquad \text{when } x_i^{\inf} < 0\,, \tag{42}$$

$$x_i = b_i^{\max} - \frac{1}{2}\frac{(b_i^{\max} - b_i^{\min})}{10^{x_i^{\inf}}} \qquad \text{when } x_i^{\inf} \geq 0\,. \tag{43}$$

Even though variables mapping has primarily been proposed to facilitate homotopy path tracking close to problem domain boundaries, mapping is also beneficial when utilizing locally convergent solving methods. By utilizing the proposed mapping strategy, the problem of an iteration step falling outside the feasible problem domain can be avoided. It must be noticed, however, that mapping as such is not sufficient to eliminate the problem encountered with traditional problem-independent unbounded homotopies in cases where the path exceeds the domain on which the variables are defined. Nor does variables mapping make the locally convergent solving methods robust.

5.2.2 Modified bounded homotopies for small-scale problems

The general equation of modified bounded homotopies for small-scale problems equipped with an affine-based auxiliary function can be expressed as

$$\mathbf{h}_b^{\text{mod}}(\mathbf{x}^{\inf}, \theta) = \pi(\mathbf{x}^{\inf})\mathbf{h}(\mathbf{x}^{\inf}, \theta) + \mathbf{f}'(\mathbf{x}^{0,\inf})(\mathbf{x}^{\inf} - \mathbf{x}^{b,\inf})\,. \tag{44}$$

Note that in contrast to Eq. (39), the mapped variables $\mathbf{x}^{\inf}$ are utilized herein instead of unmapped $\mathbf{x}$. In the case of a narrow bounding zone, the numerical values of auxiliary function elements become large, thus offering a strictly bounded homotopy path. In consequence, the path can be tracked without the need for shortening the step length or tightening the error tolerance. As a whole, long steps can be taken without the risk of crossing the problem domain or encountering erroneous solutions inside the narrow bounding zone.

The penalty function

$$\pi(\mathbf{x}^{\inf}) = 1 - \rho_1\left(\left\|\mathbf{W}(\mathbf{x}^{\inf} - \mathbf{x}^{b,\inf})\right\|_\infty, \delta\right) \tag{45}$$

that is utilized in the modified bounded homotopies includes the infinity norm, not the quadratic Euclidean norm($\| \ \|_2^2$) that is utilized in the bounded homotopies proposed by Paloschi (1995, 1996, 1998).

The elements in vector $\mathbf{x}^{b,\inf}$ are obtained from

$$x_i^{b,\inf} = x_i^{\inf} - \rho_2\left(W_{ii}(x_i^{\inf} - x_i^{b',\inf}), \delta\right)\left(x_i^{\inf} - x_i^{b',\inf}\right), \tag{46}$$

and the elements in diagonal weighting matrix $\mathbf{W}$ as

$$W_{ii} = \frac{1}{u_i^{\inf} - l_i^{\inf}}, \tag{47}$$

where the upper and lower constraints, i.e. $u_i^{\inf}$ and $l_i^{\inf}$, form the inner boundaries for the bounding zone in the mapped variable space. In practice it is a good idea to specify them as having the same absolute value with an opposite sign, that is $u_i^{\inf}$ positive and $l_i^{\inf}$ negative. The higher their absolute value, the narrower the bounding zone in the finite variable space.

Based on the chosen $u_i^{\inf}$ and $l_i^{\inf}$, $x_i^{b',\inf}$ is defined as

$$x_i^{b',\inf} = \begin{cases} l_i^{\inf}, & x_i^{\inf} < l_i^{\inf}, \\ x_i^{\inf}, & l_i^{\inf} \le x_i^{\inf} \le u_i^{\inf}, \\ u_i^{\inf}, & x_i^{\inf} > u_i^{\inf}. \end{cases} \tag{48}$$

The functions ρ_1 and ρ_2 existing in Eqs. (45) and (46) are defined as

$$\rho_1(\xi, \delta) = \begin{cases} a_5\xi^5 + a_4\xi^4 + a_3\xi^3, & 0 \le \xi \le \delta \\ 1, & \xi > \delta \end{cases} \tag{49}$$

and

$$\rho_2(\xi, \delta) = \rho_1(|\xi|, \delta). \tag{50}$$

The parameters a_5, a_4 and a_3 are specified as $6/\delta^5$, $-15/\delta^4$ and $10/\delta^3$. By defining the parameter values this way the functions ρ_1 and ρ_2 become twice as continuously differentiable (Paloschi 1996). A graph for ρ_1 as a function of ξ/δ is presented in Figure 17.

Herein, the positive constant δ is not a relative measure for the bounding zone width in the finite variable space. Instead, it is a measure of how close to the actual domain boundary the bounded homotopy path may run in the finite variable space. The higher the value of δ, the closer to the domain boundary the path is allowed to run. In Papers I and III–VI, the value $\delta = 0.5$ has been utilized.

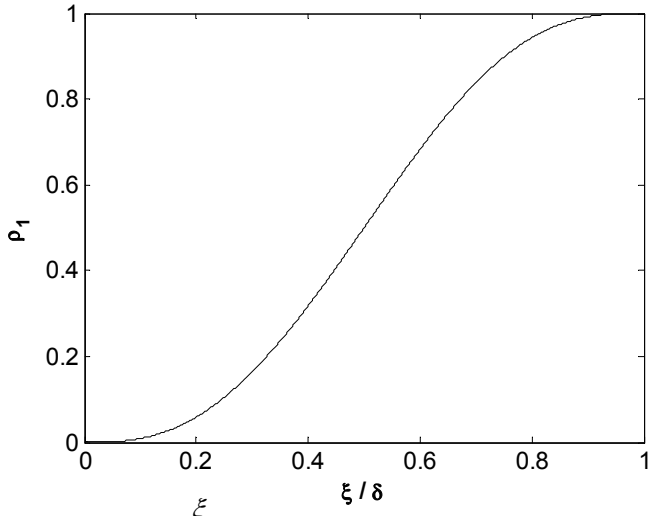

Fig. 17. ρ_1 **as a function of** $\frac{\xi}{\delta}$.

5.2.3 Modified bounded homotopies for large-scale problems

Since the penalty term $\pi(\mathbf{x}^{\inf})$ in Eq. (44) is a scalar, the equation is not useful as such for large-scale problems. When the homotopy path runs into the bounding zone, the single penalty term focuses the annihilation effect on every equation in the equation set $\mathbf{h}(\mathbf{x}^{\inf}, \theta)$, and also on equations in which bounded variables do not exist. This not only weakens the accuracy, but also makes the Jacobian matrix full, forcing a switch from sparse to dense linear algebra. This admittedly decreases efficiency.

To make the modified bounded homotopies applicable for large and sparse problems requires the modification of the way annihilation is carried out. The penalty term must not surrender the possibility of exploiting the problem sparsity in numerical Jacobian matrix approximation. To fulfil this, the modified bounded homotopies for large-scale problems are expressed as

$$\mathbf{h}_b^{\mathrm{mod}}(\mathbf{x}^{\inf}, \theta) = \mathbf{\Pi}(\mathbf{x}^{\inf})\mathbf{h}(\mathbf{x}^{\inf}, \theta) + \mathbf{f}'(\mathbf{x}^{0,\inf})(\mathbf{x}^{\inf} - \mathbf{x}^{b,\inf}) . \qquad (51)$$

The diagonal penalty matrix $\mathbf{\Pi}(\mathbf{x}^{\inf})$ enables the composition of an individual penalty term for every single equation in the equation set $\mathbf{h}(\mathbf{x}^{\inf}, \theta)$ separately. Thus the annihilation effect can be focused on every equation where a bounded variable exists, and thereby the annihilation becomes more precise and

theoretically acceptable. The elements of the diagonal penalty matrix $\mathbf{\Pi}(\mathbf{x}^{\inf})$ fulfilling this property can be defined as

$$\Pi_{ii}(\mathbf{x}^{\inf}) = 1 - \left\| \mathbf{e}_i^T \left[\mathbf{K}\, diag(\mathbf{e} - \boldsymbol{\pi}(\mathbf{x}^{\inf})) \right] \right\|_{\infty}. \tag{52}$$

$\mathbf{e}_i$ describes a unit vector, whose *i*th element has the value one, and $\mathbf{e}$ describes a vector in which every element has the value one. To make it possible to exploit the sparsity pattern of the original equation set $\mathbf{f}(\mathbf{x})$, the occurrence matrix $\mathbf{K}$ is defined as:

$$\mathbf{K}_{i,j} = \begin{cases} 0, & \mathbf{f}'(\mathbf{x}^0)_{i,j} = 0, \\ 1, & \mathbf{f}'(\mathbf{x}^0)_{i,j} \neq 0. \end{cases} \tag{53}$$

The elements of the vector function $\boldsymbol{\pi}(\mathbf{x}^{\inf})$ can be defined as

$$\pi_i(x_i^{\inf}) = 1 - \rho_1\left(\left| W_{ii}(x_i^{\inf} - x_i^{b,\inf}) \right|, \delta \right), \tag{54}$$

where ρ_1 is the function defined by Eq. (49) and $x_i^{b,\inf}$ defined by Eq. (46).

5.2.4 The concept of bounding the homotopy parameter

In several cases, non-linear equation sets have multiple solutions. Even if the homotopy path ran through every root, it might make long unnecessary curves outside the homotopy parameter space [0 1]. This undoubtedly increases the number of predictor-corrector steps needed and time spent in path tracking. It is also possible that multiple solutions may lie on homotopy path branches, which do not have a real space connection. Even if a connection existed in complex space, the utilization of complex arithmetic would considerably complicate the solving.

To tackle homotopy parameter unboundedness and make it possible to connect separate branches with starting point or solution multiplicities, a homotopy parameter-bounding concept has been presented in Paper II. This concept is based on the following expression:

$$\mathbf{h}_{b\theta}(\mathbf{x}, \theta) = \pi_\theta(\theta)\mathbf{h}(\mathbf{x}, \theta) + \boldsymbol{\upsilon}_\theta(\theta) - \boldsymbol{\upsilon}_\theta(\theta^b). \tag{55}$$

The magnitude of the selected homotopy function $\mathbf{h}(\mathbf{x}, \theta)$ is annihilated with the penalty function $\pi_\theta(\theta)$ whenever the path runs outside the predefined homotopy

parameter space. The auxiliary functions $\mathbf{v}_\theta(\theta)$ and $\mathbf{v}_\theta(\theta^b)$ are utilized to compensate the annihilation.

Since the homotopy parameter θ exists in every equation of the homotopy function $\mathbf{h}(\mathbf{x},\theta)$, it is worth subjecting every equation to both annihilation and compensation . To achieve this, the auxiliary function may be defined as

$$\mathbf{v}_\theta(\theta) = M\theta\,\mathbf{e}\,. \tag{56}$$

$\mathbf{e}$ is a column vector where every element has the value one. The parameter $M \in \left]-\infty\ +\infty\right[$ scales the elements of vector $\mathbf{e}$ properly. Situating Eq. (56) into Eq. (55) gives

$$\mathbf{h}_{b\theta}(\mathbf{x},\theta) = \pi_\theta(\theta)\mathbf{h}(\mathbf{x},\theta) + M(\theta-\theta^b)\mathbf{e}\,. \tag{57}$$

To make the compensation $M(\theta-\theta^b)\mathbf{e}$ reasonable it is worth scaling the equations of the original equation set $\mathbf{f}(\mathbf{x})$ adequately, in order to obtain elements in the equation vector $\mathbf{h}(\mathbf{x},\theta)$ of the same order of magnitude.

Since in problem-independent homotopies the homotopy parameter θ is an artificial parameter without physical meaning, there are no evident boundary values for it. One choice is to force the homotopy parameter to run inside the domain $[\,-1\ \ 2\,]$ and the bounding to occur when the homotopy path runs outside the domain $[\,0\ \ 1\,]$. To attain this, selections of $\delta = 1$ and $\xi = \theta - \theta^{b'}$ in Eqs. (49) and (50) have been made. In this case the parameter θ^b is defined as

$$\theta^b = \theta - \left(6\left|\theta-\theta^{b'}\right|^5 - 15\left|\theta-\theta^{b'}\right|^4 + 10\left|\theta-\theta^{b'}\right|^3\right)\left[\theta-\theta^{b'}\right], \tag{58}$$

and the penalty function generating a scalar value as

$$\pi_\theta(\theta) = 1 - \left(6\left|\theta-\theta^{b}\right|^5 - 15\left|\theta-\theta^{b}\right|^4 + 10\left|\theta-\theta^{b}\right|^3\right). \tag{59}$$

The parameter $\theta^{b'}$ existing in Eq. (58) is defined as

$$\theta^{b'} = \begin{cases} 0, & \theta < 0, \\ \theta, & 0 \le \theta \le 1, \\ 1, & \theta > 1. \end{cases} \tag{60}$$

6 Problem-tailored solving procedures

"A procedure is a specified series of actions or operations which have to be executed in the same manner in order to always obtain the same result under the same circumstances (for example, emergency procedures). Less precisely speaking, this word can indicate a sequence of tasks, steps, decisions, calculations and processes, that when undertaken in the sequence laid down produces the described result, product or outcome. A procedure usually induces a change" (Wikipedia)

As discussed in Section 2.4, flow ratios offer an easy way to obtain a converged result even for complicated separation systems. In this case, however, a considerable amount of manual simulation activity as well as several iteration rounds may be required in order to approach the desired separation target. Therefore, rather than approaching the solution with numerically easy flow ratio specifications, it is desirable to aim to solve the state distribution directly with exact product purity specifications. However, this makes numerical solving more demanding and establishes a need for robust solving methods.

Homotopy continuation methods have a superior convergence property compared to locally convergent solving methods. This means that homotopy methods are able to achieve a solution for an equation set from a substantially larger starting point domain compared to locally convergent solving methods. However, homotopy continuation methods are computationally heavier, requiring considerable computational effort. In simulation this is seen as longer solving times compared to the computationally lighter locally convergent solving methods. In addition, despite a large convergence domain, homotopy continuation methods cannot guarantee that a solution is always obtained. Even though the solving of complex systems can be enhanced by selecting an equation-oriented solving approach rather than a modular-solving approach and by utilizing homotopy continuation methods instead of locally convergent methods, the overall robustness may still be not good enough to guarantee successful solving.

Usually, solving strategies rely largely on the skills of the engineer carrying out the simulation. From the overall robustness point of view, this is not a recommended way of seeking a solution. Since there is often cumulated knowledge about how the solution can be obtained, it is logical to exploit this knowledge systematically when systems possessing the same kind of

characteristics are being solved. To put this into practice, it is worth applying a solving strategy that relies on a problem-tailored solving procedure.

6.1 The purpose of problem-tailored solving procedures

Problem-tailored solving procedures aim to systematize solving by dividing the solving task into ordered phases. The phases are based on certain kinds of heuristic guidelines, which in a step-by-step manner direct the solving in such a way that the desired state distribution is attained, even if the initial guess in the first phase of the solving procedure were relatively far from the solution finally achieved in the last phase of the solving procedure. Thus, problem-tailored solving procedures increase the overall robustness. At their best, problem-tailored solving procedures can be assembled into the form of a solving algorithm so that no intervention is needed between the solving procedure phases.

Depending on the difficulty of the problem, the number of required solving procedure phases may vary. When the complexity of the process increases in the form of a complex structure or non-ideal component mixture, the number of required phases generally increases.

The number of required solving procedure phases may also vary depending on the solving method utilized. With locally convergent solving methods more phases might be needed on the way to the solution than with methods featuring more global convergence.

In addition to phases including non-linear equation set solving tasks, procedures may also include phases in which the considered system is optimized with respect to some optimization criterion. One example of this is the determination of a state distribution of the distillation configuration such that the product purity requirements are fulfilled with the minimum overall energy consumption.

6.2 General procedure for thermally coupled column configurations

A problem-tailored solving procedure is illustrated in Figure 18 that forms the basis for the solving procedures applied in Papers IV–VI. The procedure is intended to solve the state distribution for a distillation configuration whose structure has been determined in advance. The procedure cannot be applied as

such to optimization tasks in which the aim is to determine both the process structure and operation conditions in terms of minimum total annual costs.

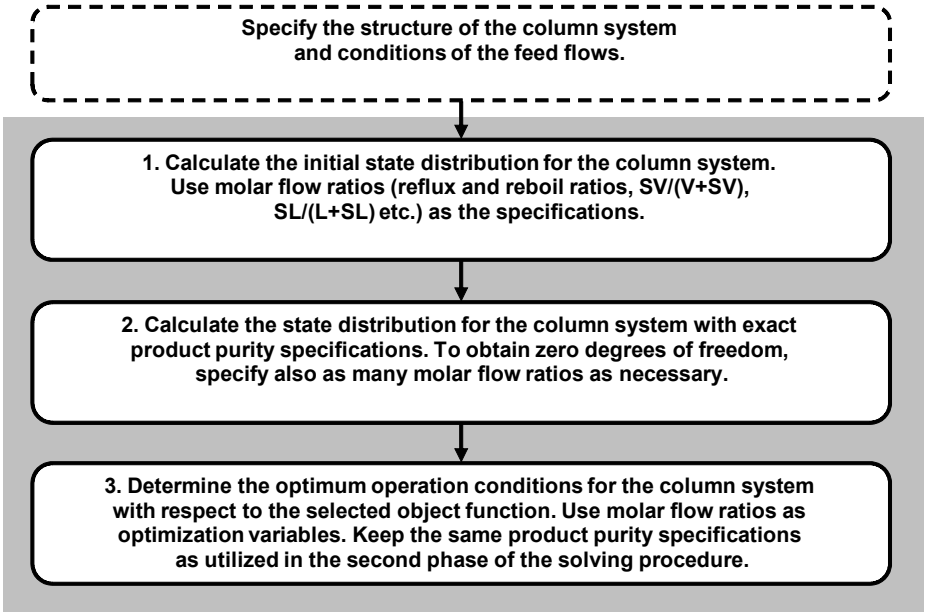

Fig. 18. General problem-tailored solving procedure applied in Papers IV–VI to solve state distribution for thermally coupled column configurations.

In addition to the conditions of the feed flows, i.e. flow rates, compositions, temperatures and pressures, the column configuration structure must be defined before the actual solving procedure phases. The structure includes specifications for the number of stages in the columns, feed and side draw stages, types of condensers, and flow connections between the columns.

In the first actual solving procedure phase, the column configuration model is solved based on flow ratios. Flow ratios may include reflux and reboil ratios, and other flow ratios such as side draw relative to the total flow leaving the side draw stage. Because product flow compositions are functions of internal flows in the column (Kister 1992), flow ratios offer a practicable means to obtain an initial profile for a distillation configuration without any special information about the material distribution in the system.

In the second phase of the solving procedure, the system is solved with exact product purity requirements. Based on the experience obtained with thermally

coupled side-column configurations (Paper V), it is advantageous to carry out solving with exact mole fraction specifications one column end at a time.

In cases where the number of degrees of freedom of the system is higher than the number of product flows in the system, flow ratios may be utilized as additional specifications in the second phase of the solving procedure. The same flow ratios are also utilized as optimization variables in the third phase of the solving procedure, where the performance of the system is minimized or maximized with respect to the object function. The minimum total duty of reboilers has been utilized as a criterion in the optimization studies carried out in Papers IV–VI.

The general problem-tailored solving procedure illustrated in Figure 18 can be supplemented for various applications. For instance, a fourth solving procedure phase may be added in studies where the target is to determine whether an intermediate heat exchanger situated in a thermally coupled side-column configuration decreases the total energy requirement (Paper VI).

7 Performance of the proposed improvements

"The important thing in science is not so much to obtain new facts as to discover new ways of thinking about them." – Sir William Bragg

One objective of this thesis is to illustrate the benefits of modified bounded homotopies as well as the performance of problem-tailored solving procedures on various distillation systems. The issues affecting fundamental robustness have been taken into particular consideration in the studies carried out in Papers I–VI.

The path tracking algorithm as well as the models of the considered column configurations and the thermodynamic correlations utilized have been implemented in the MATLAB environment. Thus the development and testing is transparent and independent of the restrictions set by commercial process simulation and non-linear equation set solving packages.

7.1 Implementation of path tracking strategy

The implemented predictor-corrector path tracking strategy exploits the routines of arc length type parameterization presented in Section 4.2.2. In the predictor phase, the tangent vector $\mathbf{z}^{(k)}$ is normalized with respect to the homotopy parameter, in which case Eq. (28) takes the form:

$$\begin{bmatrix} \left(\dfrac{\partial \mathbf{h}}{\partial \mathbf{x}}\right)^{(k)} & \left(\dfrac{\partial \mathbf{h}}{\partial \theta}\right)^{(k)} \\ \mathbf{e}_{n+1}^{T} & \end{bmatrix} \mathbf{z}^{(k)} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix}. \tag{61}$$

The elements of the solved tangent vector $\mathbf{z}^{(k)}$ are divided by its Euclidian norm according to Eq. (29). The unit tangent vector $\mathbf{u}^{(k)}$ obtained is then substituted into the first-order Euler predictor Eq. (30).

To mimic arc length path tracking as precisely as possible, the auxiliary vector $\mathbf{q}$ is introduced as $\mathbf{q}^{(0)} = \mathbf{u}^{(k)}$. The predicted point $(\bar{\mathbf{x}}^{(k+1)}, \bar{\theta}^{(k+1)})$ is corrected based on corrective iterations, which are carried out according to the following four steps:

$$\begin{bmatrix} \left(\dfrac{\partial \mathbf{h}}{\partial \bar{\mathbf{x}}}\right)^{(i)} & \left(\dfrac{\partial \mathbf{h}}{\partial \bar{\theta}}\right)^{(i)} \\ \mathbf{q}^{(i-1)} & \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ \Delta \theta \end{bmatrix}^{(i)} = \begin{bmatrix} -\mathbf{h}(\bar{\mathbf{x}}^{(i)}, \bar{\theta}^{(i)}) \\ 0 \end{bmatrix}, \tag{62}$$

$$\mathbf{q}_{unscaled}^{(i)} = \lambda^{(k)}\mathbf{q}^{(i-1)} + \begin{bmatrix}\Delta\mathbf{x}\\ \Delta\theta\end{bmatrix}^{(i)}, \tag{63}$$

$$\mathbf{q}^{(i)} = \frac{1}{\left\|\mathbf{q}_{unscaled}^{(i)}\right\|_2}\mathbf{q}_{unscaled}^{(i)} \tag{64}$$

and

$$\begin{bmatrix}\bar{\mathbf{x}}\\ \bar{\theta}\end{bmatrix}^{(i+1)} = \begin{bmatrix}\mathbf{x}\\ \theta\end{bmatrix}^{(k)} + \lambda^{(k)}\mathbf{q}^{(i)}. \tag{65}$$

The corrector iterations are continued until the Euclidian norm of $\mathbf{h}(\bar{\mathbf{x}}^{(i+1)}, \bar{\theta}^{(i+1)})$ is under the error tolerance.

As illustrated in Figure 19, the corrector phase iterations carried out in the way described above guarantee that the distance between the corrected steps $(\mathbf{x}^{(k)}, \theta^{(k)})$ and $(\mathbf{x}^{(k+1)}, \theta^{(k+1)})$ is exactly $\lambda^{(k)}$, i.e. the desirable step length. Thus over-lengthened steps are avoided.

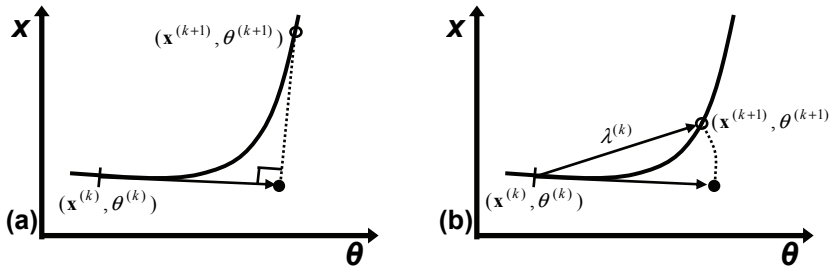

Fig. 19. The characteristics of two predictor-corrector path tracking routines. (a) The strategy presented in Section 4.2.2 and (b) the routine implemented in this thesis.

In order to ensure robust path tracking, a tight error tolerance has been utilized in simulations. Thus short steps have been taken and thereby potential failures in path tracking have been minimized.

The applied solving algorithm exploits the Jacobian matrix sparsity pattern based on the tools offered by MATLAB. Not only has the utilization of sparse matrix routines diminished the memory requirement, but it has also rationalized the computation. The number of function evaluations required in finite-difference based Jacobian matrix approximation has been significantly reduced.

Usually, several Jacobian matrix approximations are required in the corrector phase of the predictor-corrector path tracking routine. Since corrector iterations

need to be performed essentially at every step, the possibility to save computational effort in the corrector phase is an attractive option (Allgower & Georg 2003). The number of function evaluations could be reduced and the Jacobian matrix approximation accelerated by implementing a Jacobian matrix updating method, such as Schubert's method.

Even though the robustness and efficiency of the path tracking algorithm are admittedly important issues, they are considered more as practical implementation problems, not fundamental ones. Therefore, the robustness and efficiency of the implemented solving algorithm do not belong to the core focus of this thesis.

7.2 Robustness of the modified bounded homotopies

7.2.1 Homotopy path bounding with respect to the problem variables

Even though the traditional homotopy continuation methods are quite robust in general, they may fail when the path runs out of the problem definition domain. The solving may also fail due to the homotopy path striking the problem domain boundary. In this case a predictor or corrector step in the predictor-corrector path tracking routine may fall outside the problem domain.

In principle, crossing the problem domain boundary is not dangerous. However, the substitution of an unfeasible variable value into a thermodynamic subroutine may cause a fatal error and thus cut off path tracking. In addition, the extrapolation of physical and chemical properties outside the domain of data should be avoided.

In order to keep the homotopy path inside the predefined problem domain, modified bounded homotopies have been exploited in Papers I and III–VI. In addition to simple mathematical test examples, distillation column examples have been studied to examine the performance of modified bounded homotopies.

Tackling boundary striking by means of variables mapping

Even if the homotopy path would not cross the problem domain boundary, it may run very close to the boundary. This is illustrated in Figure 20, where the mole fraction of acetone runs close to zero.

Challenges encountered close to the domain boundary can be tackled by implementing variables mapping, in accordance with Section 5.2.1. This mapping

brings sensitivity to path tracking close to the problem domain boundaries and thus makes the path tracking accurate and flexible. Unfeasible variable value predictions outside the problem domain and thus fatal errors potentially generated by thermodynamic calculation routines can be avoided completely.

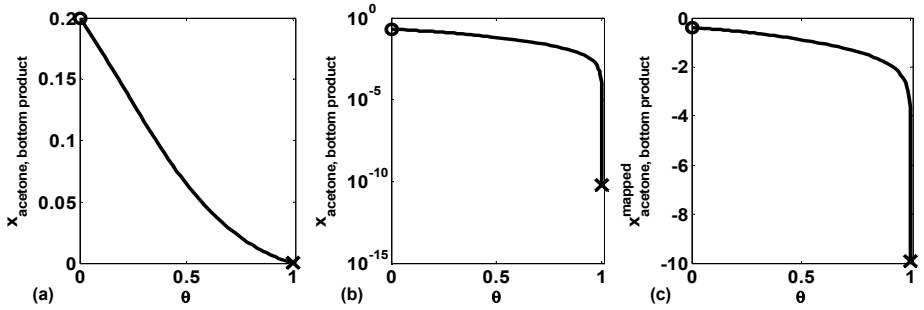

Fig. 20. Homotopy paths for the mole fraction of acetone in the bottom product flow of Case 1 of Paper III. (a)–(b) Homotopy path in unmapped variable space, and (c) path in mapped variable space.

Preventing problem domain crossings

As illustrated by the sidestream column examples in Papers I and III, variables mapping together with modified bounded homotopies make it possible to restrict the homotopy path so that it runs only inside the predefined problem domain. The path can be kept bounded even though the bounding zone is kept narrow.

By tracking the path in the mapped variable space instead of the unmapped one, the numerical values of the auxiliary function elements in Eq. (36) become significant and, in contrast to the bounded homotopies presented by Paloschi, erroneous solutions inside the narrow bounding zone are avoided.

Tackling the total unboundedness of the homotopy path

Bounded homotopies prevent the homotopy path from running to infinity and thereby becoming totally unbounded. However, when unbounded homotopy path branches run into opposite infinity directions, bounded homotopies are not able to form a trackable path between these branches. In this case homotopy path bounding with respect to the problem variables does not bring any advantage. This situation is illustrated in Figure 21a. Correspondingly, Figure 21b illustrates the situation where the homotopy path branches run to the same infinity direction.

In this case bounded homotopies are usable by offering a way to connect the branches.

In order to tackle the challenge illustrated in Figure 21a, some kind of branch jumping technique, like that proposed for example by Christiansen *et al.* (1996), would be required.

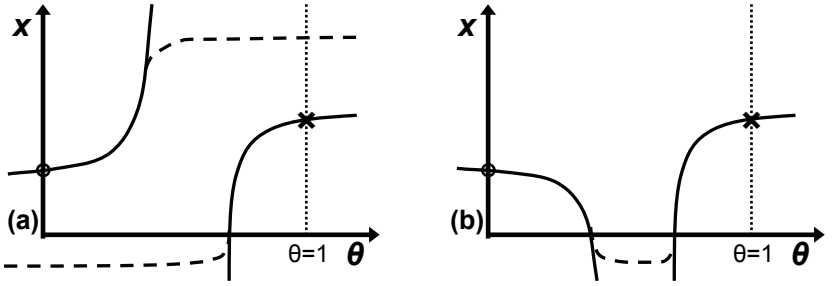

Fig. 21. Separate homotopy path branches running through the starting point (○) and the solution (x). (a) The starting point and the solution lie on separate homotopy path branches and run to the opposite infinity directions. (b) The starting point and the solution lie on separate homotopy path branches and run to the same infinity direction.

Challenge posed by an extremely narrow bounding zone

When the aim is to narrow the width of the bounding zone so that none of the actual solutions fall inside the bounding zone, numerical challenges may appear in path tracking. Even though homotopy path tracking in the mapped variable space makes path tracking accurate and flexible inside a narrow bounding zone, numerical problems appear when the path is getting close to or diverging from a narrow bounding zone. As illustrated in Paper I, in these cases elements of the Jacobian matrix may become nearly singular. Even though the analytical values of the Jacobian matrix elements would not be exactly zero, the imprecision in numerical Jacobian matrix approximation would cause singularities.

As a general conclusion, it is not possible to narrow the bounding zone endlessly in order to prevent the actual solution from lying inside the bounding zone. There must be a limit for the bounding zone width set by the precision of the computing machine that should not be crossed.

7.2.2 Homotopy path bounding with respect to the homotopy parameter

Occasionally some real space solutions lie on separate homotopy path branches. When the selected (trivial) starting point and the actual solution of the problem lie on separate branches, it is not possible to approach the root from the starting point.

As illustrated in Figure 22, by bounding the homotopy path outside the predefined homotopy parameter space [0 1], the homotopy path branches can be forced to bend in such a way that the separate branches are connected. Thus, the solutions on separate branches can be approached from a single starting point by tracking the path in real space. Correspondingly, starting point and solution isolas can be broken and connected to other branches. By connecting the separate branches, the ability of approaching one or more otherwise unattainable solutions from a single starting point only based on real space arithmetic is undoubtedly improved.

The ability to connect separate homotopy path branches is useful when determining:

- starting point multiplicities of the Newton homotopy, and
- solution multiplicities.

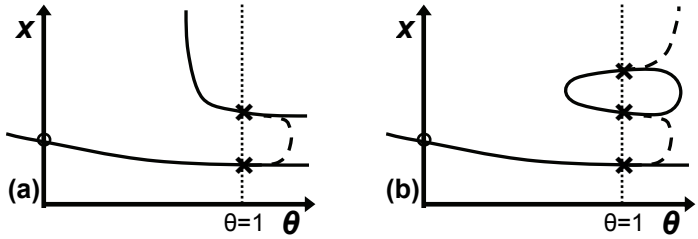

Fig. 22. Advantages achieved by bounding the homotopy path with respect to the homotopy parameter. (a) Connecting separate homotopy branches, and (b) breaking the isola branch and connecting it to another branch.

Determination of starting point multiplicities

The Newton homotopy method may have several solutions at $\theta = 0$. In this case it may happen that there is no trackable real space path from the selected starting point to the actual solution.

As is illustrated by a sidestream column (Case 3 of Paper III), the actual solution may be approached from the other starting point multiplicities, but not from the trivial one. As shown in Paper II, the described starting point multiplicity problem can be tackled by bounding the homotopy path with respect to the homotopy parameter. By properly defining the absolute value and sign of the parameter M in Eq. (57), the connections between starting point multiplicities can be formed or broken.

By offering a way to determine one or more starting point multiplicities lying on separate homotopy path branches, the overall ability of approaching at least one and possibly several actual solutions is enhanced. Thus the implementation of the homotopy parameter bounding concept improves the robustness of the Newton homotopy method.

Determination of solution multiplicities

The concept of homotopy parameter bounding can also be utilized in the determination of solution multiplicities. As demonstrated in Paper II, the concept enlarges the starting point domain from where it is possible to approach multiple solutions with homotopy methods. This reduces the need for numerous starting points in multiplicity studies.

Because the concept of homotopy parameter bounding makes it possible to approach multiple solutions by relying only on real space arithmetic, the concept is well suited to chemical engineering computations.

Solution multiplicities on unreachable isolas

Even though the domain from where the homotopy methods may converge to one or more solutions can be enlarged by utilizing the concept of homotopy parameter bounding, the solution isolas lying at $\theta = 1$ are still unreachable unless there is a trackable real space path from the starting point at $\theta = 0$ to one of the solutions at $\theta = 1$. This situation is illustrated in Figure 23b.

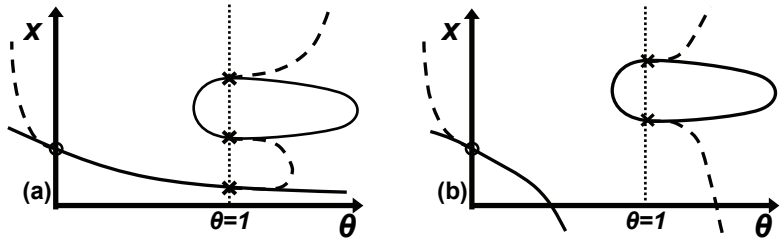

Fig. 23. The achievability of solutions on isola. (a) Solutions on isola can be approached by bounding the homotopy path with respect to the homotopy parameter, and (b) isola solutions cannot be approached by bounding the homotopy path only with respect to the homotopy parameter.

7.3 Overall robustness in solving distillation configurations

Even if the solving method itself is relatively robust, it may not bring any benefits unless the solving strategy is well formulated. Therefore, in order to improve the overall robustness, problem-tailored solving procedures are worthwhile.

7.3.1 Solving based on modified bounded homotopies

Modified bounded homotopies together with the variables mapping improve the robustness of homotopy continuation methods by keeping problem variables inside the prescribed problem domain. Thus fatal errors caused by unfeasible variable values are avoided.

As illustrated in Papers I and III, modified bounded homotopies may make it possible to obtain the solution directly with exact mole fraction specifications even though the starting point is trivial. In fact, it requires no special initial state distribution profile for the column system. As illustrated in Paper II, no special initial state distribution profile is required when the starting point and solution multiplicities are determined based on homotopy methods that exploit the concept of homotopy parameter bounding.

Despite the improved robustness, modified bounded homotopies are not able to tackle all the problems existing in non-linear equation set solving. Perhaps the most challenging is the problem of solution multiplicities on unreachable isolas (Section 7.2.2). In addition, homotopy path bounding with respect only to the problem variables (Section 7.2.1) does not bring benefits when the unbounded homotopy path branches run to opposite infinities.

7.3.2 Solving based on problem-tailored solving procedures

Because the solving methods developed for solving non-linear equation sets still have fundamental problems in their robustness, problem-tailored solving procedures have been applied in Papers IV–VI. In these papers, fully thermally coupled column configuration (the Petlyuk system) and thermally coupled side-rectifier and side-stripper configurations have been studied in crossing a distillation boundary.

By incorporating heuristically justified guidelines in the form of problem-tailored solving procedure, the solution may be approached straightforwardly even with a locally convergent solving method. However, as an example in Paper V illustrates, a locally convergent solving method may not allow successful convergence within a reasonable number of iterations or even not allow it at all. In this case, modified bounded Newton homotopy can be utilized to enlarge the convergence domain.

7.4 Overall findings and discussion

The simulations carried out in Papers IV–VI show how thermally coupled column configurations are able to cross the distillation boundary from the concave side. Even though the crossing with the fully thermally coupled column system (the Petlyuk system) is only moderate (Figure 24a), the crossing with thermally coupled side-column configuration is significant (Figure 24b).

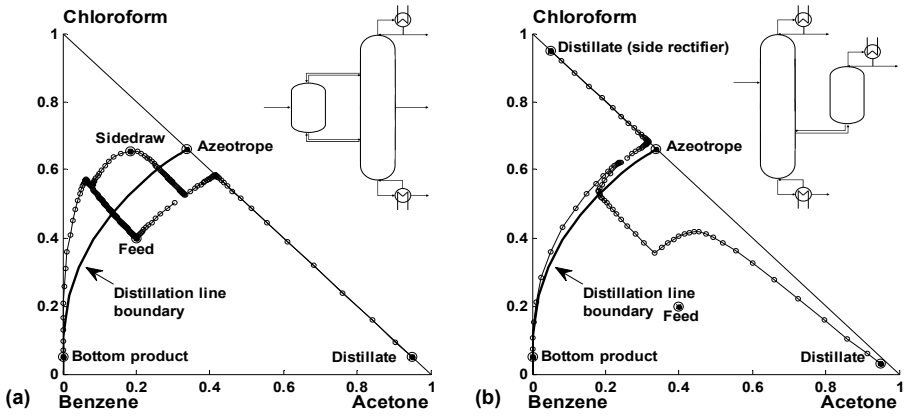

Fig. 24. Distillation boundary crossing by (a) the fully thermally coupled column configuration (Case 13 of paper IV), and (b) the thermally coupled side-rectifier (Case 6 in Table 3 of Paper V).

The simulations carried out in Papers V–VI also show that thermally coupled side-column configurations make it possible to cross the distillation boundary with higher product flow purities than is possible with conventional direct and indirect column sequences. In addition, implementing an intermediate heat exchanger into the thermally coupled side-column configurations may decrease the energy requirement with respect to the conventional column configuration.

On the basis of the general trend of increasing labour costs against decreasing computation costs, it is supposed that interest in improving the robustness of solving methods and utilization of problem-tailored solving procedures will increase in process simulation. Thus, even though the bounded homotopies and problem-tailored solving procedures increase the focus towards the solving resources (hardware), approaching the same simulation target in a more manual way might be even more expensive.

8 Conclusions and suggestions for future research

As illustrated in Papers I–III, boundary striking, the homotopy path exceeding the problem definition domain, the total unboundedness of the homotopy path, and the existence of starting point multiplicities are the fundamental causes of failure in the problem-independent homotopy continuation methods, which can be tackled by utilizing modified bounded homotopies with variables mapping. Since the problem of turning point singularity can be simply tackled with arc length parameterization, the occurrence of isolated solutions is the only issue that has not been satisfactorily solved to date.

It is argued that at present there is no absolutely robust problem-independent solving method that would always guarantee approaching the solution for a complex column configuration from an arbitrary starting point. However, as illustrated in Papers IV–VI, utilizing problem-tailored solving procedures can enhance solving by offering well-ordered phases for the solving algorithm. The phases are based on heuristic guidelines, which direct the solving towards the state distribution that exactly fulfils the product purity specifications set for the system.

Based on the results summed up in this thesis, it can be concluded that modified bounded homotopies together with problem-tailored solving procedures improve the *overall robustness* of an equation-oriented solving approach. The improved overall robustness diminishes the need for intensive manual simulation activity, thus rationalizing simulation work. Therefore, enhanced robustness also indirectly improves *overall efficiency*. This means that the state distribution for a process with the desired specifications is obtained faster with more robust solving methods and problem-tailored solving procedures than without. As a whole, improved robustness is expected to lower the threshold of designing and implementing novel process configurations.

At the moment, the proposed modified bounded homotopies and problem-tailored solving procedures have not been implemented in any special flowsheeting package. Even though the basics have largely been set out in this thesis, the practical implementation aspects in particular require some effort before the fruits of this work can be harnessed in engineering practice. For instance, consolidation of the homotopies developed with a robust and efficient homotopy path tracking algorithm would be advantageous.

In addition, the incorporation of the branch-jumping technique into modified bounded homotopies could be fruitful. Particular attention must be directed to tackling the problem posed by solution isolas, which cannot be approached systematically either by the traditional homotopies or the bounded homotopies proposed so far.

The application of modified bounded homotopies is justified especially for numerically challenging process models, where the structure of the process system is complex and/or component mixtures are highly non-ideal in thermodynamic terms. In addition to separation systems, reactive systems and systems combining reaction and separation in the same process unit certainly represent applications where the benefits offered by modified bounded homotopies in the form of enhanced solution and improved possibility of determining multiple steady-states can be exploited.

References

Allgower EL & Georg K (2003) Introduction to numerical continuation methods. Philadelphia PA, Society for Industrial and Applied Mathematics.

Barton PI (2000) The equation oriented strategy for process flowsheeting. Massachusetts Institute of Technology, Chemical Engineering System Research Group. URI: http://yoric.mit.edu/abacuss2/Lecture1.pdf. Cited 4.3.2008.

Bekiaris N & Morari M (1996) Multiple steady states in distillation: ∞/∞ predictions, extensions, and implications for design, synthesis, and simulation. Industrial & Engineering Chemistry Research 35(11): 4264–4280.

Bogle IDL (1983) The numerical solution of flowsheeting problems characterised by large sparse Jacobian matrices. Doctoral thesis, London, Imperial College of Science and Technology.

Bogle IDL & Perkins JD (1988) Sparse Newton-like methods in equation oriented flowsheeting. Computers & Chemical Engineering 12(8): 791–805.

Chavez CR, Seader JD & Wauburn TL (1986) Multiple steady-state solutions for interlinked separation systems. Industrial & Engineering Chemistry Fundamentals 25: 566–576.

Choi SH, Harney DA & Book NL (1996) A robust path tracking algorithm for homotopy continuation. Computers & Chemical Engineering 20(6–7): 647–655.

Christiansen AC, Morud J & Skogestad S (1996) A comparative analysis of numerical methods for solving systems of nonlinear algebraic equations. Proc. of the 38th SIMS Simulation conference, June 11–13. Trondheim, Norway: 217–230.

Dalal NM & Malik RK (2003) Solution multiplicities in multicomponent distillation column. A computational study. In Kraslawski A & Turunen I (eds) Proceeding of ESCAPE–13. Elsevier: 617–622.

Dennis JE & Schnabel RB (1983) Numerical methods for unconstrained optimization and nonlinear equations. Englewood Cliffs NJ, Prentice-Hall.

Doherty MF & Malone MF (2001) Conceptual design of distillation systems. New York, McGraw-Hill.

Duff IS, Erisman AM & Reid JK (1986) Direct methods for sparse matrices. Oxford, Clarendon.

Fidkowski ZT, Doherty MF & Malone MF (1993) Feasibility of separations for distillation of nonideal ternary mixtures. AIChE Journal 39(8): 1303–1321.

Gritton KS, Seader JD & Lin W (2001) Global homotopy continuation procedures for seeking all roots of a nonlinear equation. Computers & Chemical Engineering 25(7–8): 1003–1019.

HYSYS manual (2007) Aspen HYSYS: Simulation basis. Aspen Technology Inc.

Henley EJ & Seader JD (1981) Equilibrium-stage separation operations in chemical engineering. New York NY, Wiley.

Holland CD (1981) Fundamentals of multicomponent distillation. New York, McGraw-Hill.

Kannan A, Joshi MR, Reddy GR & Shah DM (2005) Multiple-steady-states identification in homogeneous azeotropic distillation using a process simulator. Industrial & Engineering Chemistry Research 44(12): 4386–4399.

Kelley CT (1995) Iterative methods for linear and nonlinear equations. Philadelphia PA, Society for Industrial and Applied Mathematics.

King CJ (1980) Separation processes. New York, McGraw-Hill.

Kister HZ (1992) Distillation design. New York, McGraw-Hill.

Kovach JW (1987) Heterogenous azeotropic distillation–homotopy-continuation methods. Computers & Chemical Engineering 11(6): 593–605.

Krolikowski LJ (2006) Determination of distillation regions for non-ideal ternary mixtures, AIChE Journal 52(2): 532–544.

Kuno M & Seader JD (1988) Computing all real solutions to systems of nonlinear equations with a global fixed-point homotopy. Industrial & Engineering Chemistry Research, 27(7): 1320–1329.

Lin W-J, Seader JD & Wayburn TL (1987) Computing multiple solutions to systems of interlinked separation columns. AIChE Journal 33(6): 886–897.

Paloschi JR (1995) Bounded homotopies to solve systems of algebraic nonlinear equations. Computers & Chemical Engineering 19(12): 1243–1254.

Paloschi JR (1996) Bounded homotopies to solve systems of sparse algebraic nonlinear equations. Computers & Chemical Engineering 21(5): 531–541.

Paloschi JR (1998) Using sparse bounded homotopies in the SPEEDUP simulation package. Computers & Chemical Engineering 22(9): 1181–1187.

Perry (2008) Perry's chemical engineers' handbook. 8th ed. New York, McGraw-Hill.

Poling BE, Prausnitz JM & O'Connell JP (2001) The properties of gases and liquids. Boston, McGraw-Hill.

Richter S (1983) Continuation methods: Theory and applications. IEEE Transactions on Circuits and Systems 30(6): 347–352.

Rooks RE, Malone MF & Doherty MF (1996) A geometric design method for side-stream distillation columns. Industrial & Engineering Chemistry Research 35(10): 3653–3664.

Seader JD & Henley EJ (2006) Separation process principles. New York, John Wiley & Sons Inc.

Seader JD, Kuno M, Lin W-, Johnson SA, Unsworth K & Wiskin JW (1990) Mapped continuation methods for computing all solutions to general systems of nonlinear equations. Computers & Chemical Engineering 14(1): 71–85.

Seider WD, Brengel DD & Widagdo S (1991) Nonlinear analysis in process design. AIChE Journal 37(1): 1–38.

Seider WD, Seader JD & Lewin DR (2004) Product and process design principles: synthesis, analysis and evaluation. 2nd ed. New York, Wiley.

Seydel R (1988) From equilibrium to chaos: practical bifurcation and stability analysis. New York, Elsevier.

Seydel R & Hlavacek V (1987) Role of continuation in engineering analysis. Chemical Engineering Science 42(6): 1281–1295.

Stichlmair JG & Herguijuela J-R (1992) Separation regions and processes of zeotropic and azeotropic ternary distillation. AIChE Journal 38(10): 1523–1535.

Thong DY-C & Jobson M (2001) Multicomponent homogeneous azeotropic distillation 1. Assessing product feasibility. Chemical Engineering Science 56(14):4369–4391.

Taylor R (2007) (Di)Still Modeling after All These Years: A View of the State of the Art. Industrial & Engineering Chemistry Research 46(13): 4349–4357.

Vadapalli A & Seader JD (2001) A generalized framework for computing bifurcation diagrams using simulation programs. Computers & Chemical Engineering 25(2–3): 445–464.

Wahnschafft OM, Koehler JW, Blass E & Westerberg AW (1992) The product composition regions of single-feed azeotropic distillation columns. Industrial & Engineering Chemistry Research 31(10): 2345–2362.

Wahnschafft OM, Le Rudulier J-P & Westerberg AW (1993) A problem decomposition approach for the synthesis of complex separation processes with recycles. Industrial & Engineering Chemistry Research 32(6): 1121–1141.

Wayburn TL & Seader JD (1987) Homotopy continuation methods for computer-aided process design. Computers & Chemical Engineering 11(1): 7–25.

Westerberg AW (1979) Process flowsheeting. Cambridge, Cambridge UP.

Westerberg AW & Chien HH (1984) Proceedings of the Second International Conference on Foundations of Computer-Aided Process Design. Ann Arbor MI, CACHE.

Wikipedia (2010) Procedure. URI: http://en.wikipedia.org/wiki/Procedure_(term). Cited: 17.5.2010.

Yeomans H & Grossmann IE (2000) Optimal Design of Complex Distillation Columns Using Rigorous Tray-by-Tray Disjunctive Programming Models. Industrial & Engineering Chemistry Research 39(11): 4326–4335.

Original papers

I Malinen I & Tanskanen J (2008) Modified bounded homotopies to enable a narrow bounding zone. Chemical Engineering Science 63(13): 3419–3430

II Malinen I & Tanskanen J (2010) Homotopy parameter bounding in increasing the robustness of homotopy continuation methods in multiplicity studies. Computers & Chemical Engineering 34(11): 1761–1774

III Malinen I & Tanskanen J (2007) Modified bounded Newton homotopy method in solving sidestream column configurations. In: Plesu V & Agachi PS (Eds.) Proceeding of 17th European Symposium on Computer Aided Process Engineering (ESCAPE–17), May 27–30, Bucharest, Romania. CD-ROM

IV Malinen I & Tanskanen J (2007) A rigorous minimum energy calculation method for a fully thermally coupled distillation system. Chemical Engineering Research and Design 85(A4): 502–509

V Malinen I & Tanskanen J (2009) Thermally coupled side-column configurations enabling distillation boundary crossing. 1. An overview and a solving procedure. Industrial & Engineering Chemistry Research 48(13): 6387–6404

VI Malinen I & Tanskanen J (2009) Thermally coupled side-column configurations enabling distillation boundary crossing. 2. Effects of intermediate heat exchangers. Industrial & Engineering Chemistry Research 48(13): 6372–6386

Original papers are not included in the electronic version of the dissertation.

Book orders:
Granum: Virtual book store
http://granum.uta.fi/granum/

UNIVERSITY OF OULU P.O.B. 7500 FI-90014 UNIVERSITY OF OULU FINLAND

ACTA UNIVERSITATIS OULUENSIS

ISBN 978-951-42-9337-5 (Paperback)
ISBN 978-951-42-9338-2 (PDF)
ISSN 0355-3213 (Print)
ISSN 1796-2226 (Online)

UNIVERSITY of OULU
OULUN YLIOPISTO