

The ChemSep Book

Second Edition

Harry A. Kooijman

Amsterdam, The Netherlands

kooijman@chemsep.org

Ross Taylor

Clarkson University, Potsdam, New York

taylor@chemsep.org

Copyright © 2007, 2019 by H.A. Kooijman and R. Taylor. All rights reserved.

This publication is public domain. No responsibility is assumed by the authors for any injury and/or damage to persons or property as a matter of products liability, negligence, or otherwise, or from any use or operation of any methods, products, instructions, information, or idea's contained in the material herein. The use of general descriptive names, registered names, trademarks, etc. in this publication does not imply, even in absence of specific statement, that such names are exempt from the relevant protective laws and regulation and therefore free for general use.

Contents

Preface	7
Part A. Using ChemSep	1
1 A Simple Equilibrium Stage Column	3
1.1 A Short Tour of the <i>ChemSep</i> Interface	3
1.2 Distillation of Benzene and Toluene	9
1.3 Results	18
1.4 Real stages	33
1.5 Parametric Study	37
1.6 Exercises	38
2 Nonequilibrium Columns	45
2.1 Equilibrium and Nonequilibrium Models	45
2.2 A Sieve Tray Column	54
2.3 A Depropanizer	64
2.4 A Packed Absorber	69
2.5 Extractive Distillation	74
2.6 Exercises	74
2.7 Model Uncertainties	77
3 CAPE-OPEN Flowsheet Simulations with <i>ChemSep</i>	79
3.1 Using the COCO Simulator	79
3.2 Flowsheet Examples	94
3.3 Di-Ethyl Ether from Ethanol-Water Feed	103
4 Analysis of Separation Processes	109
4.1 Plot Data File	109
4.2 Parametric Study	114
4.3 Column Rating	117
4.4 Stream Curves	119
4.5 Phase Diagrams	119
4.6 Residue Curve Maps	126
4.7 Property Diagrams	126

4.8	References	128
5	Databanks	131
5.1	Pure Component Data (PCD)	131
5.2	Pure Component Libraries (LIB)	131
5.3	Group Contribution Data (GCD)	132
5.4	Interaction Parameter Data (IPD)	133
5.5	Column Internals Parameter Libraries (INP)	135
6	Tools and Settings	137
6.1	Commandline Options	137
6.2	Interface Settings and Files	138
6.3	Tools and Tool Configuration	141
6.4	Design Mode Settings	142
6.5	ChemSep Internals and Model Development	146
7	Reactive Distillation	171
8	Extraction	173
9	Three Phase Distillation	175
Part B. Technical Reference		177
10	Property Models	179
10.1	Thermodynamic Properties	179
10.2	Physical Properties	193
11	Numerical Solution of Systems of Equations	221
11.1	Repeated Substitution	221
11.2	Newton's Method	221
11.3	Homotopy-Continuation Methods	224
12	Flash Calculations	237
12.1	Equations	238
12.2	Solution of the Flash Equations	239
12.3	The Rachford-Rice Equation	240
13	Equilibrium Stage Simulation	243
13.1	Introduction	243
13.2	The Equilibrium Stage Model	245
13.3	Tearing Methods	251
13.4	Simultaneous Correction Methods	262
13.5	Continuation Methods for Difficult Problems	272
13.6	Relaxation Methods	279
13.7	Three Phase Distillation	280
13.8	Reactive Distillation	286

14 Nonequilibrium Column Modelling	299
14.1 The Model Equations	299
14.2 Model Requirements	306
14.3 Flow Models	307
14.4 The Design Mode	308
15 Design Models for Trayed Columns	313
15.1 Mass Transfer Coefficient Correlations	313
15.2 Pressure Drop Models	316
15.3 Entrainment and Weeping	319
15.4 Packing Flooding and Minimum Wetting	319
15.5 Column Design	320
15.6 Comparison with Experimental Distillation Data	323
16 Design Models for Packed Columns	339
16.1 Mass Transfer Coefficient Correlations	339
16.2 Pressure Drop Models	344
16.3 Entrainment and Weeping	348
16.4 Packing Flooding and Minimum Wetting	348
16.5 Column Design	349
17 Design Models for Extraction Columns	357
17.1 Introduction	357
17.2 Sieve Tray Columns	358
17.3 Packed Columns	362
17.4 Rotating Disk Contactors	364
17.5 Spray Columns	368
17.6 Modeling Backflow	370
18 Nonequilibrium Distillation Dynamics	375
References	377
Index	377

Preface

This book is a *companion* to *ChemSep*, a set of programs that perform multicomponent separation processes simulations. It is *not* designed to be a textbook for teaching separation processes. Instead, the reader will find a guided tour on how to use *ChemSep*, many examples solved with the help of *ChemSep*, and a complete technical reference with detailed descriptions of all the models and algorithms used in the programs. We hope this book provides the *ChemSep* user with all the information needed to make good use of the program.

The *ChemSep* project was started in February 1988 at the University of Technology Delft in the Netherlands, by Arno Haket, Harry Kooijman, and Ross Taylor. *ChemSep* was designed to be easy to use by students with no experience of engineering software, while having sufficient flexibility and power to appeal to expert users. In pursuit of these objectives *ChemSep* features a menu-driven, user-friendly interface with an integrated help system and an autopilot mode that leads the novice user through the data input phase. In March of 1991, when the nonequilibrium model was added, the programs and user interface were completely rewritten. Over the years various extensions of the nonequilibrium model were published and added: an extractor model, various dynamic models, a (reactive) cell model, a three phase distillation model, and a total reflux model (for simulating distillation test data). In 2004 a start was made with a new windows GUI that is still being further developed to include all the column models. In 2005 *ChemSep* became Cape-Open compliant. As a result it is now possible to use the program inside flowsheeting tools such as Aspen Plus (Aspentech), PRO/II (SimSci/Escor) and COCO (Amsterchem). In 2006 we made a LITE version of the Cape-Open version of *ChemSep* available so that anybody can test-drive the program. The LITE version is included in the COCO flowsheeting package (<http://www.cocosimulator.org>).

We continue to develop the simulators to enable more accurate simulation of industrial chemical and petroleum column operations. We do this by developing and adding published thermodynamic and physical property models as well as by developing and implementing of published mass transfer models. Please visit the *ChemSep* website at <http://www.chemsep.org> for more information. Since *ChemSep* is a work in progress, so must be this book. We welcome your suggestions on how the material presented here can be improved.

Harry Kooijman and Ross Taylor

Part A

Using ChemSep

Chapter 1

A Simple Equilibrium Stage Column

In this chapter we take you on a quick tour of *ChemSep* by showing you how to input the specifications for the simulation of a simple distillation column for a simple binary separation. We analyse the results of this example by looking at the various plots *ChemSep* offers, including McCabe-Thiele diagrams.

1.1 A Short Tour of the *ChemSep* Interface

When you start *ChemSep* you first obtain an introduction screen that disappears after a couple of seconds. *ChemSep* starts automatically with a 'new' problem where you are asked to enter a descriptive title for the simulation. The user interface is divided into a main menu, a tool bar, a tree which lists various input sections on the left and the first input section (with a tabbed screen) for the title and optional comments on the right, previous/next buttons, and a status line.

Note that in the tree list there are icons next to the items. In the case of the 'title' there is a gray checkmark in front meaning that this item is optional. Other items have a red cross meaning that they still require further specification. The interface shows you the next item that needs to be specified by labeling or coloring it red (the actual color can be customized). Here, as soon as we type something in for the title input box, the checkmarks in front of 'title' (both in the tree list as well as on the tab) change from gray to green, which means that the specification has been completed for this tabscreen.

In *ChemSep* we follow the philosophy that we only show you what you need to see. Tabsheets

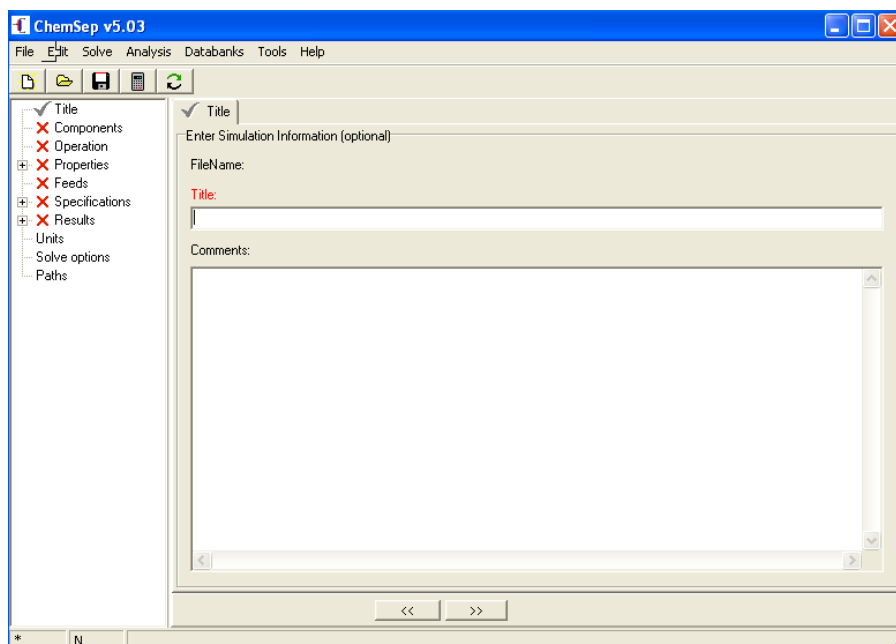


Figure 1.1: *ChemSep* opens with a new problem.

with specification have either a red cross (action required) or a green checkmark (no further action required). Tabsheets without icons contain results or optional inputs. Note that the cross next to the results only indicates whether or not any results are present. You can navigate through the sheets by means of the next (*F6*) and previous (*F5*) buttons on the bottom, or by clicking on the tabs, or by clicking on the items in the tree listing on the left of the interface.

When you inspect the main menu you notice there is no 'window' item in the main menu. This is because for each problem you want to solve, we instantiate a separate window. This has the advantage that you can switch problems with the *Alt-Tab* windows key, similarly as other MS Office products behave. Each of the windows behaves completely autonomically, that is, they do not share memory space and define each their own problem. Do note that when you save the interface settings, you do this globally. However, as the problems already running are not affected, you would need to close all other windows before you'd see the changes in the other windows. The definitions of the units of measure, path information, and the solve options are stored in the global options settings as well as locally in the sep-file for the problem.

1.1.1 Keys

Just like in any other windows program, *ChemSep* assigns several default function key behaviors: *F1* calls up (context sensitive) help and *F10* selects the main menu. You can navigate in menu's with the *arrow* keys. Use *Enter* to select and *Escape* to go back up one level. *F5* and *F6* have been programmed as shortcuts for the selection of the previous and next tabsheets to quickly navigate through the problem specification. *ChemSep* also allows you to define the functionality of the other function keys, see the Tools menu.

1.1.2 File Menu

The *ChemSep* file menu has the usual entries such as New, Open, Save, Save as, and Close. These behave just like most programs define these functions, except for that 'new' and 'open' both open a new window to start the new problem. The 'close' option returns the current window to the state of a 'new' problem (and will ask you to save the problem when any changes would get lost in the process). This is consistent with the fact that *ChemSep* uses a different window for each problem it opens. To enable people to open a file in the current window, the filemenu of *ChemSep* contains the options 'Load' and 'Reload'. The latter reloads the currently opened file. This is particularly handy when you want to shed any changes that you made and want to go back to the problem as it was saved to disk.

Furthermore, the file menu contains the usual print, print setup, and exit menu items. Extra, however, is the option to export results to reports in specific formats (for example HTML). Also extra (in comparison to regular windows based programs) are options to view or edit a file on disk, and to open a window with a operating system (OS) shell for typing commands on the commandline.

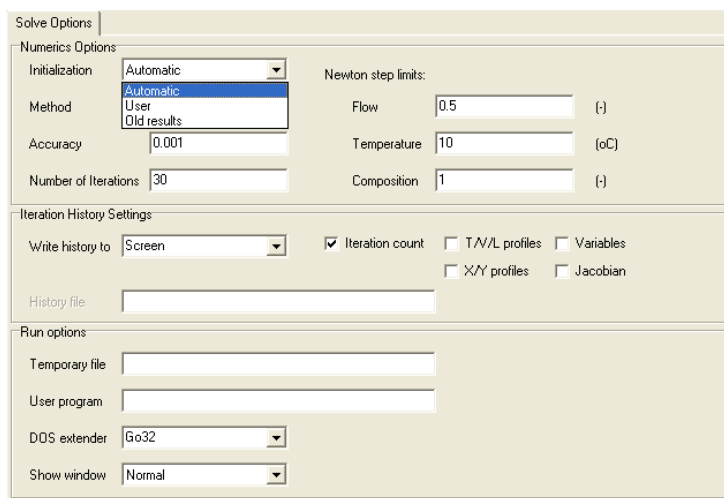
1.1.3 Edit Menu

The edit menu has the usual cut/copy/paste/delete options, but also contains options to switch to the previous and next input screen, just like the previous and next buttons below the input screen. When a solved problem has been loaded, the edit menu also contains an option to generate a new problem using one of the output streams in the simulation. This option allows the quick evaluation of a separation distillation train.

1.1.4 Solve Menu

The solve menu includes options to check the input on completeness and to solve the problem. *ChemSep* includes a quick solve that will bypass any checks for experts only. The method used

to solve and to initialize the problem are set in the solve options which you can set by clicking on 'solve option' in the tree list.



Numerics Options	
Initialization	Automatic
Method	Automatic User Old results
Accuracy	0.001
Number of Iterations	30
Newton step limits:	
Flow	0.5 (-)
Temperature	10 (oC)
Composition	1 (-)

Iteration History Settings	
Write history to	Screen
Iteration count	<input checked="" type="checkbox"/>
T/V/L profiles	<input type="checkbox"/>
Variables	<input type="checkbox"/>
X/Y profiles	<input type="checkbox"/>
Jacobian	<input type="checkbox"/>
History file	

Run options	
Temporary file	
User program	
DOS extender	Go32
Show window	Normal

Figure 1.2: Solve options input screen.

1.1.5 Analysis Menu

The analysis menu contains various options to investigate the behavior of the mixture as defined in the input. You can view phase diagrams, residue curve maps, and diffusion coefficients for binary and ternary systems. As soon as the problem has been solved the parametric study option is available which allows you to study the effect of each of the input variables. One or more parameters can be varied over a specified range. This is useful for optimisation studies. When a column was solved you can rate it to determine the diameter using a general tray capacity function or transfer column information to vendor software for detailed hardware design.

1.1.6 Tools Menu

The tools menu allows you to set the interface settings and save/load these. It allows you to add your own tools to *ChemSep*. For example, our units conversion tool is a separate program that is added to the user interface. In the interface settings you define the formatting of the input and results and the interactive behavior of the interface. Note that these global settings are saved together with the units of measure, path, and solve options of the problem you have loaded. These will then be used as defaults when you start a new problem!

1.1.7 Input

Wherever you need to enter any numeric input, *ChemSep* shows a '*' to indicate a value is not specified. Where possible, *ChemSep* will supply a default during the simulation so that you do not need to make the specification. However, some values need to be entered and this is indicated by the automatic checking of the input; the values that are required will be indicated in red. When you select the field and press *Enter* *ChemSep* will provide you with a default value, if it can. Or it might provide you with a selection list of models from which you have to choose.

ChemSep can process algebraic calculations wherever numerical input is required. This is useful since, if you don't know the actual numerical value that should be entered but you know how to calculate it, you may enter the calculation. Numerical formulae may include the four basic arithmetic operations, +, -, *, and /. Operations may be nested within parentheses () as well. When you have typed in the formula, press *Enter* to evaluate the result and *Enter* a second time to accept that result. *ChemSep* does not remember formula entries, only the final result so you may edit the formula until you press *Enter*. Here are some examples of numerical formula entry:

```
3           Enter
5-2        Enter
(2-1) * (5-2)  Enter
```

All of these result in the number 3. Formula entry can be useful in the feed spreadsheet where you are asked to enter the component flows. Perhaps you know the total flow rate and the mole fractions rather than the component flows. Instead of using your calculator to compute the component flows, you can let *ChemSep* do the calculations for you. By way of an example consider a column with a feed flow of 573 mol/s containing 36.5 mole percent ethanol, the rest being water. The component feed flows of ethanol and water could be typed in as:

```
573 * 0.365      Enter
573 * (1 - 0.365)  Enter
```

1.1.8 Units of Measure

ChemSep data entry fields also accept units. This feature is particularly useful if you know a quantity in some units other than the current set of units. You don't have to change the global units settings for this, simply type in the numerical value of the quantity and follow it with the units. The number will then be displayed in the units defined globally for the problem. For example, what if the default flow units are kmol/s but we know the feed flows in lbmol/h?

Simply type in the feed flow as, for example, *375 lbmol/h* followed by *Enter* and *ChemSep* will automatically convert the number to the correct value in the default set of units. You can use this feature in any data entry field in *ChemSep*. Spaces are ignored when evaluating the expression with units. Any combination of the units listed under Other Units may be used.

ChemSep checks the dimensions of the units you enter and displays a warning message if they do not have the correct dimensions. If the dimensions are correct a message with the conversion is displayed (this behavior can be switched off in the interface settings). A numerical formula and a unit string can be entered in the same field at the same time. All results of formula and unit entry are displayed in the default set of units. Note that all internal calculations are carried out in SI units and that unit entries are case sensitive when using their abbreviations.

Units that *ChemSep* recognises are shown in Table 1.1. *ChemSep* also recognizes the standard prefixes for multiples of 10. For example, *mmol/s* is recognized as *mol/1000 / s*. The complete list of prefixes and their abbreviations are listed in Table 1.2.

All units and their prefixes can be typed in data entry fields as well as selected to be part of the default unit set, see Figure 1.3.

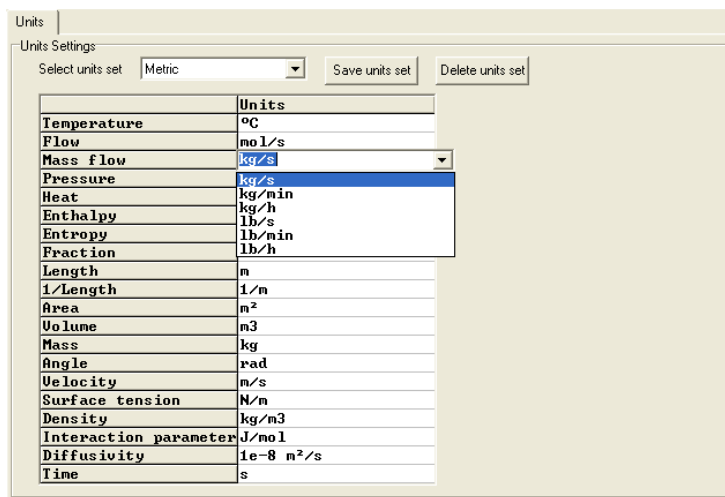


Figure 1.3: Units of measure specification for mass flows.

To change the units select the 'units' item from the list tree. Now click on any of the units and you see a pull-down list of the available units. This is just a sample list, you can enter any units by entering this from hand in the selection field. After selecting a unit or after pressing *Enter* while editing a unit definition a dimension analysis is performed to check its correctness.

Table 1.1: Units of measure

Abbreviation:	Unit:	Abbreviation:	Unit:
kg	kilogram	lbf	pound force
m	meter	kgf	kilogram force
s	seconds	atm	atmosphere
K	Kelvin	psia	lbf/square inch
kmol	kilomole	psig	lbf/sq in gauge
rad	radian	yd	yard
N	Newton	ton	ton
Pa	Pascal	USton	US ton
J	Joule	oz	ounce
W	Watt	lbmol	poundmole
°	degree	mol	mole
lb	pound	erg	erg
g	gram	dyn	dyne
min	minute	P	Poise
h	hour	mi	mile
day	day	°F	Fahrenheit
UKgal	UK gallon	°C	Celcius
USgal	US gallon	F	Fahrenheit
l	liter	C	Celcius
cal	calorie	R	Rankine
Btu	British Thermal Unit	bbl	barrel
in	inch	-	dimensionless
”	inch	%	percent
ft	feet	‰	per thousand
torr	torr	ppm	parts/million
bar	bar	ppb	parts/billion
barg	bar in gauge		

1.2 Distillation of Benzene and Toluene

Now let us start with a simple binary distillation column. As title enter "Distillation of Benzene and Toluene". You'll notice the title input window is now fully specified. So use the next button to move to the component entry. In this window you notice the default library (here chemsep1.pcd) is selected and that on the left list the components in this PCD library are listed by name. Now type the names of the components you want to select in the 'Find' inputbox. When you enter "benzene" you will notice there are many other components listed. This is because the interface uses a synonyms file to search and that all these components contain an

Table 1.2: Prefixes that can be used on any unit

Short:	Full:	Factor:	Short:	Full:	Factor:
T	Tera	10^{12}	d	deci	10^{-1}
G	Giga	10^9	c	centi	10^{-2}
M	Mega	10^6	m	milli	10^{-3}
k	kilo	10^3	μ	micro	10^{-6}
h	hecto	10^2	n	nano	10^{-9}
da	deka	10	p	pico	10^{-12}
			f	femto	10^{-15}

benzene ring. You can use the *down* and *up* arrow keys to select the actual component you want, in this case Benzene, and then select it by pressing *Enter*. This will also clear the find string so we can go ahead with selecting the next component, Toluene. Note that as soon as you defined the first component, the interface marked the component selection as complete (and assigned the green checkmark to the tab). Of course, for a flash problem one component selected suffices to do calculation, but for a distillation problem we need to select at least two components. Select Toluene as 2nd component.

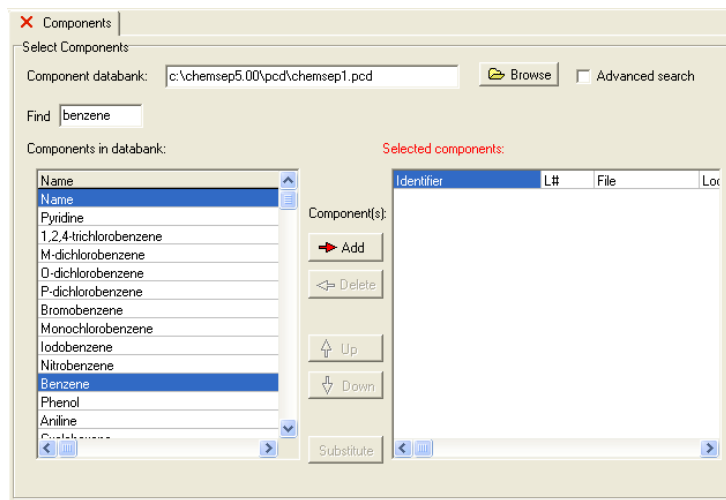


Figure 1.4: Selection of the component Benzene.

Alternatively you can select a component with the right arrow 'Add' button. Once the components are selected you can reorder them with the Up and Down arrow buttons. Delete components by using the left arrow 'Delete' button. You can select other component libraries by

clicking the browse button. *ChemSep* names its binary libraries *.PCD and its text libraries *.PCT. To modify or inspect the component libraries use the pure component data manager, in the main menu under 'Databanks'. Note that the components are identified by their names. We would like instead to use the letters "B" and "T" as identifiers. That can easily be done by editing the identifier fields for the selected components in the list on the right. We are then finished with the components and press *F6*.

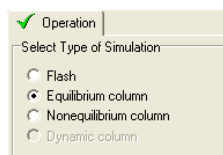


Figure 1.5: Selection of the type of simulation.

The next step is to select the type of simulation: in this case we are selecting an 'Equilibrium column'. As soon as we make this selection we need to specify the configuration of our distillation column. Here the interface supplies us with all kinds of defaults and we keep picking these: Simple distillation; Total condenser; Partial reboiler; 10 stages; feed on stage 5. Note that for a simple distillation column there are no sidestreams or pumparounds allowed. If you would want to enter those select 'Complex column' as kind of operation. As soon as we entered the feed stage we have fully specified the column configuration and a schematic representation of the column is shown on the right, see Figure 1.6.

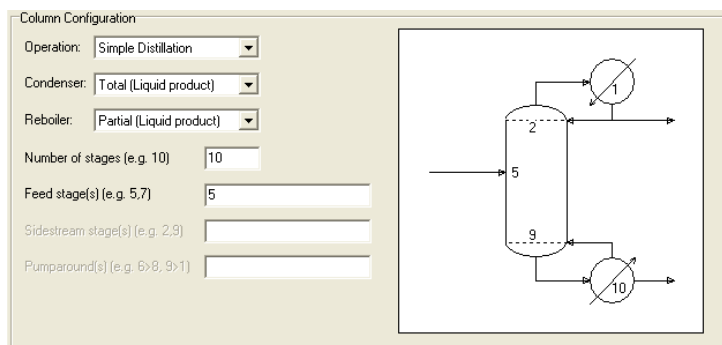


Figure 1.6: Column configuration for the BT column.

Having finished with the specification of the operation we continue by expanding the 'Properties' in the tree list on the left by clicking the little plus. We see that there are three sub entries: Thermodynamics, Physical properties and Reactions. The latter is already completed as no reactions are assumed per default.

We have to specify the thermodynamic models used to simulate the vapor-liquid equilibria on

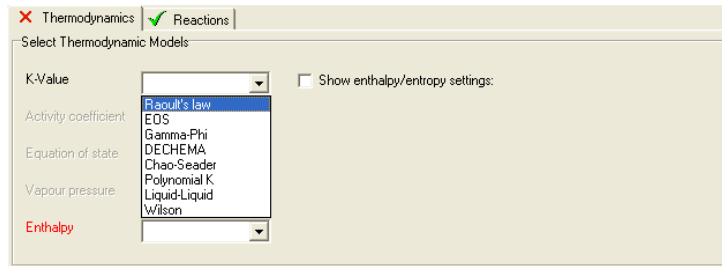


Figure 1.7: Thermodynamic models.

the stages in the column. Process simulation makes extensive use of thermodynamic properties, and they can have a profound influence on the simulation results. Selecting the right models sometimes requires insight and experience from the user. It is your responsibility to assess whether or not the thermodynamic model you have selected is adequate for your needs. The main specification here is the selection of the K-value model. Other model specifications depend on the type of K-value model selected. K-values describe the ratio of vapour (y_i) to liquid (x_i) mole fractions: $K_i = y_i/x_i$; for each component. The various models employ different assumptions and models for the calculation of the vapour and liquid fugacities from which the equilibrium phase compositions are derived via the equilibrium relation (which states that at equilibrium the component fugacities in both phases are equal, see Chapter 10).

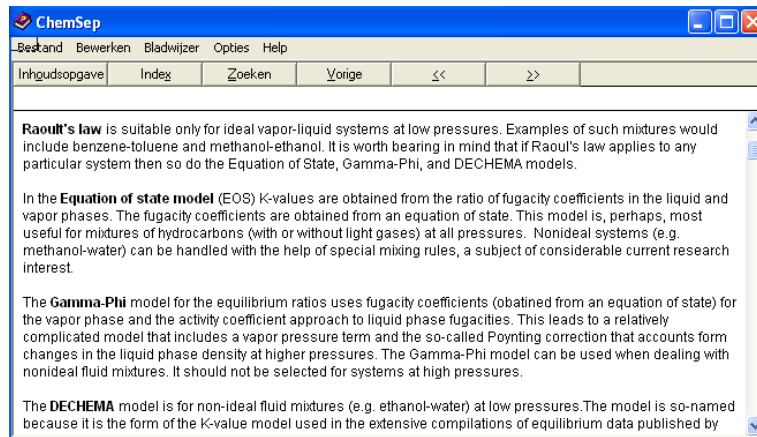


Figure 1.8: Context sensitive help.

Upon selecting a K-value model the items in the list for K-models are explained further when you press *FI* for the online help, see Figure 1.8. You can browse the help by selecting the links that are visible in the various help screens. Use *Esc* to get out of the online help and select

the DECHEMA K-model, UNIFAC for the activity coefficients, and Antoine for the vapor pressures. These are reasonably good model selections for the fairly ideal Benzene/Toluene system of our example. Use as enthalpy model 'none'. This means that the enthalpies for the liquid and vapor are assumed constant and equal for all the components. This assumption is often referred to as 'Equimolar Overflow' or 'Constant Molar Overflow' (CMO). It implies that the vapor and liquid flows up and down the column are constant from stage to stage (except around stages that involve feed or product streams).

After selecting the thermodynamic models we need to load the data needed by these models. As Antoine parameters are already part of the pure component data loaded, and the UNIFAC model is a group contribution method for which we know the component's groups, no further input is required. This is shown in the area where the thermodynamic model parameters are to be entered with a message that no further input is required. The tab sheet also has a green checkmark indicating it is finished. However, other models most likely will require specification of model parameters.

Using *F6* we switch to the next input screen. This brings us to the physical properties where we can define models for densities, viscosities, thermal conductivities, surface tension, and diffusivities. *ChemSep* provides a *default* set of model selections so no user input is required. If you want your own selection, uncheck the checkbox "Use default methods" and all the selections will become visible. This is not needed now and we use *F6* again for the next input screen, which is reactions. As benzene and toluene do not react with each other, we can skip this item and move to the next item in the input: Feeds.

Feed Stream(s) Specifications	
Feed	1
Name	
Stage	5
State	Temperature & pressure
Pressure (bar)	Temperature & pressure
Vapour fraction (-)	Pressure & vapour fraction
Temperature (oC)	*
Flowrates (mol/s)	*
B	*
T	*
Total flowrate	*

Figure 1.9: Feed state specification.

The next item in the menu is the input of the column feed data. The feed stage has already been specified in the operation spreadsheet,

Note that *ChemSep* already specified a name for the feed we entered during the specification of the column operation. It numbers the feeds in sequence of the feed stage numbers we supplied but it can be changed here if necessary. Two different types of specifications can be given to fix the feed state: the temperature and pressure or the vapor fraction and the pressure (of the feed).

This specification is necessary to determine how much of the feed is vapor or liquid. This helps to determine the internal flow rates.

In this example we want to specify a 'saturated feed' and we do this by selecting the vapor fraction specification. When a vapor fraction of zero is specified, the feed is taken to be at its boiling point (in the simulation a flash calculation is performed to calculate the boiling point of the feed mixture). Of course, we must specify a pressure to determine the feed temperature. Here our column operates just above atmospheric pressure in order to prevent ingress of any air in case any leak might develop.

We enter "1.1 atm" and we are asked to confirm the conversion to the default units for the pressure, namely in bar. If you enter the wrong type of units *ChemSep* will warn you and let you try again. Next we need to set the feed composition and total flow rate. Here we want an equimolar feed with a total *mass* flow rate of 1 kg/s. To do this we first enter equimolar feed rates of Benzene (B) and Toluene (T). Any amount will do as long as we enter the same rates for the two components. Then we enter "1 kg/s" in the total flowrate field. As the interface knows the molecular weights of the components it can do the conversion of the mass to the molar flowrate. The 1 kg/s corresponds to 11.7471 mol/s as total feed flow.

Property	Value
Feed:	1
Name	
Stage	5
State	p & V
Pressure (bar)	1.11458
Vapour fraction (-)	0
Temperature (oC)	
Flowrates (mol/s)	
B	5.87357
T	5.87357
Total flowrate	11.7471

Figure 1.10: The feed's total *mass* flow rate specification of 1 kg/s is automatically converted to molar flows.

As our simple distillation column only has one feed, we are done and can move to the next input screen: expand the tree for the 'specifications'. There are five tab screens listed for this input: Analysis, Pressures, Heaters/Coolers, Efficiencies, and Column specifications. The analysis screen shows us the degrees of freedom for the equilibrium staged column and therefore what all needs to be specified.

We see that we need to specify the pressures on each stage and we do this by selecting the Pressures tab after we opened the Specifications. Since we configured the column to have a condenser *ChemSep* asks us to specify pressure in the condenser first.

There you will notice that each item that needs to be specified is shown as an asterisk (*) in

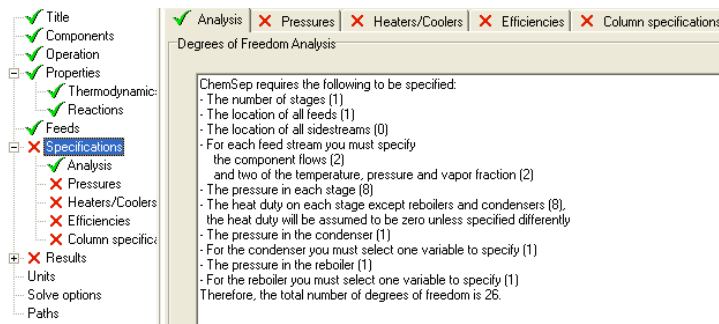


Figure 1.11: Degrees of freedom analysis for the BT column.

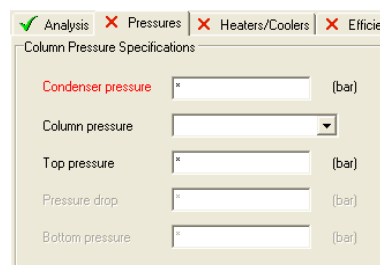


Figure 1.12: The pressures tab sheet.

ChemSep. Upon selecting such a field, the asterisk usually is replaced by a default value or else a value that is consistent with the other specifications that already have been made. In this example we already have entered a feed pressure and thus *ChemSep* assumes that the condenser pressure equals this pressure. Normally, the condenser is not located on top of the column but at some lower elevation. There is, therefore, a considerable length of pipe between the column and the condenser over which a certain pressure drop exists. Thus, it is likely that the condenser has a lower pressure than the top of the column but to keep this problem simple we assume that the pressure is the same throughout the column as well as in the condenser. Thus, we accept the defaults *ChemSep* provides for the condenser and column top pressures and select the constant pressure specification.

The degrees of freedom also tells us that on each stage we need to specify a heat duty, representing either a heater when positive or a cooler when negative. These duties are optional, and you therefore would need to "insert" such a cooler or heater. When doing so the interface asks you to specify the stage number, name of the heat exchanger, and its duty. As columns typically operate at temperatures above ambient, there is typically also a heat loss over the column. This is modeled as a separate duty that gets subtracted from each stage, except for the condenser and reboiler. The default name for this the heat loss is set to 'Qcolumn' and you only need to

Figure 1.13: Column heat loss and stage coolers/heaters.

specify a value for it. In this example we will assume the column is insulated well enough that the heat loss is negligible and therefore we assume the default value of zero.

Figure 1.14: Efficiencies.

After the specifications of the heaters and coolers the next sheet brings us to the stage Efficiencies. A default stage efficiency for all the stages is to be specified. This value is used for all stages in the column except the condenser and reboiler, which are assumed to operate at equilibrium. We can specify stage specification for specific stages by inserting and deleting stage efficiencies. These are Murphree vapour efficiencies (see /chaprefSEXrealstages). Here we use the default efficiency value of one.

Figure 1.15: Condenser specifications.

The next and also the last specification tabsheet is that for the column product specifications. As we have equipped our column with both a condenser and a reboiler, we have to make two

additional specifications that are going to fix the heat duties of these heat exchangers and that will determine the internal flow rates as well as the overall mass balance by means of the distillate and bottoms flows. Again, we see that *ChemSep* has provided default names for the product flow names (Top, respectively Bottom) and heat duties of the condenser and reboiler (indicated with a Q). For the top specification there is a whole list of specifications which we are allowed to make. The first is the reflux ratio, RR, a very common specification that determines the ratio between the distillate top product and the flow that is refluxed back to the column. *ChemSep* provides a default value for the reflux ratio of two, however, this value will depend very much on the application and system at hand. This means that the ratio of refluxed liquid flow rate (R) to the product distillate flow (D) is two (twice as much liquid is returned to the top of the column as reflux compared to the liquid drawn as distillate product).

For the reboiler select the total bottoms flow rate, B. As we fed an equimolar mixture to the column, we want to draw equal bottoms as distillate flows, so we specify a bottoms flow of 5.874 mol/s . The operation specifications spreadsheet shows both these specifications. The combination of reflux ratio and bottoms flow rate is one that gives the most rapid convergence is our experience. You can make other combinations but not all will converge very quickly, and some combinations might even give rise to impossible specifications. That's why we advocate the RR-B combination for each problem you start with. After you have solved the problem with this specification you can see what the column does and when necessary switch to a different set of specifications.

Figure 1.16: Column specification for the Benzene-Toluene example.

After the column specifications have been set all the red crosses in the input tree list have become green checkmarks, meaning we have successfully completed the input specification for our binary distillation example and can go ahead with solving the column. We can do so by pressing *Alt-S* or clicking the little calculator on the toolbar. This will also check our input and if we forgot something bring us to the window where more input is required or where there is a problem with the specifications. To solve the problem we must also save it into a simulation file in order to pass the problem on to the simulator programs. They are separate programs that read the problem from the simulation file and append the results to the same file. Thus, we are

prompted for a valid filename and path. The default extension (.sep) is automatically appended by *ChemSep*. After the simulation filename is specified, say "bt1", the input is saved and the column simulator called.

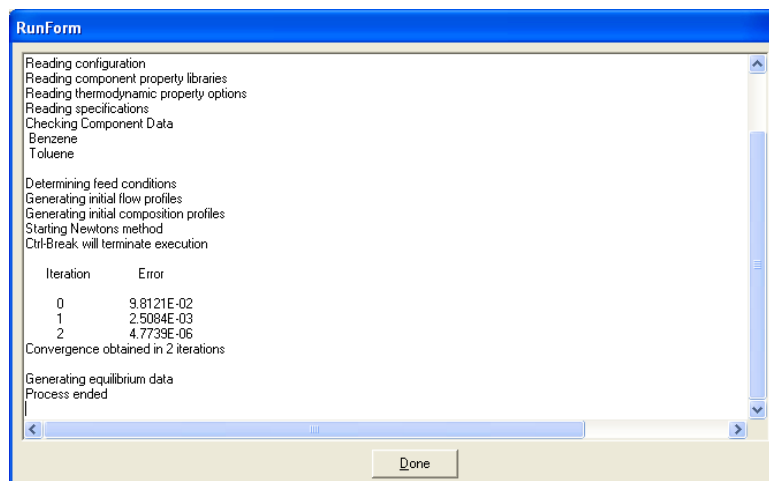


Figure 1.17: Solving the benzene-toluene column.

During the solve process a window shows the progress of the calculations in the form of messages of reading the problem and while solving a sum of errors for each iteration. This error sum is a norm for the convergence obtained and needs to become less than the accuracy set under the solve options (for a column typically 10^{-3}). When the norm is small enough the process ends. Click the window away with the "Done" button.

1.3 Results

1.3.1 Tables

After the simulation has been completed successfully *ChemSep* will reread the sep-file and enable the results tabsheets: Tables, Graphs, McCabe-Thiele, and FUG, the latter is an acronym for Fenske-Underwood-Gilliand. The interface will show the tables tab with the mass and energy balances for the column (Fig. 1.18) as default table. Here we can see in whether the overall and component mass balance have been met. The overall balances should be equal to zero or close to 0 (within the criterion used to converge the simulation). Also, the component balances must be (approximately) zero. When this is not the case, the solution found is not a physically meaningful solution. It is important to *always* check this.

To see a different table, select it from pull down list of the table selector. For example the T/P/Flow profiles. When selecting this table we see that the reflux ratio we specified was indeed also obtained and that together with the bottoms flow rate specification this implied a boilup ratio of three (within the tolerance we supplied at the solve options). We see that the pressures are all the same throughout the whole column – as we specified – and the temperatures vary from 85°C in the condenser to 111°C in the reboiler. As we add heat in the reboiler and remove heat in the condenser, this is the temperature profile one would expect for a distillation column.

Mass and Energy Balances	
Stream / Apparatus	Mass (mol/s)
Feed1	11.7471
Top	-5.8731
Bottom	-5.874
Balance	4.6566E-07 +
Component discrepancies (mol/s)	
B	1.5461E-06
T	9.8144E-08

Figure 1.18: Mass and energy balances.

Stage	Temperatur (oC)	Pressure (bar)	Flow rates (mol/s)		Feed	Product
			Liquid	Vapour		
1	84.9813	1.11458	11.7462	RR=2		5.8731 L
2	87.211	1.11458	11.7462	17.6193		
3	90.1329	1.11458	11.7462	17.6193		
4	93.2968	1.11458	11.7462	17.6193		
5	96.1318	1.11458	23.4933	17.6193	11.7471	
6	98.5414	1.11458	23.4933	17.6193		
7	101.669	1.11458	23.4933	17.6193		
8	105.092	1.11458	23.4933	17.6193		
9	108.23	1.11458	23.4933	17.6193		
10	110.688	1.11458	BR=2.99954	17.6193		5.874 L

Figure 1.19: Profiles for the temperature, pressure, and column flows.

Note that we can use one of the buttons above the table to export the table into an Microsoft Excel table or an ASCII table we can edit in a text editor. Alternatively, we can copy the table to the clipboard. The nice thing is that the table can also be edited from hand in the interface before we do that. Of course, as soon as we select a new table with the table selector we loose any changes we make.

The next table (Fig. 1.20) shows the compositions and K values for benzene. Here we see that the benzene concentrates in the top of the column and that its K-values are larger than one. As we recall, the K value was defined as the ratio of the vapor over the liquid mole fraction, this means that the relative amounts of benzene are higher in the vapor than in the liquid. This is why the fraction of benzene increases as the vapor rises up the column, condensing at 92

mole% whereas the feed was only 50%.

Profile for B			
Stage	Liquid (-)	Vapour (-)	K (-)
1	0.925473	0.97095	1.07974
2	0.826416	0.925473	1.1506
3	0.706363	0.859435	1.24184
4	0.586978	0.779399	1.34184
5	0.488109	0.69981	1.43502
6	0.409455	0.625969	1.53208
7	0.314039	0.521093	1.67021
8	0.217372	0.393866	1.82865
9	0.135155	0.264972	1.97395
10	0.074592	0.155345	2.08334

Figure 1.20: Profile for the compositions and K values for benzene.

Stream	Feed1	Top	Bottom
Stage	5	1	10
Pressure (bar)	1.11458	1.11458	1.11458
Vapour fraction (-)	0	0	0
Temperature (oC)	95.78	84.981	110.688
Mole flows (mol/s)			
B	5.87355	5.4354	0.438153
T	5.87355	0.437703	5.43585
Total molar flow	11.7471	5.8731	5.874
Mole fractions (-)			
B	0.5	0.925473	0.074592
T	0.5	0.0745268	0.925408
Mass flows (kg/s)			
B	0.458804	0.424578	0.0342257
T	0.541192	0.0403302	0.500862
Total mass flow	0.999996	0.464909	0.535087

Figure 1.21: Stream table.

The stream table shows a complete overview of the feed(s) and products of the distillation column. It shows pressure, vapor fraction, temperature, enthalpy, molar and mass flows for the (component) flows of each stream. In this table we see that the feed is equimolar; both component mole fractions are 0.5. The distillate top product and the bottoms product compositions are almost exactly mirrors of each other in respect to the mole fractions. As there is a small difference in molecular weights between the components, the weight fractions are somewhat different. Note that Figure 1.21 we have edited the stream table. Just place the mouse in the table and start editing.

Two more tables are available: a summary of the specifications and a table showing the output from the column simulator which was written to the run window. The specification summary is a very useful table for it contains the complete problem specification of the simulation, with this table anyone can reproduce the simulation. It also includes the little schematic drawing of the column flowsheet to show the column configuration.

```

ChemSep (TM) - Flash/Col2 v5.00s
Copyright(c) H.Kooijman and R.Taylor, 1988-2005
Checking data file C:\tmp\bt.sep
Reading configuration
Reading component property libraries
Reading thermodynamic property options
Reading specifications
Checking Component Data
  Benzene
  Toluene
Determining feed conditions
Generating initial flow profiles
Generating initial composition profiles
Starting Newtons method
Ctrl-Break will terminate execution

      Iteration      Error
          0          7.2639E-04
Convergence obtained in 0 iterations
Generating equilibrium data
c:/chemsep5.00/bin/go-col2.exe done

```

Figure 1.22: *ChemSep* simulator output.

1.3.2 Graphs

ChemSep has a rich set of plotting options for the column simulation results. It is much easier to analyze the results in graphical than in table format (as is true for most other problems). The graphs tab sheet has a selector that shows the different standard graphs that are available. When you select a graph it will be displayed using the default graph program, GNUplot. To copy this graph to the clipboard or to print it, just right-click the GNUplot graph window and you get a selection of options. Use *Alt-F4* to click the graph window away. For a detailed discussion of GNUplot see the chapter on GNUplot.

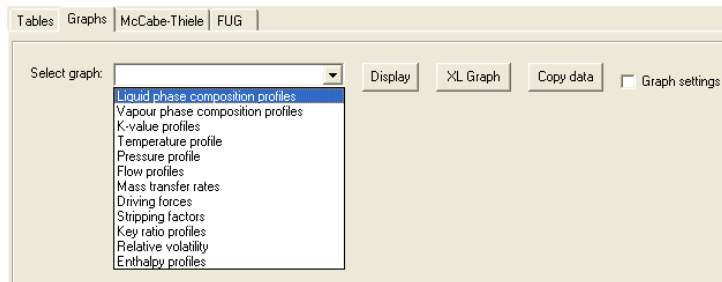


Figure 1.23: Graph menu.

The first two kinds of graphs listed in the graph picklist are the liquid and vapor mole fraction profiles that show how the components are distributed over the column. The default orientation in *ChemSep* is to plot the stages vertically, with the condenser at the top and the reboiler on the bottom. One can imagine the profile projected over the column standing straight up on the left axis.

If we want to change any plot setting, simply select the graph settings checkbox on the right hand side. This will show the current settings of the graph, the horizontal and vertical axes, and the data sets plotted in the selected graph. To plot the graph after making changes in any of these settings, use the 'Display' button. With the settings visible we can change the colors, point types, axes, etc. of the plot as well as add new profiles to be plotted (possibly using the same or a second axis).

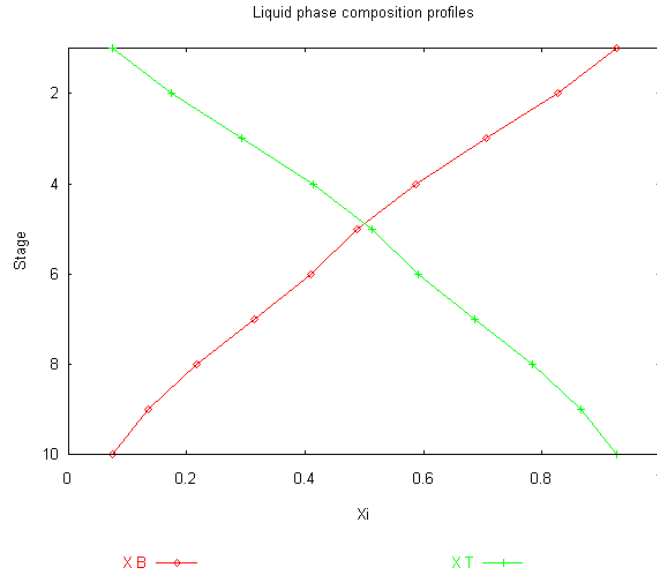


Figure 1.24: Liquid compositions.

Looking at the compositions in Figure 1.24 we observe that benzene concentrates in the top of the column and toluene in the bottom. This is to be expected as benzene boils at 80 and toluene at 110.6°C at atmospheric pressure. Note that the composition profiles show an almost symmetric behavior. This is an indication that benzene and toluene form a relatively ideal mixture.

The components in the mixture divide over the liquid and vapor phase as is dictated by the K-value, which is defined as the ratio of the mole fractions in the two phases present. As we selected to use the DECHEMA K-model the K-values are given by

$$K_i \equiv \frac{y_i}{x_i} = \frac{\gamma_i P_i^*}{p} \quad (1.1)$$

As the pressure is constant over the whole column and benzene and toluene form an relatively ideal mixture, the activity coefficients are close to unity. Therefore, the K values are dictated mostly by the behavior of the component vapor pressures, which increase with increasing temperatures. Figure 1.25 shows that the K-values and temperatures increase from top to bottom.

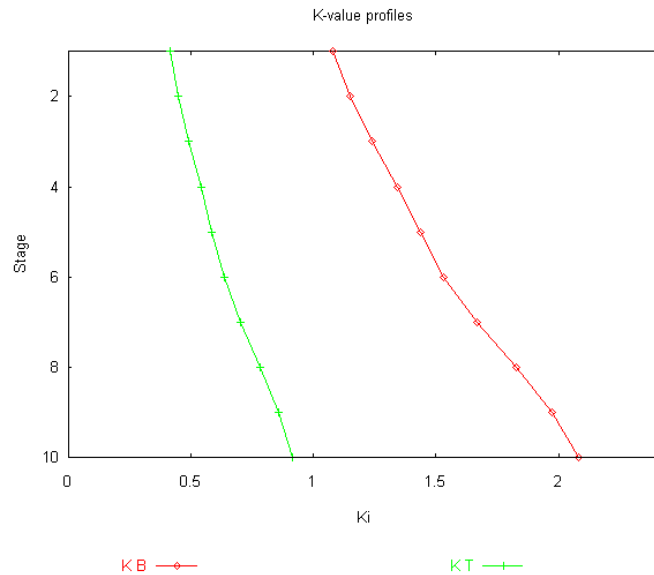


Figure 1.25: K values plot.

As the activity coefficients have very little influence here it must be caused by an increasing temperature. In Figure 1.26 we indeed see that this is the case. The temperature is increasing from top to bottom because in the top we have more benzene boiling at 80 degrees and in the bottom we have more toluene boiling at 110 degrees.

The ratio of the K-values is the relative volatility:

$$\alpha_{ij} \equiv \frac{K_i}{K_j} \quad (1.2)$$

where we usually select for component i the lighter component (which evaporates quicker, ergo, it has a larger vapor pressure and thus a larger K-value). This ratio indicates how easy it is to separate two components from each other. In our example – using the DECHEMA model – this ratio becomes:

$$\alpha_{ij} = \frac{\gamma_i P_i^*}{\gamma_j P_j^*} \quad (1.3)$$

When the activity coefficients are both close to unity (true for ideal mixtures) the relative volatility is just the ratio of the component vapor pressures. As both vapor pressures increase with temperature, the relative volatility is often only a weak function of temperature. This is especially so when the boiling points of the two components are close. In our case the relative volatility varies from 2.25 in the reboiler to 2.6 in the condenser (Figure 1.27).

However, the relative volatility is not the only thing that influences the separation, so can the

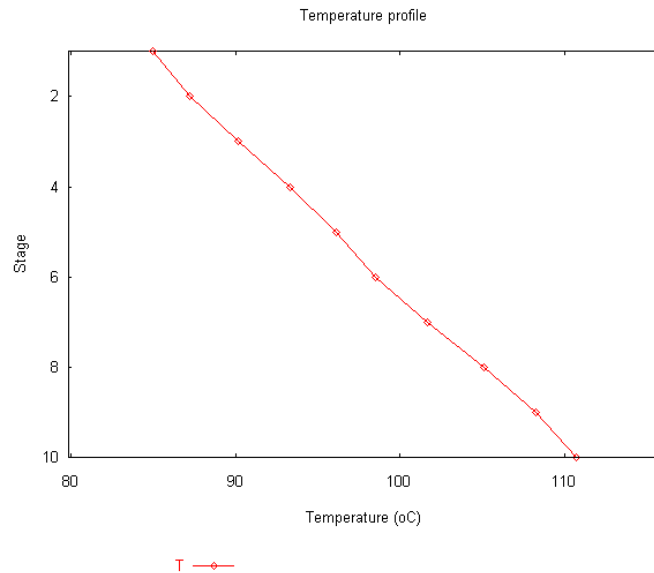


Figure 1.26: Temperature profile.

internal vapor and liquid column flows, as they are responsible for moving components through the column. Figure 1.28 shows that in this example the vapor flow rates are constant in the whole column whereas the liquid flows are discontinuous at stage 5.

This can be explained by the fact that the feed enters as a saturated liquid on stage 5. At this stage we therefore see an increase in the liquid internal flowrate. As the feed is at its boiling point, no vapor is condensed, nor is any vapor added at the feed stage. The only vapor generation is done in the reboiler. The difference in the liquid from stage 1 (the condenser) and the vapor from stage 2 (which enters the condenser) equals the distillate flowrate. We see that this difference is half the size of the reflux flow from the condenser (the liquid flow stage 1); thus, the reflux ratio is equal to 2.

The stripping factor is defined as the ratio of the component vapor flow going to the stage above and the component liquid flow going to the stage below:

$$S_i \equiv \frac{y_i V}{x_i L} = \frac{K_i V}{L} \quad (1.4)$$

Therefore, it is the stripping factor actually indicating the direction a component is moving (up or down) in the column. If it is larger than one, this means the component flow upwards in the column is larger than the component flow downwards at this stage. As we can see in Figure 1.29, the stripping factor for benzene is larger than unity at all points in the column, hence, the composition of benzene is decreasing monotonously from the top to the bottom of

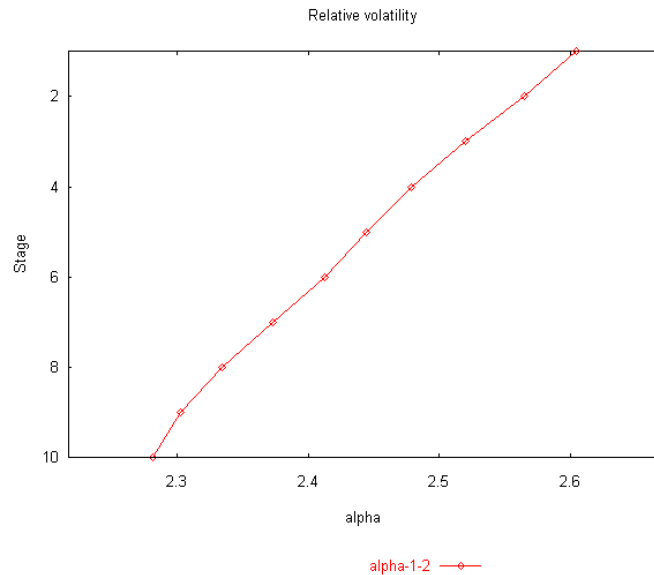


Figure 1.27: Relative volatility.

the column.

Figure 1.30 shows that the mass transfer rates for the two components are symmetrically opposite. In this plot the mass transfer rates are defined to be positive for condensing components. That is why the mass transfer rates for benzene are negative as it evaporates at each stage in the column (note the condenser and reboiler are not shown!). The fact that the profiles for benzene and toluene are symmetric is obvious when we realize that we are separating a binary mixture under equimolar overflow conditions: if one component evaporates the same amount of the other component needs to condense.

In another useful diagram we plot the ratio of the liquid mole fractions:

$$R_{ij} \equiv \frac{x_i}{x_j} \quad (1.5)$$

see Figure 1.31. This plot shows the actual separation achieved between components i and j . This ratio is plotted on a logarithmic scale as when each stage removes a certain fraction of a component, the composition of a component will decrease exponentially to zero.

This plot can be used to check whether a feed is properly placed to the column. If the ratio shows a abrupt change in direction, this is an indication that the separation could be improved by moving the feed to a different stage. In our case the curve is changing its direction in a continuous manner. A better way to determine the feed placement is the McCabe-Thiele

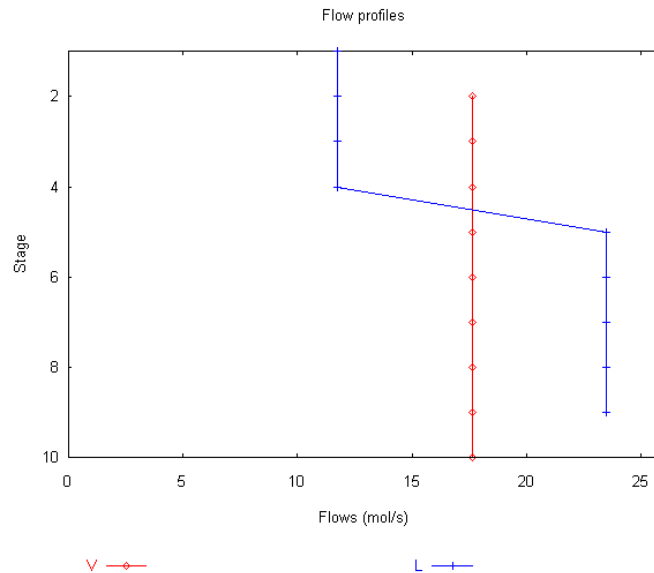


Figure 1.28: Internal flows.

analysis.

1.3.3 The McCabe-Thiele Diagram

The McCabe-Thiele diagram was used by McCabe and Thiele (1923) to provide a graphical solution to the distillation equations. It is based on a binary x - y diagram, see Figure 1.33. For ideal binary mixtures where the relative volatility is constant the vapor mole fraction can be calculated from:

$$\alpha_{12} = \frac{K_1}{K_2} = \frac{y_1/x_1}{y_2/x_2} = \frac{y_1(1-x_1)}{x_1(1-y_1)} \quad (1.6)$$

solving for y_1 this gives:

$$y_1 = \frac{\alpha_{12}x_1}{1 + x_1(\alpha_{12} - 1)} \quad (1.7)$$

This is the curved line in the diagram. Of course, *ChemSep* uses the actual K-model to calculate the equilibrium curve and in this case the relative volatility is not constant, as we saw in Figure 1.27. In the x - y diagram we add the $y = x$ line as an auxiliary line. The larger the relative volatility, the further away the equilibrium line is from the $y = x$ line and the fewer stages are required to separate the two components.

At $x = 0.5$ another small line is drawn that represents the feed or 'q' line. A number indicates

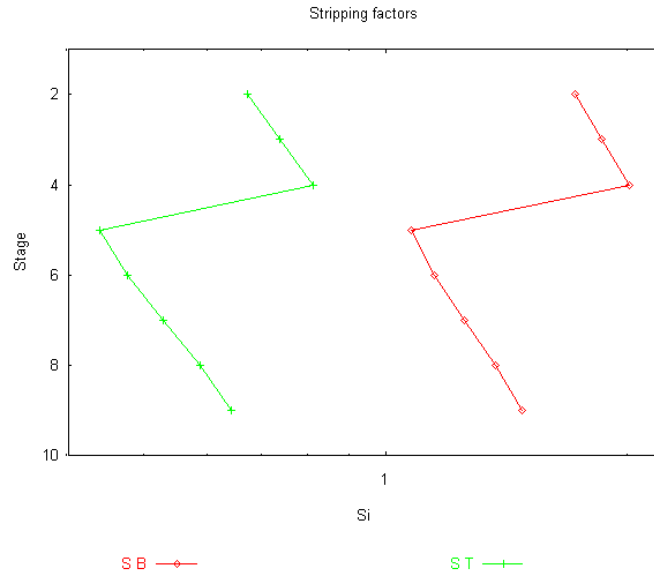


Figure 1.29: Stripping factors.

the feed stage (5). Here 'q' stand for the 'quality' or liquid fraction in the feed, L/F . Using the component balance for the feed divided by the feed flow F we find for z (the feed overall mole fraction):

$$z = x \frac{L}{F} + y \frac{V}{F} \quad (1.8)$$

substituting the total molar balance $V = F - L$ and $q = L/F$ gives

$$z = qx + (1 - q)y \quad (1.9)$$

which can be rewritten to determine y as

$$y = \frac{1}{(1 - q)}z - \frac{q}{(1 - q)}x \quad (1.10)$$

when $q = 1$ Eq. (1.9) represents a vertical line at $x = z$, as is shown in our example. From Eq. (1.10) we also see that the q -line crosses the $y = x$ line at $x = z$.

The other solid lines in the diagram are the operating lines. These lines define the molar balances for the two column sections in our column. If we look at the top (rectifying) part of the column, we have an internal vapor column flow V entering a stage, and an internal liquid flow L leaving the stage. For the condenser (stage 1) this is:

$$V_2 = L_1 + D \quad (1.11)$$

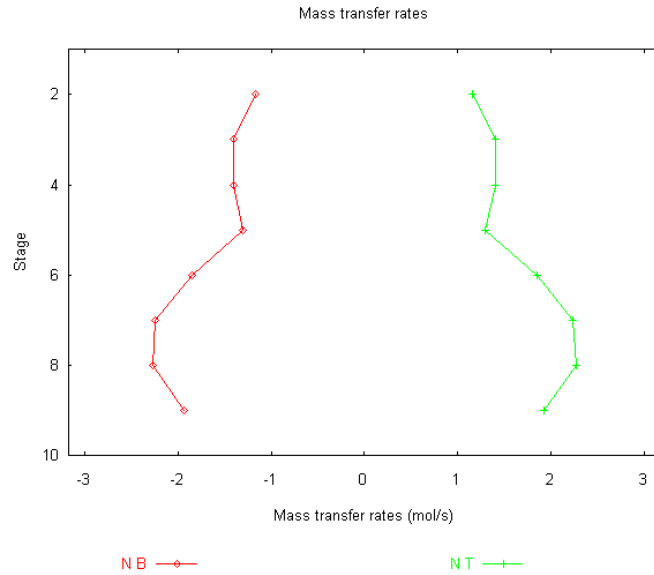


Figure 1.30: Mass transfer rates.

The reflux ratio is defined as $R = L_1/D$. We can write for the stage below the condenser:

$$V_3 = L_2 + D \quad (1.12)$$

When the the vapor and liquid enthalpies are constant, the amounts of liquid and vapor do not change (if there are no feeds!); thus, we can write:

$$V = L + D = (R + 1)D \quad (1.13)$$

The same is true for the component molar balances:

$$Vy = Lx + Dx_D \quad (1.14)$$

This is a straight line in our diagram with:

$$y = \frac{L}{V}x + \frac{D}{V}x_D = \frac{R}{R+1}x + \frac{x_D}{R+1} \quad (1.15)$$

This is a straight line with slope L/V (or $R/(R+1)$) which crosses the $y = x$ line at x_D .

Now the same derivation can be done for the lower part of the column where bottoms flow B with composition x_B and the L' and V' internal flows give:

$$y = \frac{L'}{V'}x - \frac{B}{V'}x_B \quad (1.16)$$

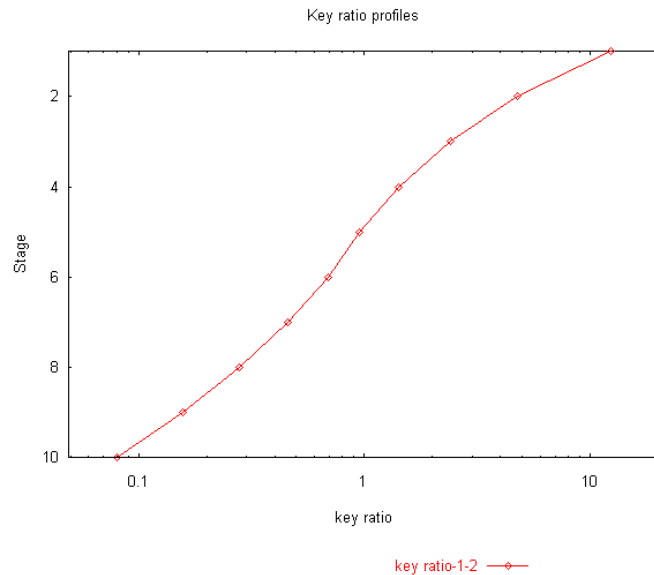


Figure 1.31: Key ratios.

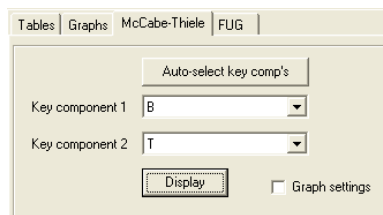


Figure 1.32: McCabe-Thiele tabsheet.

This is also a straight line with slope L'/V' which crosses the $y = x$ line at x_B . Now it becomes clear why we used the $y = x$ line as our auxiliary line: the two operating lines and the q line cross this line at the feed and top/bottom compositions, respectively.

The dotted lines drawn in between the equilibrium line and the operating lines represent the stages. We start drawing at the reboiler composition x_B on the $y = x$ line. The reboiler generates a vapor with a vapor composition of y_B , which is the y value on the equilibrium line at $x = x_B$. In other words, we draw a dotted line straight up to the equilibrium line. The liquid composition from the stage above can be obtained by moving horizontally to the operating line, as the operating line determined the mass balance for the internal flows. Then, for the stage

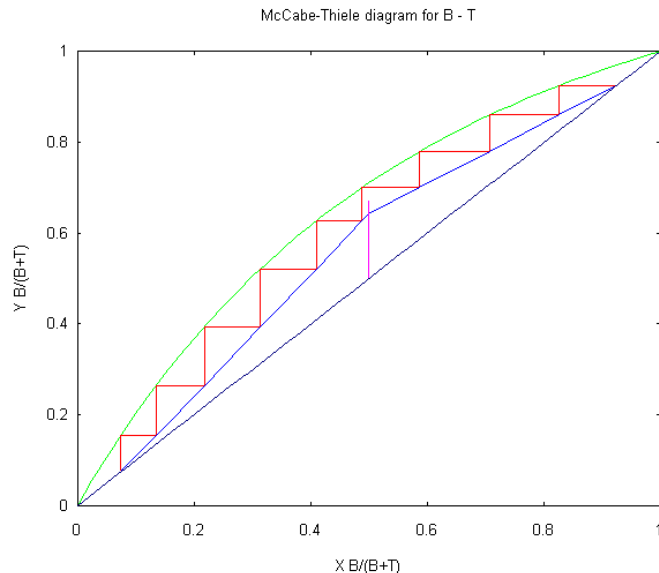


Figure 1.33: McCabe-Thiele diagram.

above we draw again a vertical line to the equilibrium line and repeat this process. When we pass the feed stage we switch from using the stripping operating line to using the rectification operating line. A total condenser does not generate another triangle as all the vapor is condensed into a liquid and no further separation is achieved.

If instead of a saturated liquid feed we want to specify a saturated vapor feed we need to go back to the feeds specification by clicking on the 'Feeds' item in the tree on the left. Specify a vapor fraction of 1 and use *Alt-S* to initiate the checks and solving. You are now asked whether to overwrite the simulation file. Don't, instead save the new problem under a different name, say *bt2.sep*.

The resulting McCabe-Thiele diagram looks quite different. Observe that the distillate and bottoms compositions are closer to the feed composition. To view what is going on around the feed line we will make an enlargement of this area: select the graph settings checkbox in the McCabe-Thiele tabsheet and enter $x_{min} = 0.2$, $x_{max} = 0.6$, $y_{min} = 0.3$, and $y_{max} = 0.7$. When we then select the 'Display' button we see that the feed is now a vapor feed as the 'q-line' is a horizontal line. The slope of the q-line is given by $q/(1 - q)$ which is zero if $q = 0$. This results in a misplaced feed and we have to keep stepping off stages on the stripping operating line when it would have been more advantageous to step off stages on the rectifying operating line. The two stages in the column are much less effective as they could be and hence the column products have lower purities.

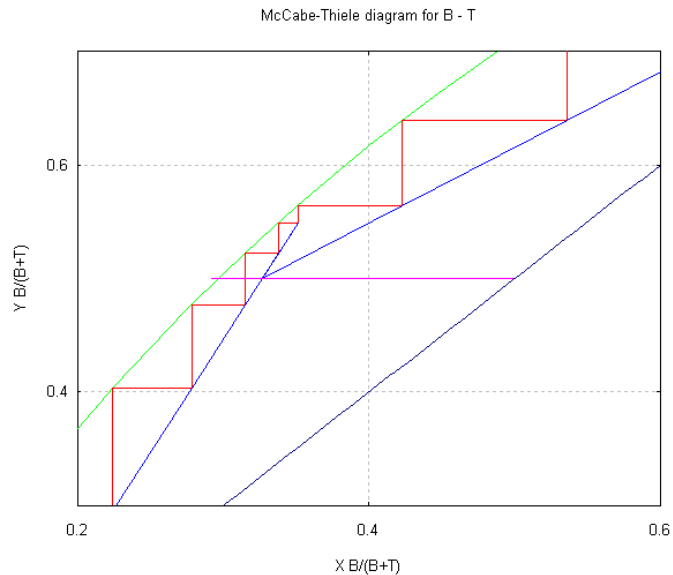


Figure 1.34: McCabe-Thiele diagram for column with a vapor feed.

If the vapor fraction is specified as 0.5, we can see that the feed line has a slope of -1 and that the feed is again much better placed (but still not optimally). In this case stage 6 would be the optimal feed stage.

The feed quality can also be defined as:

$$q \equiv \frac{L' - L}{F} \quad (1.17)$$

which has the advantage that the feed quality can also be larger than unity or smaller than zero. This occurs when subcooled liquid or superheated vapor feeds are introduced in the column. A subcooled liquid needs to be heated up to its boiling point, in which it will condense a certain amount of vapor generating a higher internal liquid flow L' . The opposite is true for a superheated vapor. For a subcooled liquid the slope of the feed line is even positive, the same for a superheated vapor.

Subcooled liquids are more common than superheated vapors for distillation columns for the simple fact that distillation feeds are often liquids (from storage) and a pre-heater represents an extra piece of equipment. The same task can be taken over by the reboiler, albeit at a higher temperature than the feed, requiring energy at a higher temperature (as well as shifting the operating line in the stripping section of the column).

When we change the reflux ratio of the column from 2 to 1 (Fig. 1.36), we see a change in the

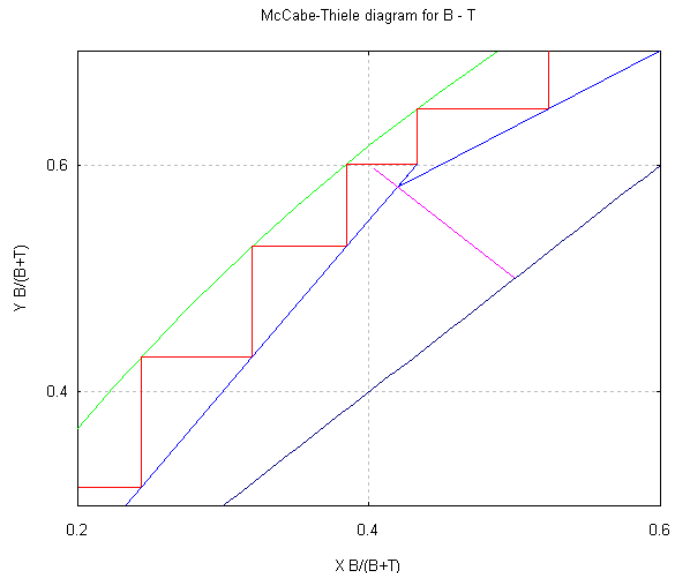


Figure 1.35: McCabe-Thiele diagram for a feed with equal amounts of vapor and liquid.

McCabe-Thiele diagram, namely that the operating lines have different slopes. As was shown before, the slopes of the operating lines are given by the ratios of the internal flows. For the rectification line we had a slope equal to $R/(R + 1)$. For $R = 2$ this gives a slope of $2/3$ and for $R = 1$ only a slope of $1/2$.

We also see that the operating lines intersect each other almost on the equilibrium line. When they intersect the equilibrium line we would need an *infinite* number of stages to switch from the stripping to the rectification operating lines. Therefore, this is called a ‘pinch-point’, as the stages get pinched between the equilibrium and the operating lines. The reflux ratio at which the operating lines intersect the point at which the feed line crosses the equilibrium line is called the *minimum reflux ratio*. At that point the column has an infinite number of stages.

On the other hand, if we increase the reflux ratio to 10 (figrefFEXrr10) we observe that the operating lines lie closer to the $y = x$ line and that the stepping of stages leads to larger changes in x resulting in a better separation. If we further increase the reflux the operating lines move closer to the $y = x$ line until, at *total reflux*, they coincide with the $y = x$ line. Of course, at total reflux ($R = \infty$), all the vapor overhead is condensed and returned to the column so no product can be drawn (and hence, there is no production). At this point the separation uses the lowest number of stages to achieve a specific separation.

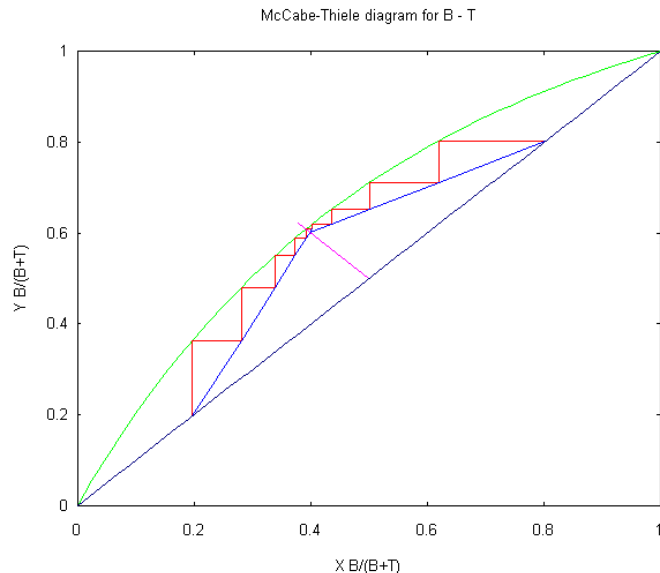


Figure 1.36: McCabe-Thiele diagram with a reflux ratio of 1.

We see that there are two extremes for the separation for a given set of distillate and bottom compositions: the minimum reflux ratio at the pinch point and the minimum number of stages at total reflux. Real operation must lie somewhere between these two extremes. Typically, distillation columns operate at reflux ratios that are 1.1 and 1.5 times the minimum reflux ratio. Reset the reflux ratio to 2.

1.4 Real stages

Previously we assumed constant equimolar overflow (CMO) causing the flow rates to only change at a feed or side draw in the column. In a real column, however, the various components have different latent heats of vaporization as a result we do not have equal moles being condensed and vaporized on each stage. When this assumption is relaxed, we must calculate the enthalpies of the internal and feed/product streams in and to/from the column. In order to do so we must select a different enthalpy model: in the thermodynamic models under 'Properties' we select the 'Excess' enthalpy model.

When we redo our column simulation (using the feed at bubble point and a reflux ratio of 2) and look at the flow profiles we see that the internal flows now *do* change from stage to stage. Figure 1.39 shows the difference between using no enthalpy and the excess enthalpy models.

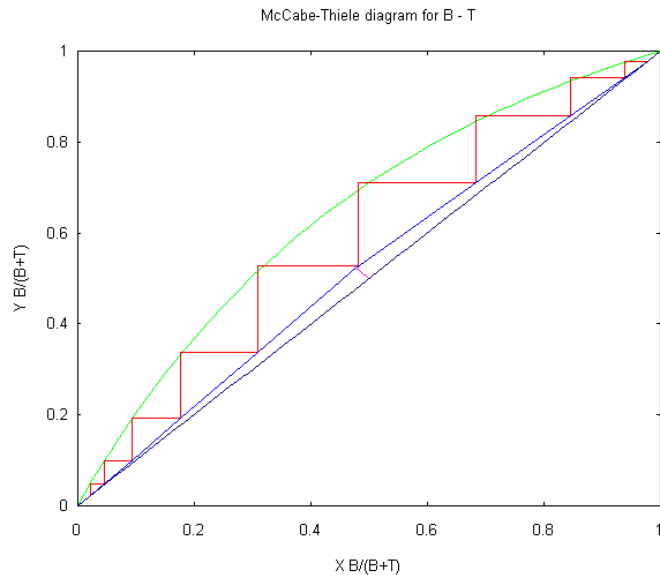


Figure 1.37: McCabe-Thiele diagram with a reflux ratio of 10.

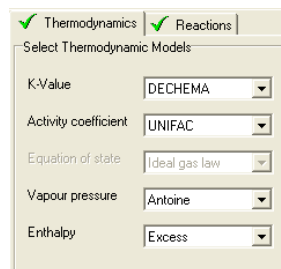


Figure 1.38: Selection of the Excess enthalpy model.

As the internal flows change slightly, so do the slopes of the operating lines. The operating lines in the McCabe-Thiele plot now become curved instead of straight lines (in this case the curvature is rather small so we have left out this plot).

When we select the Excess enthalpy model we are able to construct a Ponchon-Savarit diagram; in *ChemSep* it is actually just a H-xy plot (Figure 1.40). Note that the vapor enthalpies are positive and liquid enthalpies are negative, the difference being the heat of vaporization. Vapor enthalpies are positive because the reference point for enthalpies is taken as gas at 298 K. As the

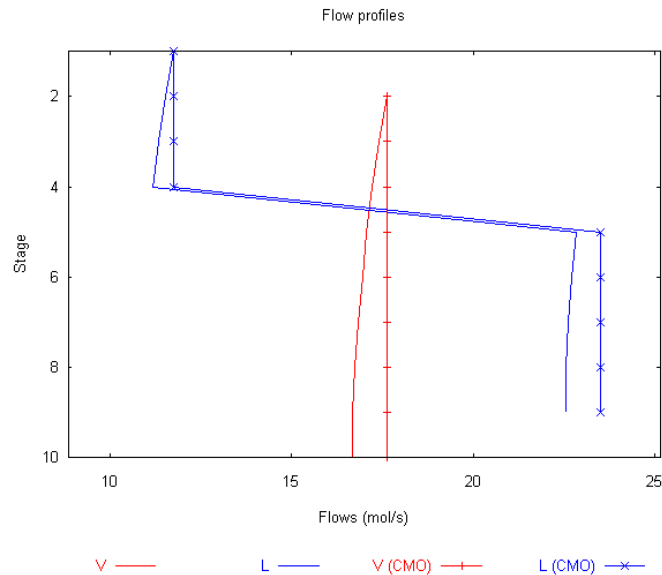


Figure 1.39: Difference in internal flows for no enthalpy model (dotted lines, CMO) and 'Excess' enthalpy model (solid lines).

column temperatures are higher than 25°C , the vapor enthalpies are larger than 0 (see 10.1.7 for how enthalpies are calculated). It is important to know what the reference point is when you are using enthalpies!

Another aspects of real column behavior is that the stages do not operate at equilibrium because of mass transfer limitations. So we make use of a stage efficiency that describes the degree of which equilibrium is approached. The most commonly used efficiency definition is the Murphree stage efficiency:

$$E^{MV} \equiv \frac{\text{actual change in vapor}}{\text{change in vapor for equilibrium stage}} \quad (1.18)$$

or

$$E_j^{MV} \equiv \frac{y_j - y_{j+1}}{y_j^* - y_{j+1}} \quad (1.19)$$

where y_i^* is the vapor composition in equilibrium with the liquid composition x_i on the stage j . With this definition the equilibrium relations are

$$0 = E_i^{MV} K_i x_i - (1 - E_i^{MV}) y_{i,j+1} - y_i \quad (1.20)$$

Note that when the efficiency is one this reduces to

$$0 = K_i x_i - y_i \quad (1.21)$$

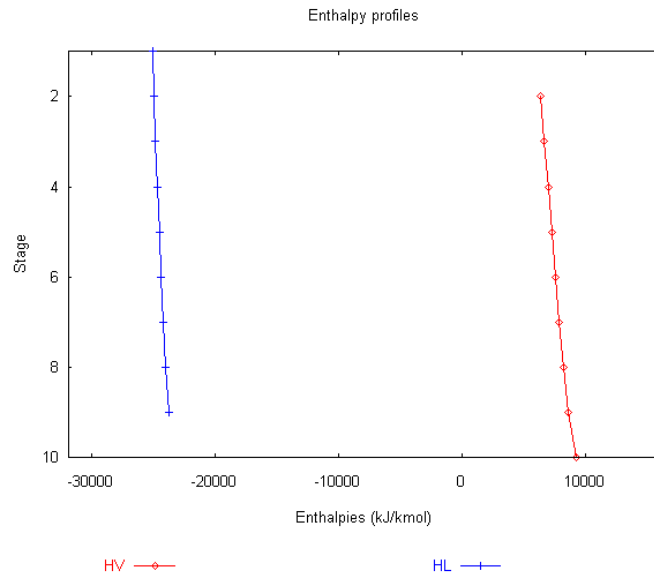


Figure 1.40: H-xy plot.

which equals our definition of the K-values.

In actual columns each stage achieves only a partial separation and the efficiency is defined as the achieved fraction of what is theoretically possible. In most cases the Murphree stage efficiency is taken to be equal for all components. In a binary system, this is theoretically correct. However, this is not the case for multicomponent systems. In fact, in the next chapter we will see the limitations of this assumption.

Stage efficiencies can be entered in the appropriate input spreadsheet under the Specifications – Efficiencies tab. Here, a default efficiency is defined for all the stages except for the condenser and reboiler (both are assumed to be at equilibrium). In an actual column the efficiencies change from stage to stage. Let's assume a stage efficiency of 50% and resolve the problem, see Fig. 1.41. When we compare Figure ?? with Figure 1.33 we see that the separation is worse when the Murphree efficiency is smaller than one — as can be expected, of course. Note that efficiencies can also change the optimal feed location and/or quality.

In the McCabe-Thiele plot the ratio in Eq. (1.18) represents a fraction of the vertical line that is drawn from the operating line to the equilibrium line. Each stage only achieves a separation equal to this fraction of the (vertical) distance between the two lines. The reboiler and condenser are exceptions: they are assumed to operate at equilibrium.

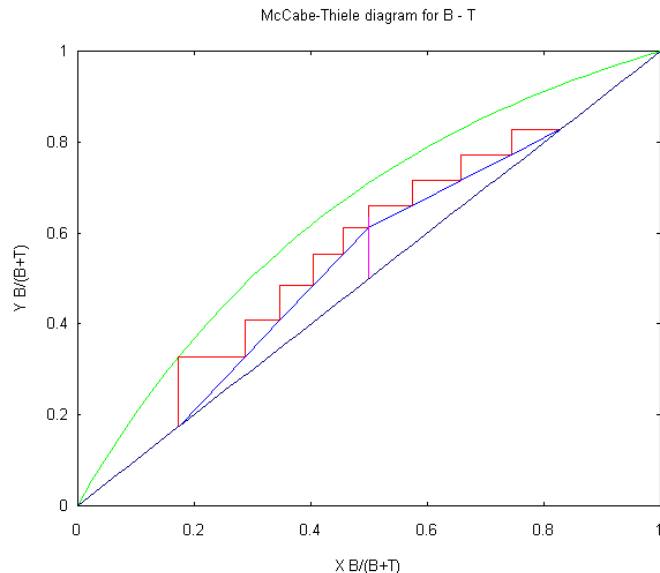


Figure 1.41: McCabe-Thiele diagram for a column with a saturated liquid feed, reflux ratio of 2, and a Murphree stage efficiency of 0.5.

1.5 Parametric Study

When the column simulation converged we often want to study the influence of our column specifications. For example, the reflux ratio has a profound influence on the column operation with respect to product purities as well as condenser temperature. *ChemSep* allow you to do this automatically via the Parametric Study option under the Analysis menu. Figure 1.42 shows how this can be done. We selected to vary the reflux ratio for our initial problem over a range of 1 to 5. We have to select each variable we vary as well as the start and end value. All input variables are varied simultaneously from their start to end values with increments determined by the number of steps and their individually set start and end values. When you want to select a new variable *ChemSep* shows you a dropdown list of all the variables you can vary.

Each result we want to see we need to select as well, so that the results are stored after each simulation and can be used for plotting at a later stage. Again we can select result variables with a dropdown list. Note that there are more variables to pick from than that can be varied as specifications. Figure 1.43 shows the plotted results for when we varied the reflux ratio. The mole fraction of the heavy component in the distillate is drawn on the left axis on a logarithmic scale. Note that the purity of the distillate increases with increasing reflux ratio. Also note that the condenser temperature asymptotically reaches the boiling point of Benzene at the specified

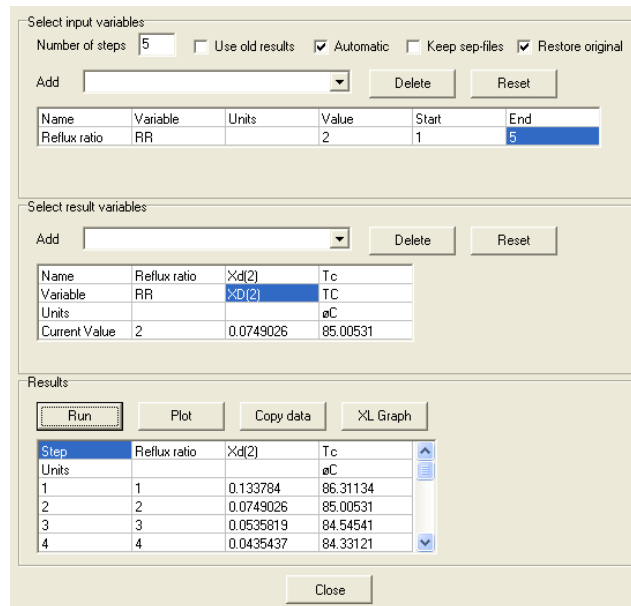


Figure 1.42: Parametric study varying the reflux ratio from 1 to 5.

pressure (1 atm).

1.6 Exercises

1.6.1 High Pressure Distillation

A 'C2 splitter' is a high pressure column used to separate the important chemical ethylene from ethane. The two components boil at very low temperatures, so the column must be operated at as high a pressure as possible in order to be able to condense the column overhead. As the relative volatility is reduced at higher pressures, this makes the separation even more difficult. At high pressures that approach the critical point we need to use an equation of state to describe the phase equilibria. The vapor phase becomes much denser and the density difference between vapor and liquid is much smaller, hence, the physical separation also becomes more difficult. At the critical point the density difference vanishes and computational problems will manifest themselves at such high pressures. The column pressure is determined by two considerations:

- it should be as high as possible to reduce refrigeration costs for the reflux
- it must be below the critical pressure of both components

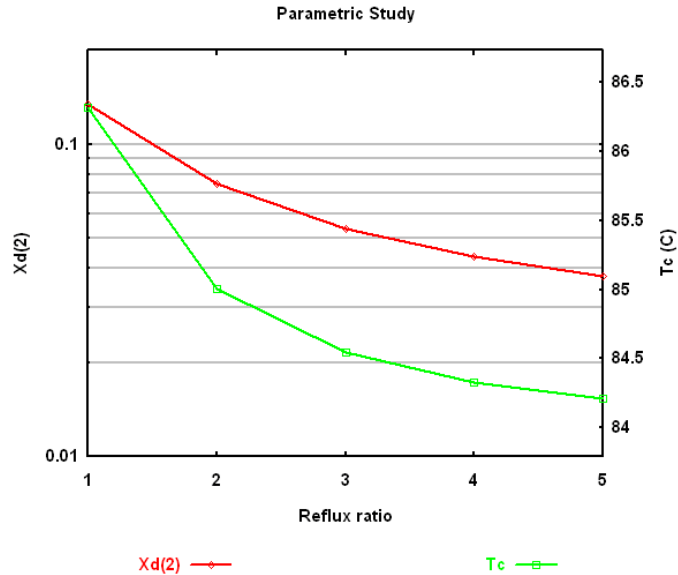


Figure 1.43: Composition of heavy component in the distillate and condenser temperature as function of the reflux ratio.

- it must be low enough so that the density difference between vapour and liquid is large enough to cause an efficient phase separation in the downcomers of the distillation trays

In practice the overhead temperature would be between -30 and -10 °F. Design a column to separate ethylene and ethane. The feed, which consists of 100 mol/s of both components is at its boiling point at the column pressure. The ethylene top product has to be very pure, with a mole fraction of ethane of less than 0.001, in order to be polyethylene grade quality. We also wish to recover at least 90% of the ethylene in the feed.

1.6.2 Column Sequence

Design a sequence of columns to separate the following mixture: 14.08 lbmol/h n-C4, 19.53 lbmol/h n-C5, 24.78 lbmol/h n-C6, and 33.94 lbmol/h n-C8. The column pressure is 1.7 atm and all product streams must be 98.5% pure.

1.6.3 Propanol Removal from Ethanol - Water Mixture using a Side Drawoff

Ethanol from fermentation contains traces of any number of components. These often have to be removed down to very low levels because of their smell. This is not always easy.

Design a distillation column (using ChemSep) to separate a feed of 10 *mol/s* ethanol and 89 *mol/s* water. These two components form an azeotrope with a mole fraction of about 0.89 ethanol (this is at atmospheric pressure, the azeotrope mole fraction changes with pressure). The top product should have a composition close to the azeotropic composition, the bottom product should be almost pure water. Use DECHEMA/UNIFAC/Antoine/Excess for the thermodynamic models.

Add a trace of 1-propanol (1 *mol/s*) to the feed of the column created above. Rerun the simulation and look at the liquid composition profiles. Why is there a maximum in the mole fraction of 1-propanol? To help answer this question look at the K-value profiles and answer the questions below.

1. What are the normal boiling points of ethanol, 1-propanol, and water? (Use the component search engine and select components by boiling point).
2. Do you think that a mixture of ethanol and 1-propanol is almost ideal or strongly non-ideal? Why?
3. Do you think that a mixture of 1-propanol and water is almost ideal or strongly non-ideal? Why?
4. How do you think you might test your answers to questions 2 and 3?
5. In the column created in part 2 above is the 1-propanol in the top part of the column (where ethanol is the major component) trying to go up or trying to go down? Why?
6. In the column created in part 2 above is the 1-propanol in the bottom part of the column (where ethanol is the major component) trying to go up or trying to go down? Why?

After some time 1-propanol can accumulate in the middle of the column and it may cause all kinds of problems. We can avoid this by removing the propanol with a side-draw. Redesign the column so that you remove all (or nearly all) of the propanol with a side stream. At the same time try to maximize the propanol concentration in the side draw (it can be much higher than in the feed).

1.6.4 Extractive Distillation

Toluene and n-heptane are difficult to separate by simple distillation. They can, however, be separated quite easily by extractive distillation using phenol as a solvent. Design a facility to process a stream containing 200 *lbmol/h* toluene and 200 *lbmol/h* of n-heptane that is at 200

$^{\circ}F$ and 20 *psia*. The design objective is to obtain a heptane product of at least 99 *mole%* and a maximum of 2.0 *lbmol/h* of heptane in the toluene product. Phenol is available at 220 $^{\circ}F$ and 20 *psia*. In view of the high cost of the solvent, your design should provide for solvent recovery and recycle.

1.6.5 Azeotropic Distillation

Azeotropic distillation columns can be very difficult to model in the sense that getting the equations to converge can be very hard to do. Some simulation programs can have serious difficulties with these problems. Here we investigate the very well known problem of separating ethanol (1) from water (3) using benzene (2) as an entrainer.

Set up an azeotropic distillation column with 41 stages, a partial reboiler, and no condenser. There is a feed containing 85.64 *kmol/h* ethanol and 14.36 *kmol/h* water at a pressure of 1.027 *bar* and a temperature of 311 *K* to stage 5. In the absence of a condenser we simulate the reflux flow with a second feed to stage 1 containing 103.21 *kmol/h* of ethanol, 215.8 *kmol/h* benzene, and 56.29 *kmol/h* water at a pressure of 1.013 *bar* and a temperature of 298 *K*. The specifications provided here are very similar to a version of this problem studied by Prokopakis and Seider (*AIChEJ*, Vol. 29, p. 49, 1983). We will make the following modifications: The bottoms flow rate is 83.0 *mol/h* and the number of stages is 41.

There are no less than three solutions for this set of specifications; your task is to find them. These calculations are VERY sensitive to the VLE models and parameters. Here we will use the DECHEMA / UNIQUAC Q' / Antoine combination of models.

Since these calculations can be hard to converge it is advisable to limit the changes in flows and compositions between iterations. Start by pressing *F6* and go to Solve options. In that spreadsheet set the default number of iterations to 100 (but don't be alarmed if you sometimes need more than this), the flow limit to 0.3 and the composition step to 0.1. This helps to stop the flows, temperatures and mole fractions from taking on ridiculous values during the computations.

Solution number 1 can be found by letting *ChemSep* do what it wants. Study the profiles carefully. Is this a good way to operate the column?

A second solution can be found as follows.

1. Set the efficiency to 0.1 and solve the problem using the Automatic initialization.
2. Go to Solve options and change the initialization option to Old Results.
3. Change the efficiency. Go back to step 1 and solve the problem again.

Look at how the profiles change. Compare the final set of profiles with those of the first case. Which is more desirable?

Now try and find the third solution. You can use a method very similar to that used above, just change the value of the efficiency used in step 1. Good luck.

Try repeating this exercise with UNIFAC instead of UNIQUAC as the activity model. Are the profiles different? Can you find multiple solutions for this case? Think about the consequences for design and operation of a plant that displays this kind of behavior. Note that some columns have even more than three steady state solutions. Examples with 5 and 9 are also known!

1.6.6 Separation of Benzene and n-Pentane using Liquid-Liquid Extraction

An equimolar mixture of n-pentane (1) and benzene (2), at a total flow rate of 100 mol/s is to be separated by liquid-liquid extraction. This operation might be carried out in equipment very much like distillation columns except that there is no reboiler or condenser. Our extractor is to operate at 50°C . It is required to recover 99.98% of the benzene. Further the purity of the extract is to be such that not more than 2.5 mole% n-pentane can be present in the extract product. The solvent is sulfolane (3).

Determine the solvent flow rate and the number of equilibrium stages required to achieve the recovery and purity targets specified above. Assume that the optimum design corresponds to the use of the lowest possible solvent to feed ratio for a specified number of stages. In addition, answer the following questions:

1. What is the chemical formula of sulfolane?
2. What is its structure?
3. What is the boiling point of sulfolane?

Table 1.3: UNIQUAC interaction parameters (K) for the LLX example.

i	j	A_{ij}	A_{ji}
1	2	179.86	-95.7
1	3	375.93	247.77
2	3	131.51	-6.28

1.6.7 Absorber Design

A gas stream flowing at 80 mol/s contains ammonia (mole fraction 0.416) and air (mole fraction 0.584). The temperature of the gas stream is 20°C and the pressure is 1 atm . The ammonia content must be reduced to ppm levels before the air stream can be vented to the atmosphere.

Fortunately, ammonia is readily absorbed in water so it has been decided to build a simple absorption column to process the stream in question.

Your task, to design the column, is complicated by the fact that the design program available to you does not have the most appropriate thermodynamic models for this system (Henry's law). The staff thermodynamicist has suggested that simple polynomial models (which are available in your software system) will suffice for the K-values and enthalpies over the temperature range of interest. Analysis of some ancient (graphical) data in the company files has shown that the K-values can be correlated by a simple expression of the form:

$$K_i^m = \alpha_i + \beta_i T \quad (1.22)$$

where T is the temperature in Kelvin. The enthalpy of the vapor and liquid phases can be calculated from similar polynomials:

$$H_i^V = A_i + B_i T \quad (1.23)$$

$$H_i^L = a_i + b_i T \quad (1.24)$$

All the parameters are given in Table 1.4.

Table 1.4: K value and enthalpy parameters ($J/kmol$) for the absorber

Component	α_i	β_i	m	A_i	B_i	a_i	b_i
Ammonia	-32.285	0.1137	1	3.57e7	3.52e4	0	7.5e4
Water	-1.231833	0.00427325	1	4.3486e7	3.35e4	0	7.5e4
Air	-191835	900	1	0	2.92e4	0	7.5e4

In addition to providing a column design adequate to the task at hand, please explain

1. The physical significance of the parameters in the K-value and enthalpy correlations.
2. The shape of the temperature profile for the optimum column design.

References

J.D. Seader, E.J. Henley, *Separation Process Principles*, Chapter 11.7, Wiley, New York (1998), pp. 631–641.

Chapter 2

Nonequilibrium Columns

In this chapter we introduce the modelling of multicomponent separation processes by using the nonequilibrium model. We start with the benzene-toluene distillation column that we encountered in the preceding chapter, followed by a depropanizer with sieve trays, a nonisothermal packed absorber where acetone is absorbed in water, and finally the extractive distillation of methyl cyclohexane using phenol as a solvent in an ‘old’ column with bubble cap trays. These examples illustrate that when working with systems with more than two compounds we need abandon the use of Murphree efficiencies and switch to a full multicomponent mass transfer model instead. And in doing so we must start using the Baur efficiency as performance indicator for our column internals hardware.

2.1 Equilibrium and Nonequilibrium Models

Most mathematical descriptions of a distillation column begins with the concept of ‘stages’: fairly large parts of the column that have well defined boundaries (Figure 2.1). For a tray column, we can regard a single tray as a stage. There are well defined flows entering and leaving one of our stages. These can be the main vapor and liquid flows through the column, or feed or drawoff flows. There is no natural boundary to the stages in packed columns (unless we consider packed sections as stages). Even so, chemical engineers often try to subdivide packed columns into stages, sometimes with ludicrous results.

Conceptually, the simplest kind of stage is the ‘equilibrium stage’ or ‘theoretical plate’. For such a stage (which does not exist in reality), the (vapor and liquid) streams leaving the stage are assumed to be in equilibrium with each other (Figure 2.2). This implies that the chemical potential of any component has the same value in vapor and liquid, and also that vapour and liquid have the same temperature.

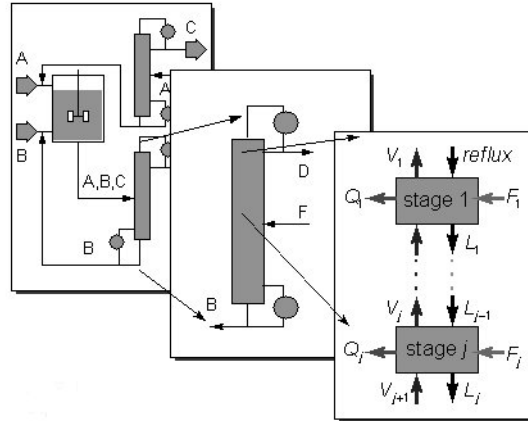


Figure 2.1: Plant, separation equipment, and stages.

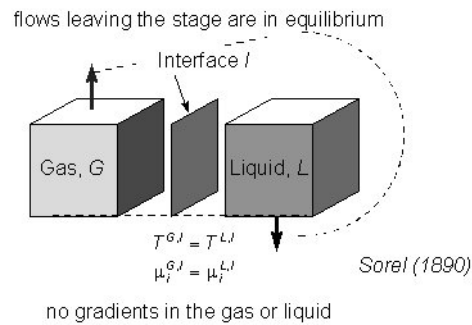


Figure 2.2: The equilibrium stage model.

To describe an equilibrium stage model, we ‘only’ need phase equilibrium data (thermodynamics) and mass and energy balances. The model is beautifully consistent, and has attracted the attention of generations of engineers and thermodynamicists. With the equilibrium stage model we can simulate many of the properties of real systems.

The equilibrium stage model it has one great weakness: *it has no direct connection to real equipment*. Of course, engineers have found ways of coupling our theoretical models to practical equipment. The two concepts mostly used for this purpose are (Figure 2.3):

- the (Murphree vapor) ‘efficiency’ of a tray or plate and

- the ‘Height Equivalent to a Theoretical Plate’ (HETP) for packings.

When using a Murphree stage efficiency, we assume that the change in vapor composition on a real tray is a certain fraction of that obtained in an equilibrium stage. This allows us to approximate the behaviour of real trays. The concept works quite well for binary separations; the two components there have equal efficiencies. Also the efficiencies are often fairly constant along a column, and therefore an efficiency is a useful way of summarising practical experience. It often has a value of about 0.7.

For packings, we look at the composition of the liquid at a certain height. We then assume that further up in the column, we will be able to find vapor with a composition that is in equilibrium with the liquid considered. The difference in height between these two points is the HETP. Also here, this works fairly well for binary mixtures. The HETP’s are the same for the two components and they are often fairly constant along a column. Experience shows that the value of the HETP is usually a few tenths of a meter.

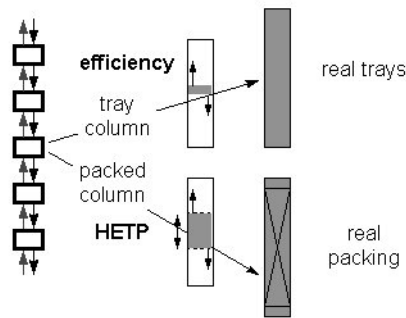


Figure 2.3: Tray efficiency and ‘Height Equivalent to a Theoretical Plate’ lead to real equipment.

Although the efficiency and the HETP have been useful for summarising experience on trays and packings, these concepts have two serious deficiencies:

- they are not easily related to the construction and behaviour of equipment, and
- in multicomponent mixtures, their behaviour can be *extremely confusing*.

2.1.1 ‘Exact’ models

If we do not wish to use efficiencies or HETP’s, what should we do? The dream of the modern engineer is to subdivide each piece of equipment into zillions of little elements $dx.dy.dz$, each of which will contain part of one of the phases in the equipment (Figure 2.4). We should then

set up difference forms of the Navier-Stokes equations to define the hydraulic flows, the Fick or Maxwell-Stefan equations to calculate diffusion flow, the Fourier equations for energy transfer together with all of the required balances and boundary conditions. This vast array of equations will then to be solved by our brilliant parallel supercomputer algorithms, and they will tell us everything we wish to know. Such a model can be used for trays, for packings or for any other piece of equipment. It will deliver beautiful and impressive multidimensional color plots. It will describe unsteady, turbulent, chaotic two phase flow, the breaking up and coalescence of drops, their moving and deforming boundaries, interfacial motion and mass transfer, incomplete and irregular wetting of packings, gas and liquid entrainment...

Unfortunately, there are still a few problems. For a realistic simulation of the important phenomena in a column, our guess is that we would need elements with dimensions of about one micrometer. That is one million elements per meter, or 10^{18} elements per m^3 of column. This number is several times the number of bytes on a multi-gigabyte hard disk. So it looks as if we still have a long way to go before we can simulate the details of multiphase flow. Of course this also means that the subject is a challenging one, and possibly a good topic to let our students and young engineers sharpen their teeth. However, with this approach, we should not expect complete solutions to practical problems in the near future.

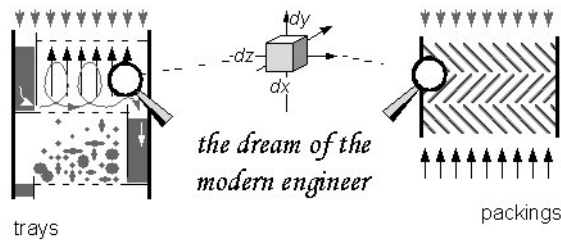


Figure 2.4: Exact models: the dream of the modern engineer.

2.1.2 The Nonequilibrium Model

The nonequilibrium model is less ambitious. It contains a smaller number of elements per m^3 than the 'exact' models above. Say, five, instead of the 10^{18} that we were considering. We choose these elements such that they allow a simple simulation of what we think are the important characteristics of our mass transfer equipment:

- the equilibrium distribution of all components between the two phases (thermodynamics),
- the effects of large scale flow and mixing patterns, and
- the effects of diffusional heat and mass transfer resistance near phase interfaces.

How few elements we choose, is a balancing act. With too few elements our model will not be realistic, with too many elements it will be impossible to obtain the model parameters (although, with 10^{18} elements, you do not need *any* model parameters).

The motion of the gas and liquid on and around the trays or packing has a large effect on equipment performance. The ideal flow pattern in separation equipment usually is countercurrent plug flow. Modern structured packings give patterns that are fairly close to this ideal. However, even there, phenomena like turbulence, differences between the channels, and maldistribution of liquid or vapor at feed points cause deviations from ideality (Figure 2.5).

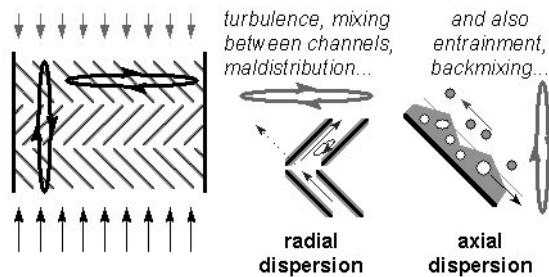


Figure 2.5: Even in a structured packing, flow is not ideal.

Entrainment of liquid by the vapor, and of vapor by the liquid, causes backmixing. In large, long column sections, irregularities of the flow have been found to develop, especially near walls and due to the contacting planes between packing elements.

On trays, the vapor is often assumed to pass upwards in plug flow. The liquid has a cross flow pattern; at the sides of columns there can be more or less stagnant zones. On short tray passes the overall effect is often similar to that of complete mixing of the liquid. If we have a spray of drops on the tray, there will be a range of drop sizes and the behaviour of small and large drops may differ considerably. We may expect small drops to approach equilibrium more rapidly than large drops. Something similar applies to the bubbles of different sizes in a froth.

The simplest flow model is that where both phases are assumed to be ideally mixed (Figure 2.6). This mixed flow model only contains two bulk phases with convective flows in and out. In addition it contains three other elements: the phase interface and two mass transfer resistances on either side of the interface. We discuss these elements in the next section.

Flow models that are a bit more realistic are (see Figure 2.7):

- on a tray: liquid mixed, vapor in plug flow, no entrainment/backmixing,
- in a packing: both phases in plug flow, no maldistribution.

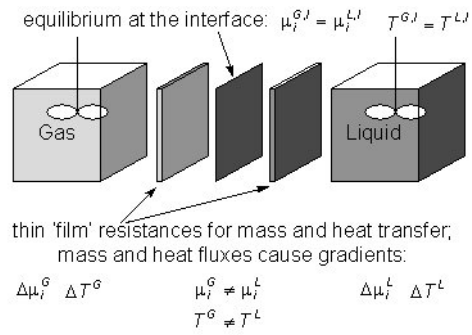


Figure 2.6: The simplest nonequilibrium model.

Even these models are not always adequate.

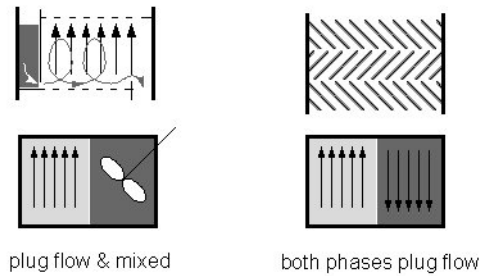


Figure 2.7: More realistic flow models.

There are many different ways to describe more complicated flow patterns (Figure 2.8). We can approximate plug flow by stacking a number of mixed elements. When a phase consists of coarse and fine parts (such as large and small drops), we may split it in two pseudo phases. We may expect a convective exchange between these pseudo phases. Entrainment and backmixing can be simulated by allowing extra flows going in and out of each phase. Finally, maldistribution can be simulated using parallel stages with different flow ratio's. The problem with these more complicated models is, that we do not know their parameters well enough.

At the interface between the two phases, we assume (also in a nonequilibrium model) that equilibrium exists. (In absorption with a chemical reaction, this may not be true, but we do not consider that here).

The same equations that we use in equilibrium stage calculations are used to yield the distribution coefficients of all components *at the interface*. In nonequilibrium calculations we also

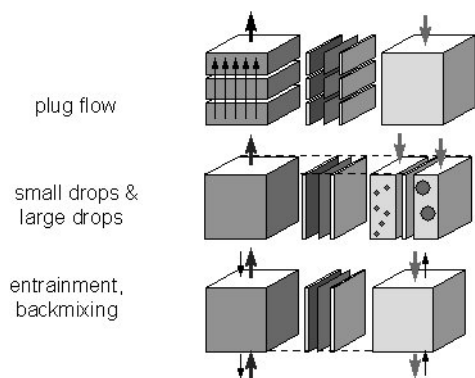


Figure 2.8: Complicated flow patterns using simple elements.

use the same equations to calculate the driving forces for mass transfer (which are the chemical potential gradients, as we shall see below).

An essential part of a nonequilibrium model is the description of mass and heat transfer between the phases. In most equipment, flow of both phases is highly turbulent (this applies both to the continuous and the dispersed phase). Turbulence provides a rapid levelling of concentration and temperature differences. So on not-too-large scales (say up to ten centimeters), the concentrations and temperatures in the bulk of fluids can be taken as homogeneous. However, near phase interfaces, turbulence dies out (Figure 2.9). Eddies do not pass across interfaces. Near these interfaces, only three transport mechanisms remain, all of which are much slower than turbulence:

- ‘drift’: convective transport of both heat and matter due to an overall displacement of matter across an interface,
- diffusion: the transport of matter due to composition gradients and molecular motion within the fluid, and
- conduction: the transport of energy by a temperature gradient and molecular motion.

The layers (or ‘films’) where turbulence does not dominate transport, are very thin: roughly one tenth of a millimeter in gases and ten micrometers in liquids.

The traditional description of mass transfer begins with Fick’s law. This states that flux of a component (in a binary mixture) is proportional to the concentration gradient of that component and a diffusivity. In addition, the component may also be carried along by drift:

$$N_i = \text{diffusion} + \text{drift} = -c_t D_i \frac{dx_i}{dz} + N_{total} x_i \quad (2.1)$$

For most engineering applications, a difference approximation of this equation suffices:

$$N_i = -k_i^{binary} c_t \Delta x_i + N_{total} x_i \quad k_i^{binary} = \frac{D_i}{\delta} \quad (2.2)$$

The parameter k_i^{binary} is a binary mass transfer coefficient.

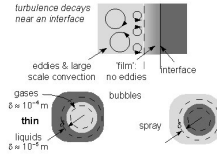


Figure 2.9: Mass transfer resistance: the ‘film’ near interfaces.

Unfortunately, the description above does not work well for mixtures with more than two components. (And most mixtures do contain more than two components!) This is for two reasons:

- it does not take the interactions between the different components into account, and
- it does not allow for the fact that the driving force for mass transfer is not the concentration gradient, but the gradient of the chemical potential.

The first effect is often important, especially for the behavior of trace components. The second effect can be important in systems where liquid resistance dominates.

If we take the trouble to make a nonequilibrium model of a column, we can best use a fundamental multicomponent model of mass transfer through the interface. A good choice are the Maxwell-Stefan equations which say that the driving force on i will cause it to move with respect to the mixture. This motion is counteracted by friction with the other species j .

The driving force is the chemical potential gradient:

$$(\text{driving force on } i) = -\frac{d\mu_i}{dz} \quad (2.3)$$

This is a real driving force, with units of Newtons per mole. If we have equilibrium, we have no gradients, no driving forces and no motion. The chemical potential contains contributions due to mole fraction and the activity coefficient of i :

$$\mu_i = RT \ln(\gamma_i x_i) + \text{constant} \quad (2.4)$$

The contribution of the mole fraction gradient is similar to that of the concentration in the Fick equation. However, that equation does not consider the effect of non-ideality via the activity coefficient.

The friction force between i and j (per mole of i) is proportional to the local mole fraction of j , a friction coefficient ζ_{ij} and the difference in drift velocities between the two components:

$$(\text{friction})_{ij} = x_j \zeta_{ij} (u_i - u_j) \quad (2.5)$$

When we equate the driving force with the sum of all friction forces on i , we obtain the Maxwell-Stefan equations:

$$-\frac{d\mu_i}{dz} = \sum_{j=1}^n x_j \zeta_{ij} (u_i - u_j) \quad (2.6)$$

These equations allow a consistent description of diffusion in mixtures with any number of components.

Usually we are not interested in species velocities through the interface, but in their fluxes (flow rates per unit area):

$$N_i = c_i u_i \quad (2.7)$$

and we normally use multicomponent diffusion coefficients instead of friction coefficients. Finally, we will use a difference form of the resulting equations that contain the multicomponent mass transfer coefficients:

$$k_{ij} = \frac{D_{ij}}{\delta} \quad (2.8)$$

We will not write out these equations; the important thing to understand, is that the multicomponent diffusivities and mass transfer coefficients can be estimated fairly well from binary data.

In addition to the mass transfer equations, we need heat transfer equations. These require estimates of the heat transfer coefficients in both phases, and they must incorporate heat transport through the interface due to drift.

2.1.3 Practicalities

We have restricted ourselves to a brief (and incomplete) summary of modern approaches to the modelling of separation processes like distillation and absorption. Chapter 13 describes the equilibrium stage model, providing all the equations and discussing in some detail many of the methods used to solve the model equations. Chapter 14 provides more detail for the nonequilibrium model. The numerical methods needed to solve the model equations are described in Chapter 11.

We only need two kinds of physical properties for the equilibrium stage model: Equilibrium ratios, also known as K-values, and enthalpies. The nonequilibrium model requires more physical properties. For this model we need, in addition to the thermodynamic data identified above, all properties that are required to describe the flow, heat transfer and mass transfer in the equipment, such as: densities, viscosities, surface tension, interfacial area, diffusion coefficients, thermal conductivity, heat and mass transfer coefficients, and so on.

Equilibrium and nonequilibrium models also differ greatly in how detailed a description of the equipment is required. In the equilibrium model, we only need to give the number of stages and

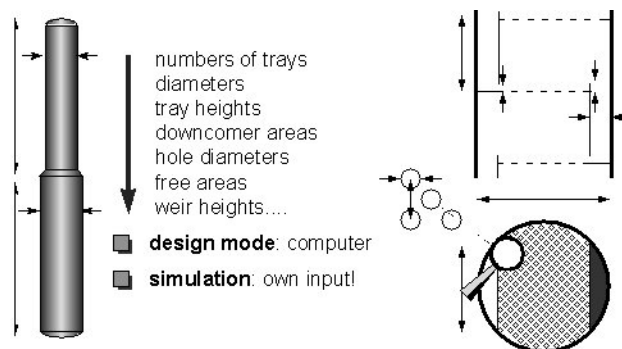


Figure 2.10: The NE-model contains a ‘drawing’ of the column.

the location of the different feed and product withdrawal points. The nonequilibrium model, in contrast, needs to contain a complete ‘drawing’ of the column (Figure 2.10): the number of trays in the different sections, the diameters of the sections, the tray heights, the downcomer configurations, the hole diameters, the free areas, the weir heights, and so on and so on... When we are designing a column (see Chapter 14), we can leave it to the program to determine all of these parameters. However, when we are simulating an existing column, we must provide this information. The additional detail also allows us to see the effects of changing equipment design parameters. This is an important extension of the classical equilibrium stage model. The nonequilibrium models also are applicable to multicomponent separations; they do not need efficiencies or HETP’s which are not easily predicted for multicomponent mixtures. Finally, *ChemSep*’s nonequilibrium models are quite easy to use. Let us illustrate that by using the example of the previous chapter.

2.2 A Sieve Tray Column

Using the same benzene-toluene example of the previous chapter we now want to simulate a sieve tray column using the nonequilibrium column model. This column model is selected in the operation menu of *ChemSep*.

Upon selection of the nonequilibrium model the column configuration is specified in the same way as done for an equilibrium-stage column. However, instead of specifying the stage efficiencies for the column, we need to specify the column layout *design* (already marked with a red cross as the design specifications are incomplete). This design encompasses the type of internals, such as trays or packing, but also the parameters of the column internals, such as the sieve tray hole diameter and the column diameter.

Upon entering the design spreadsheet the *ChemSep* interface automatically finds the appropriate

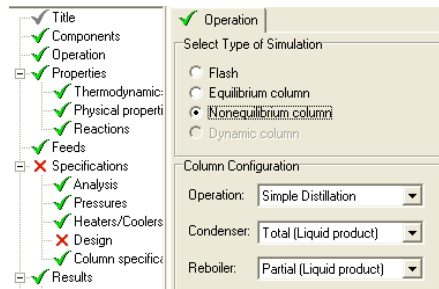


Figure 2.11: Selecting the nonequilibrium column model.

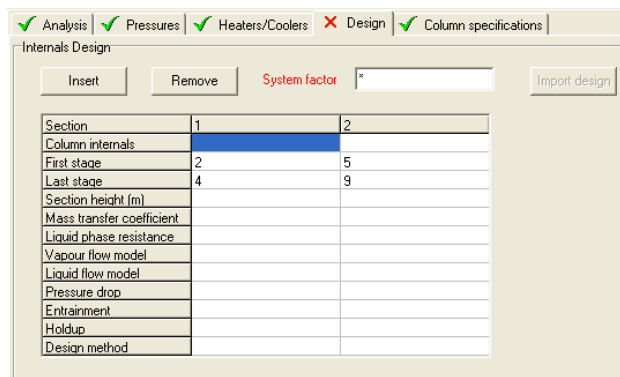


Figure 2.12: Column design.

sections in the column. Here, a column section means any part of the column between feeds and/or product streams. In our simple example, there is a rectification section (stages 2-4) and a stripping section (stages 5-9). For each section we need to select the type of internal (Fig. 2.13) to be installed as well as the layout of the internal. The column layout design can in principle be different for each stage in the column, however, in practice, the design is taken to be constant within a section of the column. Sometimes, the same design is used for the whole column.

Here we will use sieve trays as column internal. Once we selected this type of internal we are prompted for the layout parameters specific to this type of internal, see Figure ???. The most important parameter in this list is the column diameter, however, all of the other parameters play a role in determining the mass transfer and the hydraulic performance of the internal. Some of the parameters are also interrelated, such as the various area's and the column diameter, and must be set in a consistent manner. Note that you can enter the parameters in any units, but that they will always be converted to the units as selected in global units settings.

Section	1	2
Column internals		▼
First stage	Bubble cap tray	
Last stage	Sieve tray	
Section height (m)	Valve tray	
Mass transfer coefficient	Dumped Packing	
Liquid phase resistance	Structured Packing	
Vapour flow model	Equilibrium stage	
Liquid flow model		
Pressure drop		
Entrainment		
Holdup		
Design method		

Figure 2.13: Column internals type selection.

Now we come to an inherent problem of using nonequilibrium models: how do we know the column diameter for the sieve tray so that the tray will operate satisfactorily? The trays will weep when the flows are too small or flood when they are too large. To size the trays we first need to know the internal vapor and liquid flowrates and have estimates of the physical properties such as the densities. However, these depend on the simulation results themselves: the vapor density is dependent on the temperature and the composition of the vapor. The nonequilibrium model allows us to simulate an existing column where we know the layout, a simulation in *rating* mode, but how can we simulate a column for which we do not have a column layout?

ChemSep solves this dilemma through the built-in *design mode*. When we do not specify the column diameter of a section (that is, we leave it as "***"), the design mode is invoked automatically and a built-in design method computes a layout for the selected type of internal "on the fly", during the simulation. The interface shows per section which mode is selected next to the section number: each the *design* or the *rating* mode. The user can specify start values for certain layout parameters but the design methods are written in such a manner that they do not require any user input. In other words, it is only necessary to specify the *type* of internals for each column section and *ChemSep's* design mode will compute valid layouts for each section while it solves the problem. This makes it very easy to use the nonequilibrium model and it can also be used in flowsheet calculations where flowrates are not known *a priori*. The design mode is explained in detail in Section 16.5.

As here we do not yet know anything about the physical design of our benzene-toluene column, we will use the design mode for both sections and leave all the layout parameters as unspecified (as "***").

Besides the specification of internals type and the actual layout parameters, we also need to select the various models that determine how the performance of the internal is calculated. These include correlations for the Mass Transfer Coefficients (MTC's), liquid phase resistance, vapor and liquid flow models, pressure drop, liquid entrainment, and holdups. In order to facilitate the design mode a design method and parameters for this design method also need to be selected.

ChemSep assigns defaults for these models depending on the type of internals that the user

Section 1: Column internals

Library

Save

Reset

Column diameter (m)	*
Tray spacing (m)	*
Number of flow passes	*
Liquid flow path length (m)	*
Active area (m ²)	*
Total hole area (m ²)	*
Downcomer area (m ²)	*
Hole diameter (m)	*
Hole pitch (m)	*
Weir length (m)	*
Weir height (m)	*
Weir type	*
Notch depth/weir diameter	*
Serration angle (rad)	*
Downcomer clearance (m)	*
Deck thickness (m)	*
Downcomer sloping (%)	*

Figure 2.14: Sieve tray layout specification.

Section	1 (design)	2
Column internals	Sieve tray	
First stage	2	5
Last stage	4	9
Section height (m)		
Mass transfer coefficient	AIChE	
Liquid phase resistance	Included	
Vapour flow model	Mixed flow	
Liquid flow model	Mixed flow	
Pressure drop	Fixed	
Entrainment	None	
Holdup		
Design method	Fraction of flood	

Figure 2.15: Sieve tray performance and design models selection.

selected. Usually this provides a selection that will "work" and provide a converging solution. In this case we see that the AIChE mass transfer model has been selected and that we include the liquid phase resistance in the calculation of the mass transfer rates. The phases are considered to be completely mixed, the pressure drop is fixed (that is not computed from the internal flows and tray hydraulics), we do not correct for liquid entrainment, and the tray is designed on a fraction of flood basis.

For sieve trays, we may choose from several different mass transfer coefficient correlations; these are described in Chapter 16. For now, select the AIChE correlation. We recommend this correlation for your first approach to the simulation for it is well behaved as long as the liquid flows do not become very small. It is also a conservative and thus an inherently safe method for predicting the mass transfer efficiency of a tray.

For the vapor flow on the tray, we can choose to model the flows as completely mixed or in plug flow. This choice of flow model has a rather large influence on the mass transfer performance of a sieve tray. Plug flow significantly enhances the mass transfer on a sieve tray. Typically, the vapor is assumed to flow in a plug flow manner through the froth. However, such a choice

Section	1 (design)	2
Column internals	Sieve tray	
First stage	2	5
Last stage	4	9
Section height (m)		
Mass transfer coefficient	AICHE	
Liquid phase resistance	AICHE	
Vapour flow model	Chan Fair	
Liquid flow model	Zuiderweg	
Pressure drop	Harris	
Entrainment	Bubble-Jet model	
Holdup	Chen-Chuang	
Design method	Fraction of flood	

Figure 2.16: Mass transfer coefficient correlations for sieve trays.

Section	1 (design)	2
Column internals	Sieve tray	
First stage	2	5
Last stage	4	9
Section height (m)		
Mass transfer coefficient	AICHE	
Liquid phase resistance	Included	
Vapour flow model	Mixed flow	
Liquid flow model	Mixed flow	
Pressure drop	Plug flow + cross liquid flow co	
Entrainment	Plug flow average y	
Holdup		
Design method	Fraction of flood	

Figure 2.17: Vapor flow model.

does not apply for the liquid flow on a tray. Liquid on small sieve trays (say less than 0.5m) is most likely to be completely mixed, whereas the liquid flow over large trays approaches plug flow. Therefore a dispersion model for the liquid phase is actually best but this approach is more demanding computationally.

The plug flow model actually requires a simulation of infinite number of cells on the tray. This can be approximated by averaging the incoming and outgoing vapor (y) or liquid (x) compositions in the mass transfer rate calculations, or alternatively, with a theoretical model. *ChemSep* provides a whole list of dispersion models that are typically combined with specific MTC models. For example, the AIChE model was developed using the Gautreaux-O'Connell dispersions model. Care must be taken to make a sensible combination as *ChemSep* allows you to combine any set of models!

Pressure drops can either be assumed fixed or calculated from the hydraulic operation of the sieve tray. In the latter case the pressures on each stage no longer are fixed during the simulation but they are allowed to change depending to the operating conditions of the trays. In this example we choose to have the pressure drops estimated by the program.

Two different types of tray design methods are available for the sieve trays: a design based on a specified fraction of flooding or a design method based on a specified maximum pressure drop. Trayed columns often are designed on a specified fraction of flooding, a typical design value is

Section	1 (design)	2
Column internals	Sieve tray	
First stage	2	5
Last stage	4	9
Section height (m)		
Mass transfer coefficient	AIChE	
Liquid phase resistance	Included	
Vapour flow model	Mixed flow	
Liquid flow model	Mixed flow	
Pressure drop	Fixed	
Entrainment	Fixed	
Holdup	Estimated	
Design method	Fraction of flood	

Figure 2.18: Pressure drop model.

Section	1 (design)	2
Column internals	Sieve tray	
First stage	2	5
Last stage	4	9
Section height (m)		
Mass transfer coefficient	AIChE	
Liquid phase resistance	Included	
Vapour flow model	Mixed flow	
Liquid flow model	Mixed flow	
Pressure drop	Estimated	
Entrainment	None	
Holdup		
Design method	Fraction of flood	
Section 1: Design method	Fraction of flood	
	Maximum pressure drop	
Library	Fraction of flooding	*
	Fraction of weeping	*

Figure 2.19: Selecting the design method.

75% fraction of flood (which is *ChemSep*'s default). This means that the *internal* flowrates may increase by about 1/75% before the column would start to flood. Packed columns are typically designed for a specified pressure drop, however.

In this example we will first simulate the column using most of the default model selections of *ChemSep*. We therefore specify the stripping section also to contain sieve trays. It is not necessary for all sections to have identical internals; we are free to combine trayed and packed sections within one column. This might be useful to maintain the same diameter throughout the whole column and thus lower the column cost. However, we do want to make use of a new feature that the nonequilibrium model has over the regular equilibrium model: we want the simulator to compute the actual pressure drop over the trays to see the pressure profile over the column.

Finally, an overall system factor is used in the tray design of the column. This is a general safety factor and it is typically associated with the foaming tendency of the mixture in the column. To see some commonly used values, press *FI* to obtain help on the selection of the system factor. If we use the design mode, the smaller the system factor, the larger our column diameter will be. Here, we choose a system factor of 1.0.

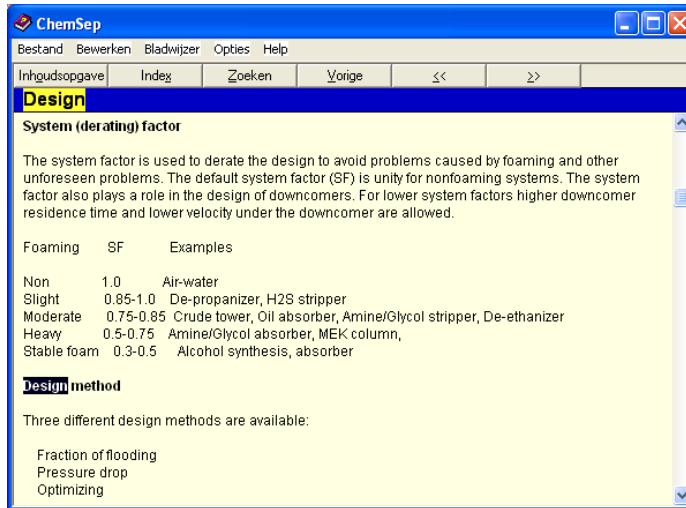


Figure 2.20: Help on system factor.

The final design spreadsheet shows us in one glance the selected type of internals and all the selected models and methods for all the sections (that can be displayed).

The screenshot shows the 'Internals Design' spreadsheet. At the top, there are 'Insert' and 'Remove' buttons, and a 'System factor' input field with the value '1'. The spreadsheet has three columns: 'Section', '1 (design)', and '2 (design)'. The data is as follows:

Section	1 (design)	2 (design)
Column internals	Sieve tray	Sieve tray
First stage	2	5
Last stage	4	9
Section height (m)		
Mass transfer coefficient	AIChE	AIChE
Liquid phase resistance	Included	Included
Vapour flow model	Mixed flow	Mixed flow
Liquid flow model	Mixed flow	Mixed flow
Pressure drop	Estimated	Estimated
Entrainment	None	None
Holdup		
Design method	Fraction of flood	Fraction of flood

Figure 2.21: Final design specifications spreadsheet.

Now we have made a complete specification of the required input and we can return to the input menu and use the solve option to check our input. We will see that there is an inconsistency in the pressure drop specification: previously we had specified a fixed pressure in the whole column but in the design specifications we selected to estimate the pressure drops for the sieve trays. When we use the solve option, the *ChemSep*'s check on the input will prompt us to accept or to correct this.

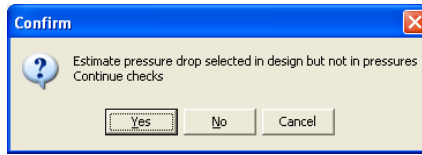


Figure 2.22: Checking the input.

We can correct this by also selecting to use the estimated pressure drops in the column pressures specification, see Figure 2.23.

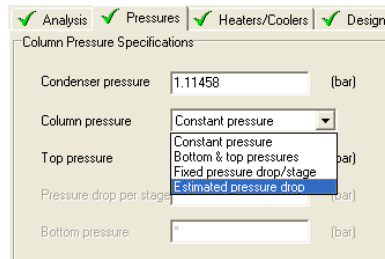


Figure 2.23: Selecting estimated pressure drops.

Now we can try to solve our problem again and we will see that there are no more input problems (assuming you typed correctly all the input mentioned above). Now the nonequilibrium simulator is called and the problem is solved. During the calculation we see that the design mode is on and that after the initialisation of the internal flows the program designs the two sections. The problem solves without there being a need to redesign the trays. The problem rapidly converges in 4 iterations. If we look at the output tables we see an extra table for the internals layout as calculated by the selected design method, see Figure 2.24.

In this case we see that the tray designs for both sections are close in diameter and that in principle we could use just one the same diameters. The actual difference for the top and bottom section is reflected in the downcomer area which is due to the large difference in liquid flowrates between the sections. Note that the tray designs have equal tray spacing, this is because the tray design starts with using certain *default* settings for many tray design parameters. You find these under tools — Design mode settings, see Figure 2.25. For a definition of these defaults, see Section ??.

The user can manipulate the tray design by initializing some of the design parameters for the specific trays in the design window. In this case these values will be used and only when necessary they will be overwritten by the design method. Alternatively you could change the defaults in the `tdesign.def` file but then this will be reflecting all designs. Of course, the tray design the simulator makes is not an optimal design but should be seen as a starting point

Column design:		
Number of sections	2	
System factor (-)	1.00000	
Section	1	2
Column internals	Sieve	Sieve
First stage	2	5
Last stage	4	9
Section height (m)		
Column diameter (m)	0.681000	0.755000
Total tray area (m2)	0.364237	0.447696
Number of flow passes	1	1
Tray spacing (m)	0.610000	0.610000
Liquid flow path length (m)	0.537221	0.515589
Active area (%total)	88.9910	80.6981
Total hole area (%active)	14.0366	13.0400
Downcomer area (%total)	5.35769	9.51865
Hole diameter (m)	0.00476250	0.00476250
Hole pitch (m)	0.0120145	0.0123267
Weir type	Segmental	Segmental
Weir length (m)	0.408074	0.530809

Figure 2.24: Internal design table showing the column layout for the sieve trays.

Design Mode Models and Settings - c:\progra-1\chemse-1\bin\tdesign.def

General | Sieve trays | Bubble-Cap trays | Valve trays | Weirs | Downcomers

Description of design file: TDESIGN.DEF: parameters for tray design

Change factors (-) Large/Medium/Small step: 0.05 | 0.02 | 0.01

Allowed deviation (-) for area for specified FF(DP): 0.05

Maximum allowed:

Column diameter (m) where switch Small/Large column: 1.3716

Liquid fractional entrainment (-): 0.05

Convergence criterion on free area ratio (-): 0.01

Vapour fractional entrainment (-): 0.35

Maximum number of iterations: 30

Froth height (fraction of tray spacing): 0.75

Generate tray parameter output: Yes

Pressure drop (Pa): 11000.0

Generate messages: No

Design method: Continuous redesign

Buttons: Load, Save, Cancel

Figure 2.25: Design mode settings from the standard tdesign.def file.

for a feasible tray design. Further optimization for tray flexibility, minimum pressure drop or maximum mass transfer efficiency is most of the time possible.

Using these tray designs, the nonequilibrium model calculated the mass transfer rates and all the column variables. We may *backcalculate* the Murphree component efficiencies for each stage. Figure 2.26 shows the efficiencies where the mass transfer rates were computed with the AIChE MTC model and mixed vapor and liquid flow models.

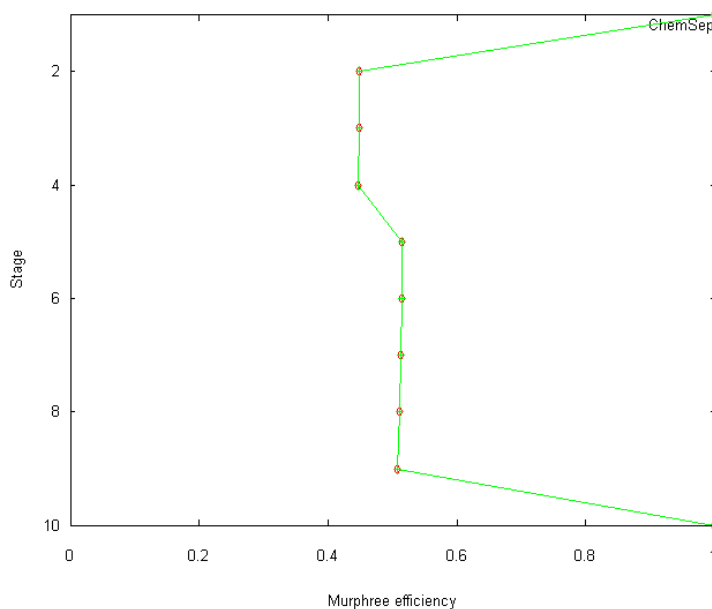


Figure 2.26: Predicted Murphree component stage efficiencies.

We see that the Murphree efficiencies for benzene and toluene are equal at about a value of 0.5. The resulting McCabe-Thiele diagram is therefore quite similar to the one we obtained by using the equilibrium model with fixed efficiencies, Figure ???. Actually, normal efficiencies for a column like this are higher. We need to use the plug flow models to properly simulate this column, bringing the efficiencies to 61-73%.

Sieve trays have a limited operation range and the calculated fraction of flood (FF) and weeping fraction (WF) are also computed during the simulation. These are plotted in the operating limits plot, one of the extra available plots when we do a nonequilibrium simulation. This plot shows the actual operating range by plotting WF and the inverse of FF. When the weeping line is larger than one the trays weep, when the flooding line is smaller than one the trays are flooded.

The pressure profile for the sieve tray column is shown in Figure 2.29 where we see that the pressure drop over the 8 trays is about 0.05 bar or about 6 mbar/tray. Note that the pressure is constant between the condenser and the column. Normally, this is not the case as there is a substantial amount of piping between the top of the column and the condenser, with an associated pressure drop.

For the design of trayed columns the actual vapor loading and the flow parameter value are important.

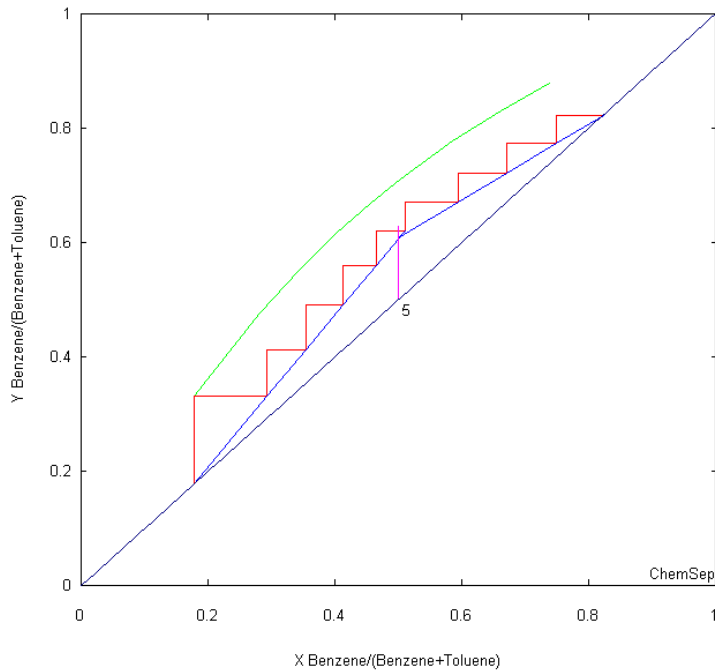


Figure 2.27: McCabe-Thiele diagram for the sieve tray column.

Mass transfer in the column is computed from number of transfer units for both the vapor and liquid phase.

2.3 A Depropanizer

The "depropanizer" is a high pressure column in an oil refinery (Figure 2.32). For our example the feed contains four components: pentane, butane, propane and ethane (in a real column there would be more, but this is enough to show most of the behavior of this column). These components form relatively ideal solutions and the heat effects will be small. The column is to split the feed between butane and propane.

We will use the Peng Robinson equation of state to describe the phase equilibria and use 'Excess' enthalpies. We choose the top temperature at 40°C (to allow cooling with ambient air). With the 'Flash' unit of *ChemSep*, we find that we need a pressure of 22 bar. With a total condenser, 28 sieve trays, a partial reboiler and a boiling point feed on stage 16, combined with a reflux ratio of 2, we obtain the desired separation (This is not necessarily an optimal design, but

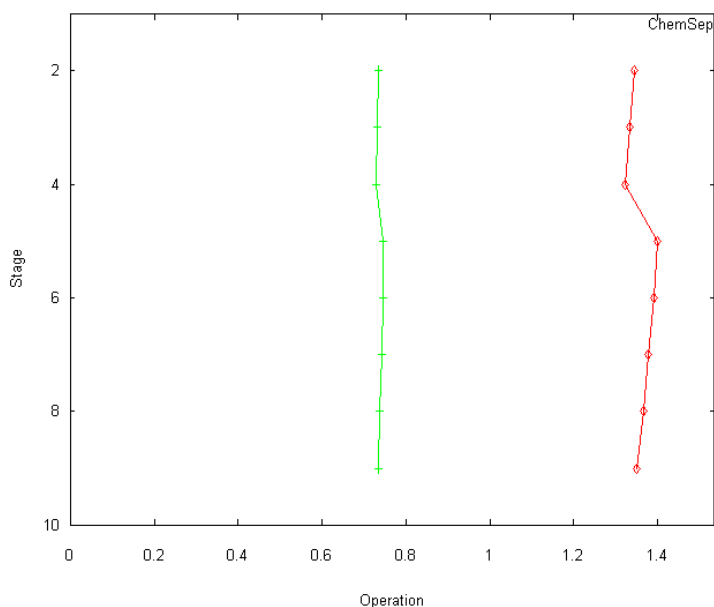


Figure 2.28: Operating limits for the sieve trays.

we may change it later). It takes about two minutes to get the required data into the program. Using the 'design mode' the program sizes the column and we need only specify the type of internal used, in this case sieve trays with a tray spacing of 0.5 m . We specify the design parameters at 60% fraction of flood and 50% fraction of weeping to obtain a more flexible design. We have used the AIChE MTC model using plug flow for the vapor and mixed flow for the liquid. The run converges without difficulty. Output graphs show flow rates (Figure 2.33) which are almost constant in the two sections, except for a sharp increase of the liquid flow at the feed tray.

The flows of the components vary strongly along the column (Figure 2.34), with ethane and propane being concentrated in the top and the other components in the bottom. The two minor components (ethane and pentane) have constant flow rates in the greater part of the two sections, the rates only changing rapidly at the ends of the column. The propane flow generally decreases as we pass down the column, while butane increases. There are however several irregularities. These are mainly because propane is displaced by the minor components in certain parts of the column. Similar remarks can be made on butane. We see that the trays above the feed stage do not provide much separation between butane and propane; this is an indication that we should put the feed on a higher tray.

The transfer rates of the components on the different trays differ wildly (Figure 2.35), showing how complicated such a system is. In the greater part of the column the transfer rates of ethane

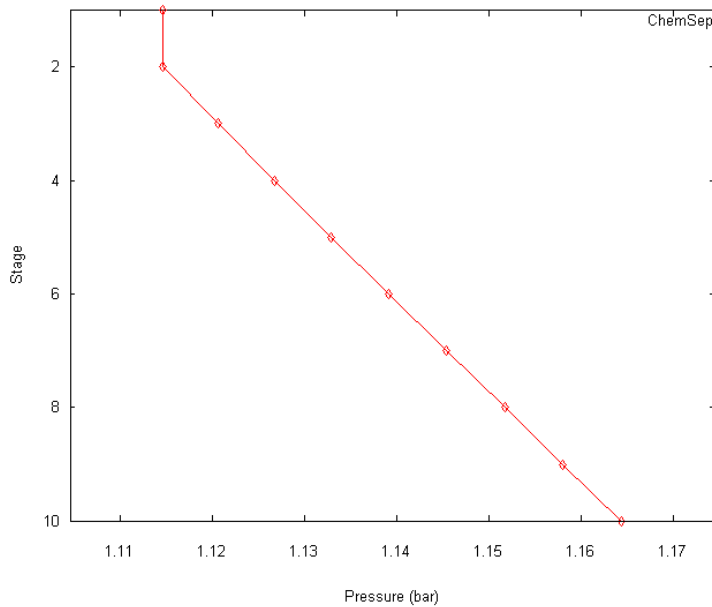


Figure 2.29: Pressure profile for the sieve tray column.

and propane are negative, which means that they are moving into the vapor. The opposite holds for butane and pentane.

We can get out any amount of detailed information on the tray layout in the column, see Figure 2.36). You can immediately find out what are the effects of modifying the design (for example the tray height, or the free areas) by running the program again. Or you can check whether all trays are operating within their hydraulic limits, and how this changes if you modify the design. This is one of the more interesting aspects of nonequilibrium models.

The nonequilibrium model does not use efficiencies. However, from its results, we can calculate the efficiencies of each component on each tray. These are shown in Figure 2.37. Have a close look at these efficiencies. The first thing to be noted is that all four efficiencies are different. The second thing is that they vary wildly along the column: from negative (!) to values far larger than the 100% maximum expected for a tray with a mixed liquid. Remember that these are results for a simple, almost ideal mixture; they are typical of any multicomponent separation. Most of the efficiencies are between 50 and 95% though and those of the two key components propane and n-butane are around 80%.

Our experience is, that many engineers have difficulty in believing these results. They know from experience that *binary* efficiencies are fairly constant and well behaved. So we have added one calculation for a similar column, but now with a binary feed of only propane and

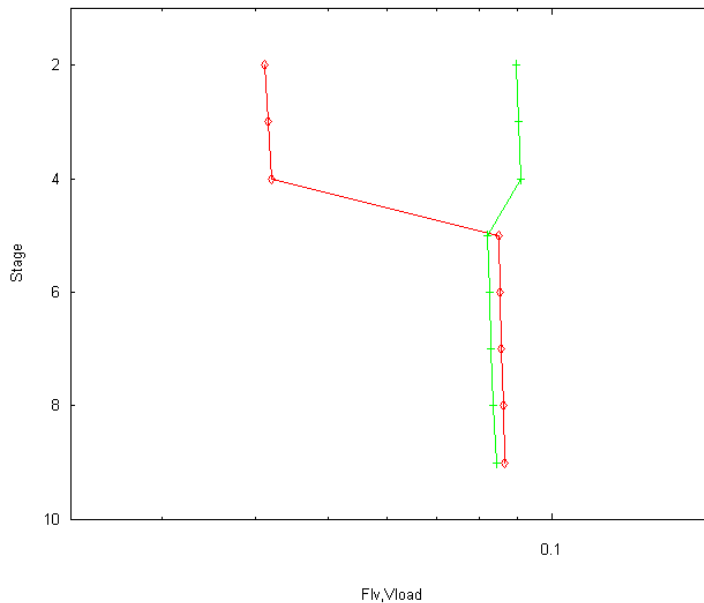


Figure 2.30: Flow parameter and vapor loading of the trays in the column.

butane (Figure 2.38). Here, indeed, we see that the efficiencies of the two components are equal, and that they have a more or less constant value of about 80%. For columns with large flow path lengths (over 1 m), we need to model the liquid flow with the plug flow model. If we do this for our depropanizer we observe that the component efficiencies lie closer to each other and are somewhat higher, on average 90%. The backcalculated overall efficiencies of actual depropanizers is around 80-90%.

From this example we see that experience with binary mixtures does not translate to multicomponent mixtures in a straightforward manner. For a binary, an efficiency may be a useful way of summarizing experience. However in multicomponent mixtures, efficiencies are hardly predictable. They are different for all components and may vary from minus infinity, through zero, up to plus infinity. In our experience, efficiencies in multicomponent separations can be thoroughly confusing.

There are many other aspects of binary distillation, that do not translate well into multicomponent mixtures. Figure 2.39 shows the McCabe-Thiele diagram of the two key components (also provided by *ChemSep*). At first, it looks much like that of a binary mixture. A closer inspection shows that the operating lines are strongly curved. In this example they even run through a small loop near the feed stage. There are other differences between this example and a proper binary. Here, although the feed is at its boiling point, the feed line is not vertical. A close look at the

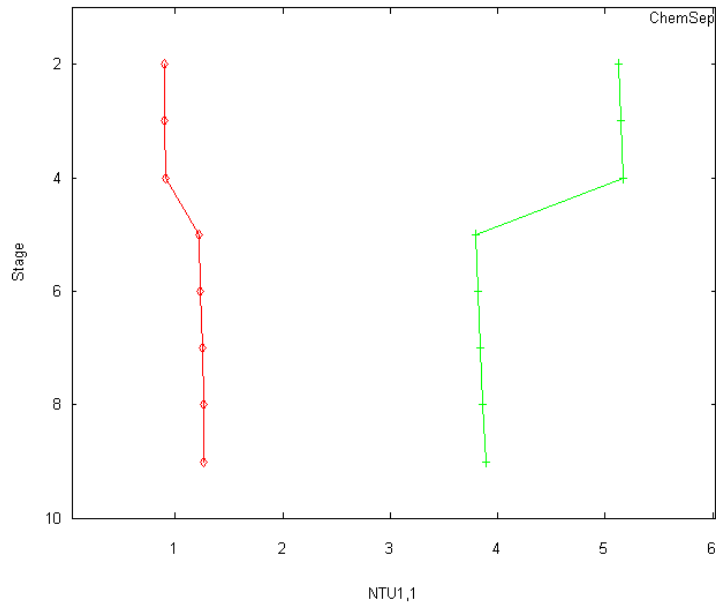


Figure 2.31: Number of Transfer Units for the vapor (red) and liquid (green, the higher values).

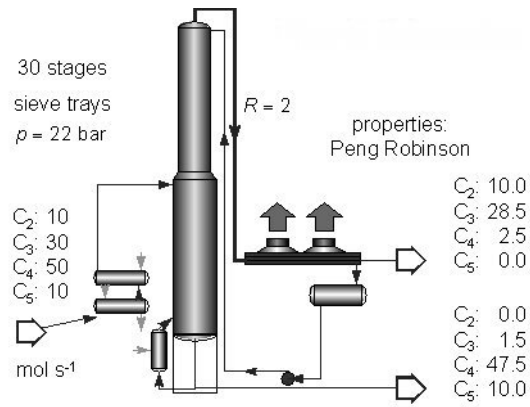


Figure 2.32: A depropanizer.

diagram shows that the 'stairs' do not extend up to the equilibrium line. This is understandable; we are dealing with a nonequilibrium model. However, we have seen that the efficiencies of the

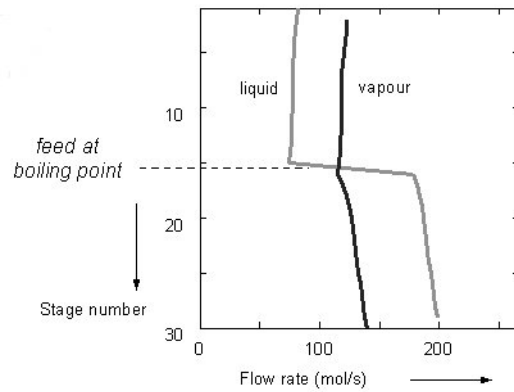


Figure 2.33: Flow rates in the depropanizer.

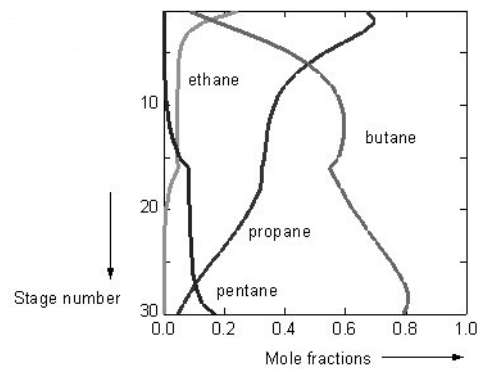


Figure 2.34: Compositions of the components in the depropanizer.

propane and butane are not equal, so the height of each step cannot be a direct measure of the vapor efficiency as it is in a binary mixture.

2.4 A Packed Absorber

Our next example (Figure 2.40) seems to be a simple problem, suitable for an introductory lecture in a separation processes course. We are to absorb acetone (10 mol/s) from a stream of nitrogen (100 mol/s) at 80°C so that the leaving vapor mole fraction of acetone must be

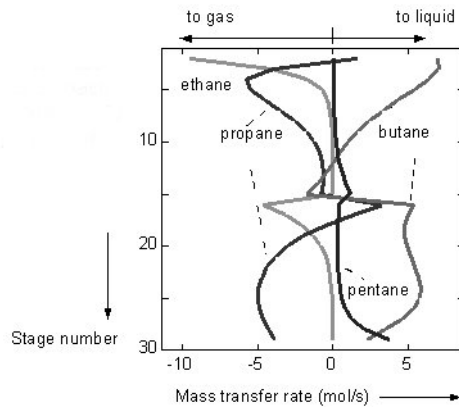


Figure 2.35: Mass transfer rates of the components on each stage.

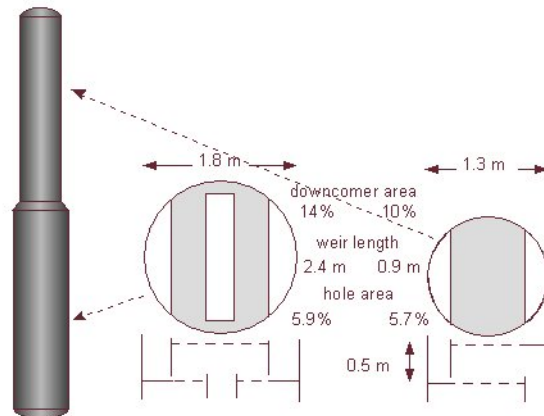


Figure 2.36: The tray layout of the depropanizer.

below 0.1%. The solvent is 880 mol/s water at 30°C , and the 10 m high column is filled with Raschig rings (as it should be in any good, old fashioned exercise).

For this exercise we use the DECHEMA K-model using the NRTL activity coefficient model, Antoine vapor pressures, and the 'Excess' enthalpy model. The NRTL parameters for the acetone-water binary are in the *ChemSep* databank. The parameters for the binaries involving nitrogen must be set to a large value (10^4) so that no nitrogen dissolves (with $\alpha_{ij} = 0.3$). The Onda correlations are used for the mass transfer coefficients. Both phases were taken to be

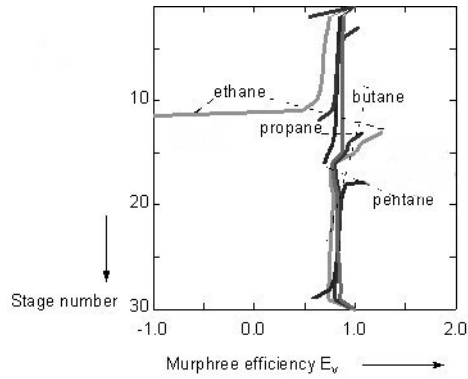


Figure 2.37: Murphree vapor efficiencies of the components in the depropanizer.

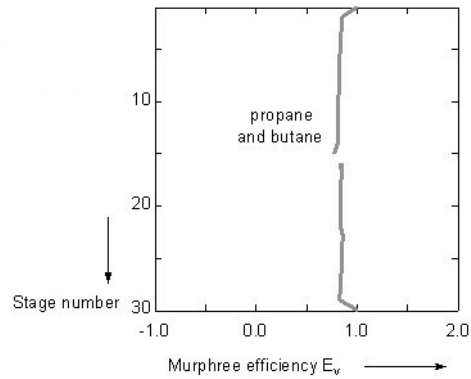


Figure 2.38: With a binary mixture, efficiencies are equal and well behaved.

in plug flow and the column consisted of 30 elements, each of one third of a meter. We obtained the water flowrate by starting at a large value (say ten times the vapor flow) and then reducing it until our specification was met. This resulted into a column of 1.5 m in diameter (at 75% fraction of flood).

Figure 2.41 shows the mass transfer rates in the different sections. We see that acetone absorbs throughout the column and that nitrogen (which is hardly soluble in water) absorbs so little that this is not visible in the plot. The most spectacular component is *water*. Water condenses in the upper part of the column, but evaporates with high rates in the bottom part of the column. This means that there is a large internal water recycle in the column, which carries a substantial amount of heat. All this shows up in the temperature profile (Figure 2.42). We see notable

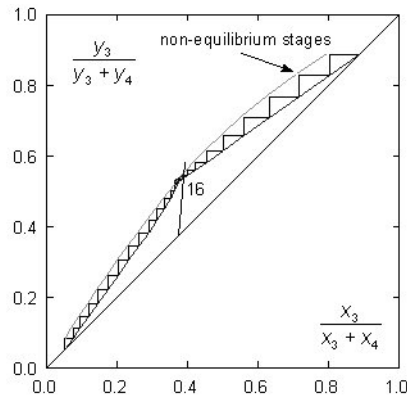


Figure 2.39: The pseudo-binary McCabe-Thiele diagram of the depropanizer.

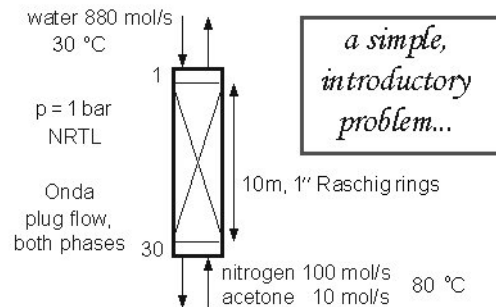


Figure 2.40: The Acetone absorber.

differences in temperature between gas and liquid, as is to be expected. We could have also plotted the interface temperature but it is so close to the liquid temperature that it would be hard to distinguish the two. Actually, the interface temperature is higher than the bulk liquid temperature in the top of the column, but lower in the bottom. This means that the direction of heat transfer changes (at just about the position of the temperature bulge). The situation is obviously far more complicated than the simple textbook problem that we had in mind!

The nonequilibrium model does not use HETP's. However, from its results, we can calculate the HETP's of each component in each element, these are shown in Figure 2.43. The first thing to be noted is that all three HETP's are different. The second is, that they vary wildly along the column. Nitrogen has an extreme value that is negative (!) and water an extremely large value, +78 m, in the top of the column (you may argue that you are not interested in the HETP of nitrogen, but the point is that an HETP can be negative). Even the HETP of acetone varies from

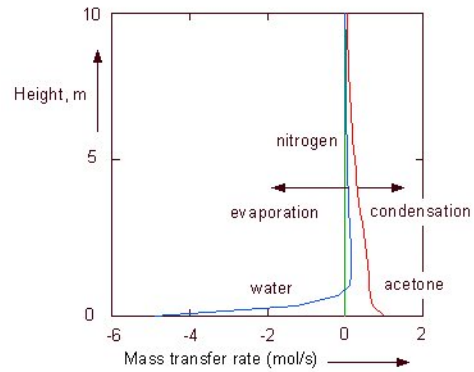


Figure 2.41: Mass transfer rates of the components in the absorber.

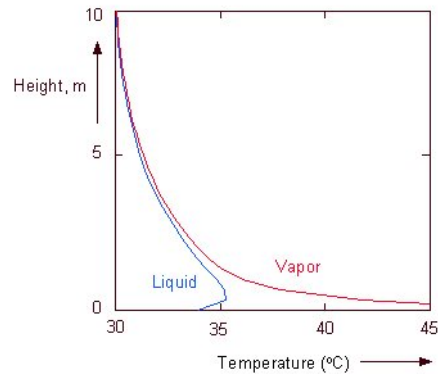


Figure 2.42: Temperature profiles in the absorber.

0.6 to 0.8 m. Remember that these are results for what you thought was a simple introductory example.

From this example we see that experience with binary mixtures does not translate to multicomponent mixtures in a straightforward manner. For a binary, an HETP may be a useful way of summarizing experience. However in multicomponent mixtures, HETP's are different for all components and may vary from minus infinity, through zero, up to plus infinity. In our experience, HETP's in multicomponent separations can be thoroughly confusing.

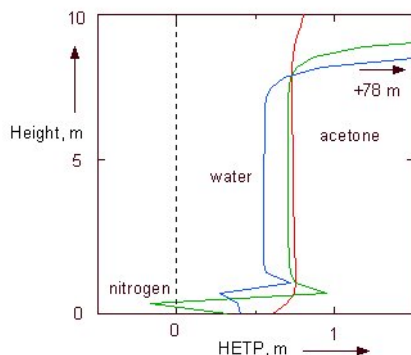


Figure 2.43: HETP's of the components in the absorber.

2.5 Extractive Distillation

The last example considers the simulation of a strongly nonideal mixture with varying internal flowratios. Methyl-cyclohexane (MCH) and toluene have close boiling points; they cannot be separated by conventional distillation. However, they do differ in polarity, MCH being the least polar of the two. We separate them by adding a large excess of a polar, nonvolatile solvent (here phenol) to the liquid. (Figure 2.44) shows this extractive distillation column. Phenol is added on a few trays below the top of the column. The activity of MCH in phenol is larger than that of toluene, so it becomes the most volatile and ends up in the top of the column.

In this problem the UNIFAC model is used to describe the (strongly nonideal) vapor liquid equilibria. As column internals we use bubble cap trays and the AIChE method for the mass transfer coefficients. Also here, we need to start with an excess of solvent to obtain convergence after which the amount of solvent is reduced until the profiles shown in Figure 2.45 were obtained. This is not an optimal design, but it can be improved. Large nonidealities do not pose any exceptional problems to the nonequilibrium model. For this column the tray efficiencies are all over the place.

2.6 Exercises

2.6.1 Separating Four n-Alkanes in One Column

This example came from Henley and Seader (exercise 15.2, 1981) and involves the separation of *four* components into products of at least 96% purity in one column, using multiple sidestreams.

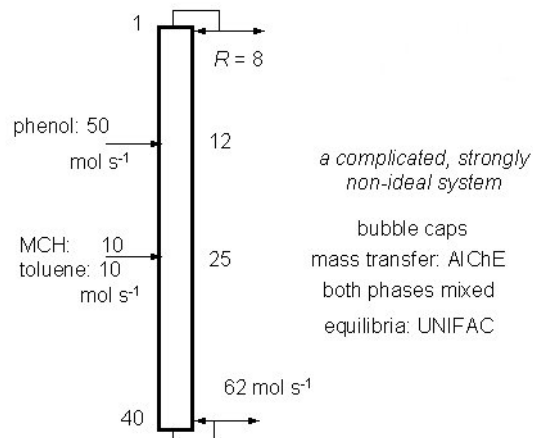


Figure 2.44: Extractive distillation of methycyclohexane (MCH) and toluene using phenol as solvent.

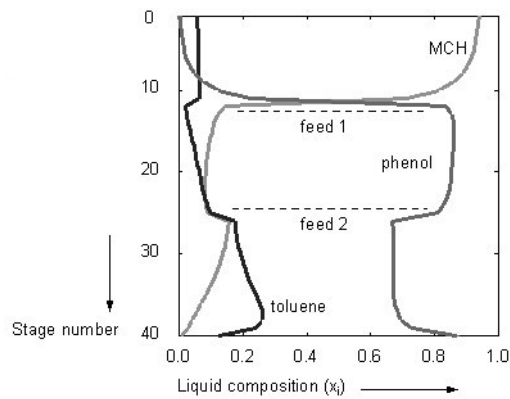


Figure 2.45: Compositions of the components in the extractive distillation.

A feed of n-C4 (14.08 *lbmol/h*), n-C5 (19.53), n-C6 (24.78), and n-C8 (39.94 *lbmol/h*) enters a column at 25 *psia* and 150 °F. The columns condenser operates at 20 *psia* and we let the nonequilibrium model estimate the pressures in the column. Use the Peng-Robinson EOS for the thermodynamics and the complete enthalpy model. Design a column configuration operating at 75% of flood using Sieve trays. How do you specify the column? Why? What happens with the component efficiencies? Could this column be modelled with an overall stage efficiency? (Hint: sidestreams do not need to be only liquid streams).

Do the sections need to be of different design or could one general column layout suffice? Try using different internals in various sections: switch to valve trays, 1 inch metal Pall rings, and Sulzer BX structured packing (you can do this within *one* simulation!). Compare and comment on the operating regions and the resulting column diameters you find for these internals. Are they realistic? Check the influence of using plug flow models and try out different MTC and pressure drop models. Put your results in a table to make comparisons.

2.6.2 Azeotropic Distillation

In this example we simulate the separation of benzene (boiling at 80.2°C) and cyclohexane (80.8°C), which form an azeotrope which boils at 77.4°C. Acetone (56.4°C) is used as entrainer. Acetone does not form an azeotrope with benzene, but does so with cyclohexane that boils at 53.1°C (thus, a minimum boiling azeotrope which will go over the top). The acetone and the pure cyclohexane are recovered from this azeotrope with a water wash, followed by a simple distillation that produces water and acetone for recycling. Figure 2.46 shows the whole process (see also Smith, 1963, p. 405).

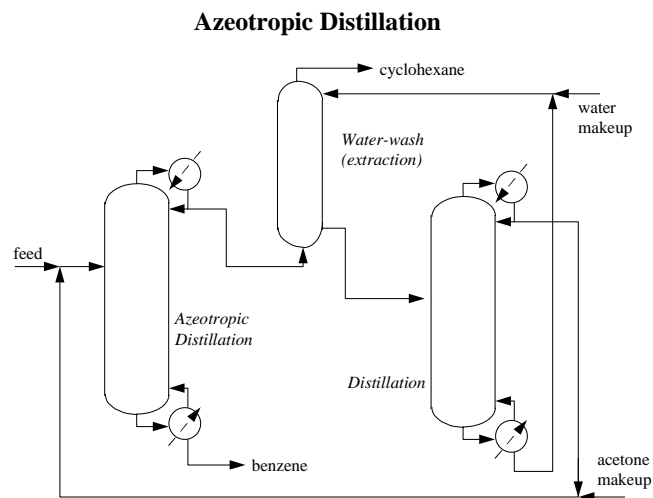


Figure 2.46: Azeotropic distillation of benzene and cyclohexane with acetone as entrainer, and a water wash for entrainer recovery.

Here we will only look at the azeotropic column. Since the acetone is recovered by means of a water wash there will be some water (3 mol%) and cyclohexane (2 mol%) in the recycle feed to the azeotropic column as well. The recycled acetone is fed to the same stage as the equimolar feed of 10 mol/s cyclohexane and 10 mol/s benzene (simulations with the nonequilibrium

require actual feedflows as actual tray performances must be calculated). The flowrate of the acetone entrainer is varied to obtain the optimum separation, in this case it is set at 35 mol/s . The column is simulated with 38 sieve trays, a total condenser and partial reboiler, with the feed entering on the 14th tray (counting from the top). To model the thermodynamic equilibrium the DECHEMA – UNIFAC model was used and for the mass transfer the AIChE model. No column layout parameters were specified, these were calculated by the program for a column to operate at 75 % of flooding. The reflux ratio was set at 4 and the bottoms flowrate was specified to be 10 mol/s .

A 99.5 mol% pure benzene bottom product is obtained and a top product containing mainly cyclohexane and acetone with a benzene content that can result in cyclohexane purity of 99 mol% or better. The column was split into two sections, one above the combined feed tray and the other one from the feed tray to the last tray in the column. Answer the following questions:

1. Look at the resulting tray layouts. Do we need different designs in the two sections?
2. Check at the Benzene concentration profile. Explain what happens in the top section of the column
3. Plot the component efficiencies. Explain the behavior of the component efficiency of Acetone. Could this column also be simulated using the equilibrium model with a constant efficiency? Explain.
4. Inspect the McCabe-Thiele plot, can you improve the column configuration? Try!
5. A second steady-state solution was found (with the same specifications) where the bottom product contains a lot of cyclohexane and the acetone concentrations in the middle of the column are much lower. Can you find both the two steady states?

2.7 Model Uncertainties

The nonequilibrium model improves our description of separation processes. This does not mean that there are no problems left to be solved in distillation. Complicated mathematical models do not solve problems; they may only improve your solution. The results are only as good as the important parts of the model. During development of nonequilibrium models it has become clear that our knowledge of the details of what happens inside a column remains incomplete. There are several areas where current models and correlations (at least those in the open literature) can be improved:

- correlations for interfacial areas,
- correlations for mass and heat transfer coefficients,
- minimum load correlations, and
- liquid flow models and the description of maldistribution in packed columns.

As distillation research focuses on these areas the nonequilibrium model will benefit from the model improvements and users will obtain more accurate simulation results.

References

E.J. Henley, J.D. Seader, *Equilibrium Stage Separation Operations in Chemical Engineering*, Wiley, New York (1981).

B.D. Smith, *Design of Equilibrium Staged Processes*, McGraw-Hill, New York (1963)

R. Taylor and R. Krishna, *Multicomponent Mass Transfer*, John Wiley and Sons, New York 1993.

J.A. Wesselingh and R. Krishna, *Mass Transfer*, Ellis Horwood, Chichester 1990.

Chapter 3

CAPE-OPEN Flowsheet Simulations with *ChemSep*

ChemSep is fully CAPE-OPEN compliant and can be used as a "plug-in" in all compliant flow-sheeting packages. This chapter illustrates the use of *ChemSep* column simulations in the COCO Simulator (www.cocosimulator.org) that is available for free. The chapter starts with a small introduction of COCO and then discusses examples ranging from simple separation trains up to the simulation of complete chemical plants.

3.1 Using the COCO Simulator

When you open the COCO simulator via the start menu you'll find an empty flowsheet loaded in the GUI, called COFE. The GUI has the usual menu entries (File, Edit, Window, Help) of multiple document application in MS windows. The other items are:

- Insert : to add a new stream, unit operation, or object (line, text, bitmap) to the flowsheet
- Flowsheet : to configure, validate, and solve a flowsheet, edit a unit operation or to make reports
 - Plot : to generate physical property plots or plot results of a specific unit operation
- View : to set aspects of the graphical representation of the flowsheet, i.e. the zoom level, grids, stream and unit labels, etc.

The COCO window is split over a menu and toolbar, an area for the flowsheet graphs, and a log screen displayed on the bottom over the status line. Multiple flowsheets can be loaded simultaneously and these are completely separate identities. COCO comes with its own packages

for thermodynamics and physical properties, called TEA (derived from the *ChemSep* code and Pure Component Data PCD files); standard unit operations, COUSCOUS (which include mixers, splitters, heat exchangers, pumps, etc.); and reactions, CORN. Let us now start to setup a simple Benzene/Toluene/p-Xylene (BTX) separation train in COCO using *ChemSep* columns.

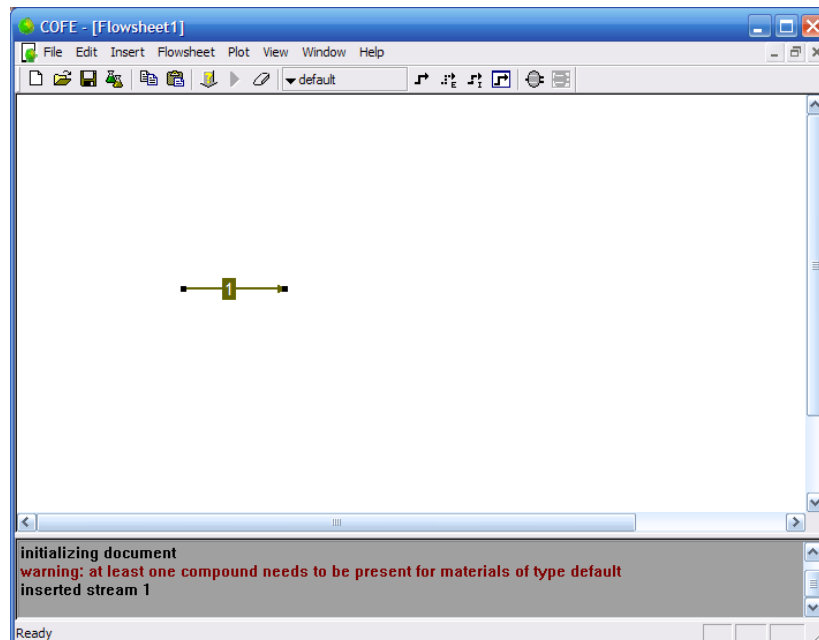


Figure 3.1: The COCO Simulator with a new stream.

Our goal is to separate an equimolar stream of BTX of 80 tonnes/day (25 degrees C, 1 atm) into 99% pure Benzene, Toluene, and p-Xylene. Start COCO and maximize the default flowsheet it opens with. We then insert a stream by selecting the menu "Insert-Stream" (or **Ctrl-I**) and clicking on the flowsheet area once, then dragging the cursor a bit to the right and clicking again, see Figure 3.1.

Note that COCO issued a warning, as can be seen in the flowsheet log: the stream we added does not have any material assigned to it. So before we go on we must configure the materials as well as thermodynamic and physical property model that will be used to describe the materials in the flowsheet. Press **Ctrl-K** or go to the Flowsheet menu and select the Configure option, this will bring up the flowsheet configuration window, Figure 3.2.

You'll see that there are many parts of the flowsheet to configure but we just need to fill in two: Compounds and Property packs. Select the Property packs tab (default for a new flowsheet) and click on the **Add** button that appears to the right. This brings up a window where you can select the type and version of property packages that are available on the current machine. If you

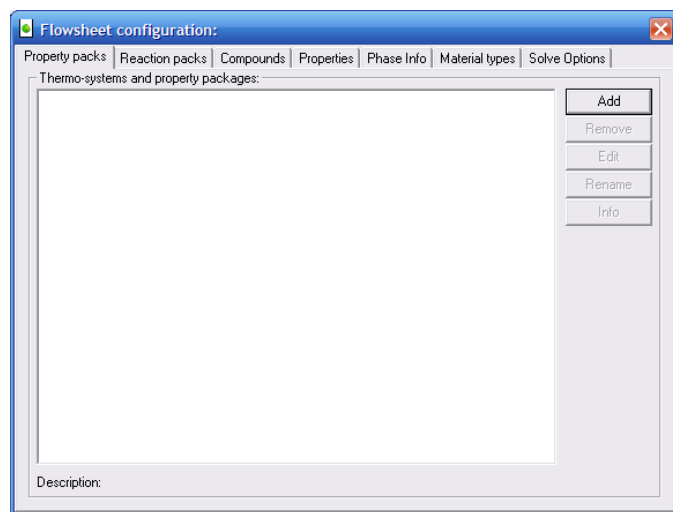


Figure 3.2: Empty configuration window.

just installed COCO (and no other CAPE-OPEN compliant software), you only get a selection between the two different versions of the package that comes along with COCO, namely TEA, Figure 3.3.

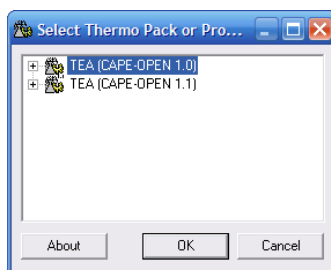


Figure 3.3: Property package selection window.

If you installed ASPEN PLUS, PRO/II, or other CAPE-OPEN software more packages will be listed. This is very useful when flowsheets are re-used: once a property package is defined, it can be used in other flowsheets, even in flowsheets of other software packages as long as the CAPE-OPEN standards are followed. For the present purposes we will use TEA as it comes with COCO. Both versions listed will work (for more information on versions of the thermo/properties standard consult the CAPE OPEN web site, www.colan.org). For now click **OK** to chose "TEA (CAPE-OPEN 1.0)" which will bring up another window, Figure 3.4.

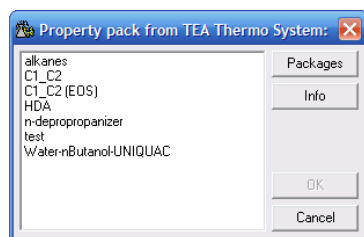


Figure 3.4: Property package selection window.

The left part of this window shows a list of CAPE OPEN property packages that were created earlier. None of these packages are useful to us, so we must first create a new package. Click on the **Edit pack** button that appears towards the upper right of this window. This will bring up the TEA window with previously defined packages, Figure 3.5.

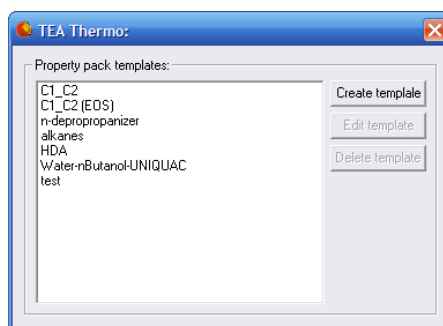


Figure 3.5: TEA Thermo window.

We want to define a new template, so click on **Create template** button, on the upper right. Finally, here we are able to actually define compounds and property models to compute physical equilibria. Type in something that describes the compounds and models for the name of the property pack template, Figure 3.6.

Use a name that helps you to identify the template for later use and also for use by others. Be aware that templates are machine specific, i.e. they are stored in a general area where any user can access all templates defined. This is a strong point for sharing information but could be a risk for confidential projects. To add actual compounds select the compounds tab. The list will be empty. Click on the **Add** button to select components for the new property template, Figure 3.7. COCO shows the compounds with their overall formula, mole weight, and CAS numbers (so you can differentiate between isomers). You can pick compounds from this list by double clicking or you can search all the components with a benzene ring by typing "benzene" in the filter edit box on the bottom. Besides Benzene you also need to select "Toluene" and

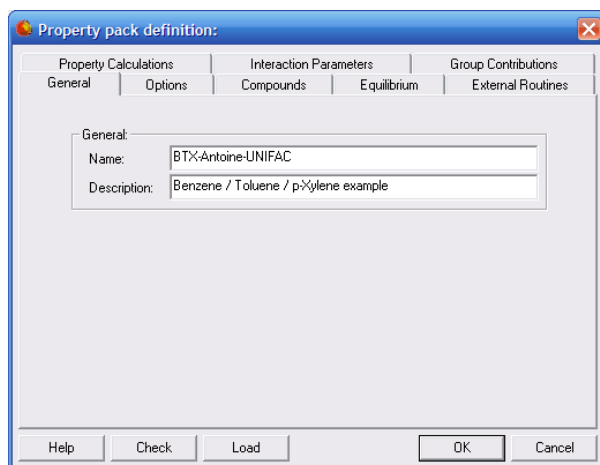


Figure 3.6: Define template for the BTX property package.

”p-Xylene” as compounds Figure 3.8. Note that these are read from a *ChemSep* PCD file called ”DefaultComponentLib.pcd” that is part of the COCO distribution.

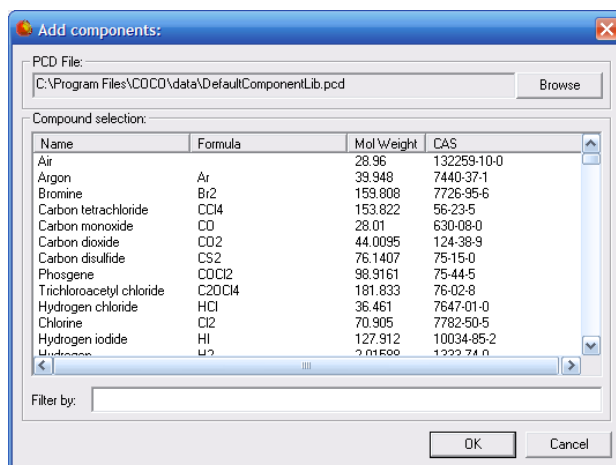


Figure 3.7: Component selection.

To complete the BTX template we need to choose the models that will be used to estimate thermodynamic and physical properties. Click on the **Property Calculations** tab. COCO shows there has not been any model selection made yet, as indicated by the ”N/A” for the three properties that are present by default, Figure 3.9. Note that TEA allows us to exactly define specific

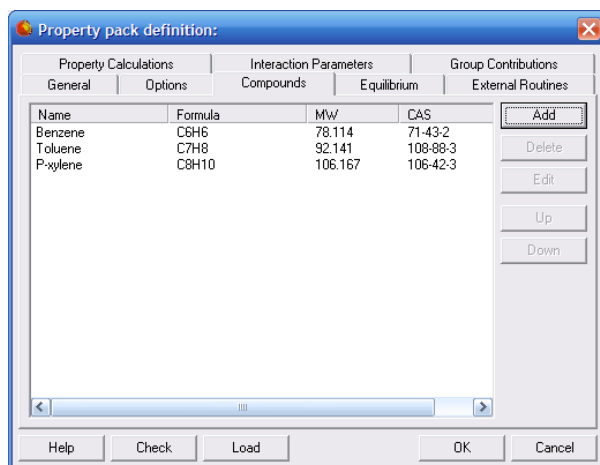


Figure 3.8: Selected BTX compounds.

sets of properties.

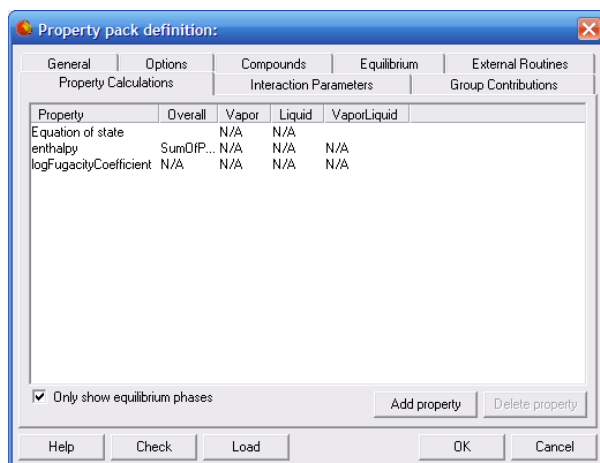


Figure 3.9: Empty model selection.

For a *ChemSep* simulation we will require property models for the (log) fugacity coefficients, enthalpy, entropy, heat capacity, density, viscosity, thermal conductivity, and surface tension. The fugacity coefficients may be computed with a variety of other models. Here we will want to use an activity coefficient model with a vapor pressure model. Use the **Add property** button to add the "Activity coefficient" property to the template, Figure 3.10.

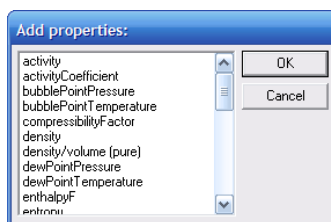


Figure 3.10: Property selection.

Note that adding the activity coefficient model automatically adds the vapor pressure model. Similarly, if one select "surface tension" COCO also adds a surface tension property for pure compounds. Go ahead and add all the required properties mentioned above that TEA must support in order to be able to solve a column in *ChemSep*, Figure 3.11.

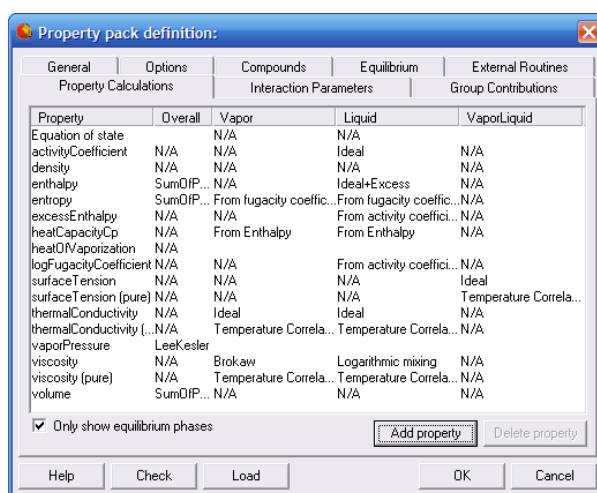


Figure 3.11: The required set of properties for *ChemSep* (activity coefficient model).

Note that in Figure 3.11 we see COCO has preselected certain models for the viscosity for example. It also assigned the "LeeKessler" method for the vapor pressure. However, we want to use the Antoine model for the vapor pressures. To change this click the selected model to get a list of the available methods, Figure 3.12.

In order to use the property package we must make a complete selection of methods for each of the properties we selected to be part of the template. For the activity coefficients we will make use of the UNIFAC-VL method which is a predictive method that does not require further input. Use Figure 3.13 to fully define the property package we will use for the BTX example. Note

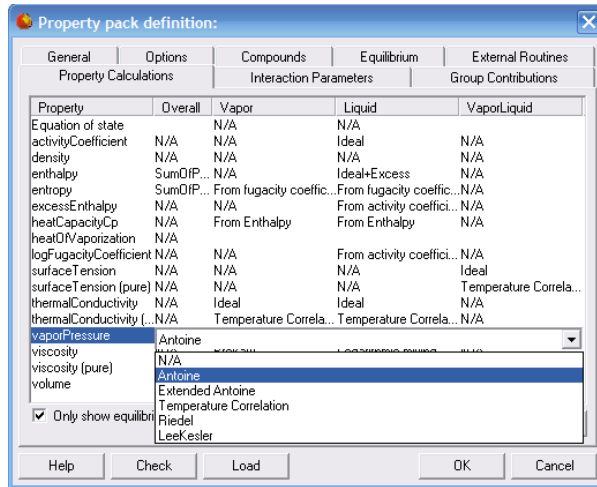


Figure 3.12: Vapor pressure methods.

that some of the properties still have "N/A" as they are not applicable to the phase in question. Note that by default TEA is making a template with vapour and liquid phases. If you would want to have other phases present, the phase definitions need to be adapted. We won't need that for this example.

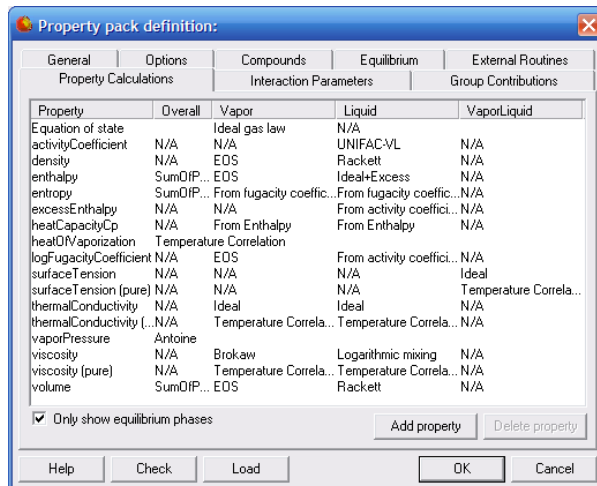


Figure 3.13: Property methods for the BTX example (UNIFAC+Antoine).

Once we finished the property template for the BTX example, click the **OK** button and we will see the new template is added to the available set of templates. Close the window displaying these templates and the window displaying the available property packs is also updated and includes our new BTX package. In other words, making a template doesn't necessarily add the template to the Flowsheet. Thus we must select the BTX package and click the **OK** button to actually add this package to the flowsheet. COCO will also confirm whether this package is to be made the default template. Click **Yes**, Figure 3.14 and close the configuration window.

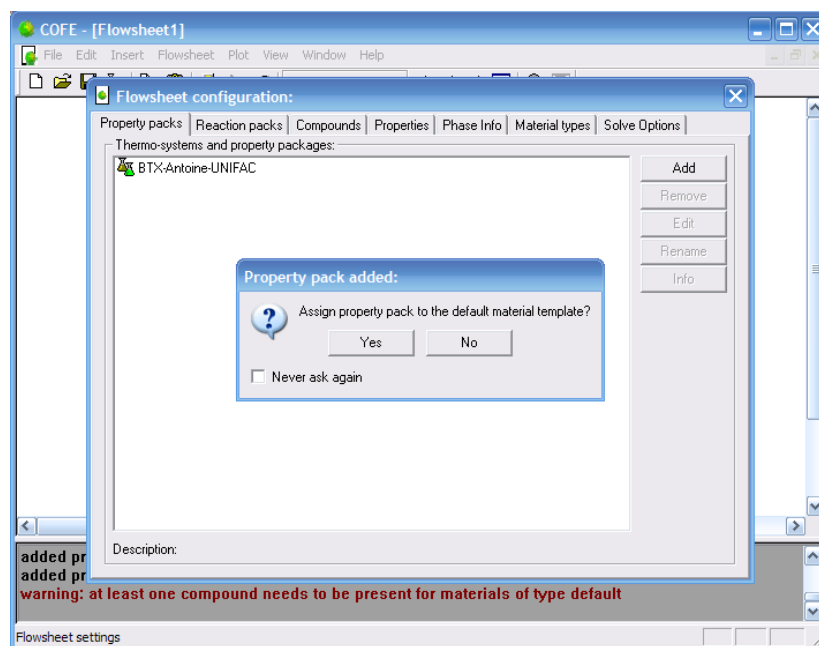


Figure 3.14: Selection of the BTX property package.

When we now double click on the stream "1" we previously added to the flowsheet, we see that COCO indeed displays a definition with the three compounds we defined to be part of the BTX property package.

We now want to define our feed. We would like to do this in convenient units, therefore click on the units string for the pressure to display the available units of measure for pressure. Switch from Pascal to Atmosphere (atm), Figure 3.15.

Similarly, define the temperature to degrees Celcius and the flow to tonnes/day and then finish the feed definition for our BTX example, Figure 3.16. Once the stream is fully defined COCO will automatically compute the stream and the properties that were defined in the property package. Note that it will just show the properties that are turned on for display in the Properties tab of the flowsheet configuration. Take some time to browse through all the properties. Note

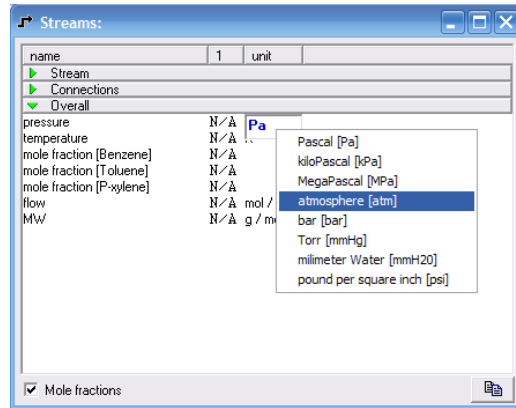


Figure 3.15: Define units of measure for pressure.

that at 1 atmosphere and 25 Celcius the feed is a liquid. You can select some more convenient units for the density for example, use kg/m^3 . When done, close the window. Notice the color of the feed has changed from brown to black. This means the feed is now fully defined and you can solve the flowsheet (the menu button is now enabled).

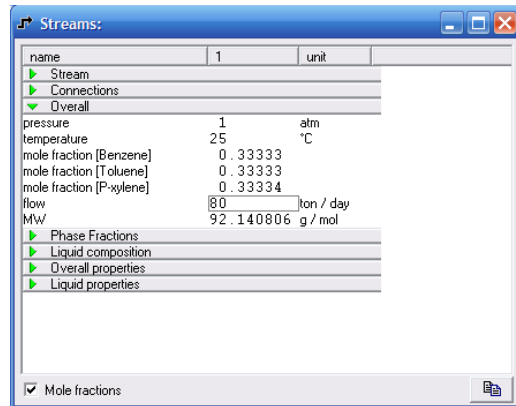


Figure 3.16: Define BTX feed.

Solve the flowsheet by pressing **F5** and notice how the feed changes color from black to green. All units that are solved are colored green. An unit or stream for which solving resulted in an error turns red. If the flowsheet does not converge, all items in it will be black, as in specification complete. A non-defined stream is colored brown.

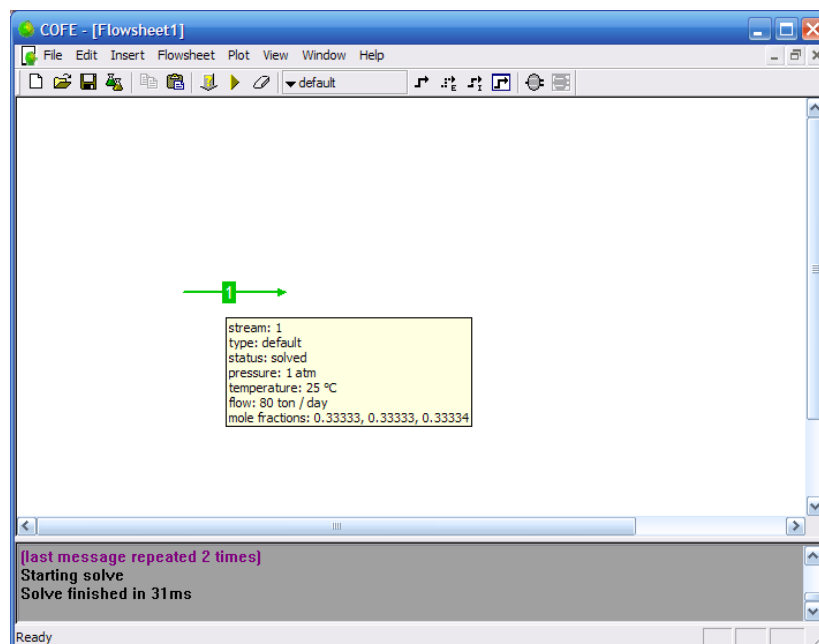


Figure 3.17: Solved flowsheet with BTX feed.

Once the flowsheet is solved COCO will display the stream status when you hover with the mouse over it, see Figure 3.17. We now add a distillation column by pressing **Ctrl-U** (or select **Unit Operation** from the **Insert** menu). You can select from a list of registered CAPE-OPEN unit operations available on the current machine, see Figure 3.18. You can select a *ChemSep* distillation column by opening the list of separator unit operations and selecting ChemSepUO.

You will get a confirmation that the new unit operation is added without any ports hooked up to it. Then a default distillation icon will be shown on the flowsheet. It's displayed in red and you can move it to any place that is safe on the flowsheet. After you placed the column on the flowsheet with a click, the color changes to brown, indicating its specification is incomplete. To make specifications for this distillation column, double click it. That opens the unit operation (UO) in the generic editor of COCO, Figure 3.18. To directly open the GUI of the unit operation itself instead of COCO's generic editor, hold the Alt-key while double clicking the column.

This is a generic editor and only shows the variables that the operation exports. These will be stored in a persistently manner when you save and open flowsheets. In our column case these include flags as to which type of flashes to use on feeds and products, or how to log CAPE-OPEN calls. To specify the column we want to launch the GUI of *ChemSep*. To do this, click the **Show GUI** button. Currently, *ChemSep* uses a wrapper for CAPE-OPEN, called ChemSepUO.DLL. It is the GUI of the wrapper which is launched by COCO, see Figure 3.19.

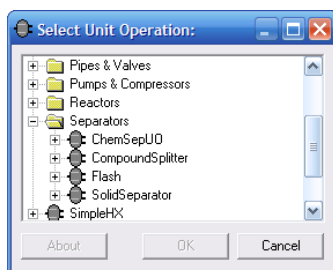


Figure 3.18: Inserting a new unit operation.

The options available in this wrapper are explained later. All you need to do for now is to click the **ChemSep Interface** button to launch the *ChemSep* GUI in CAPE-OPEN modus.

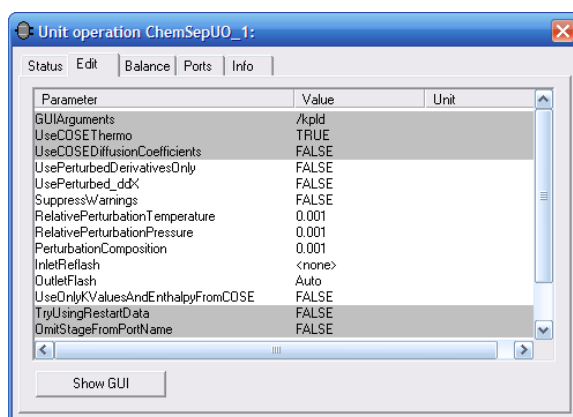


Figure 3.19: Editing a Unit Operation (UO) in COCO.

When the *ChemSep* GUI opens in CAPE-OPEN mode several features are hidden or disabled. We continue with setting up our BTX column as follows:

- Select metric units for this problem
- Use a simple distillation column configuration with 10 stages and a feed to stage 5
- Specify a constant pressure of 1 atm in the column
- There are no column heat duty or heaters/coolers and a stage efficiency of unity
- Specify a Reflux Ratio (RR) of two and the fraction of feed flow recovered in the bottom equal to the expected components; i.e. $2/3$

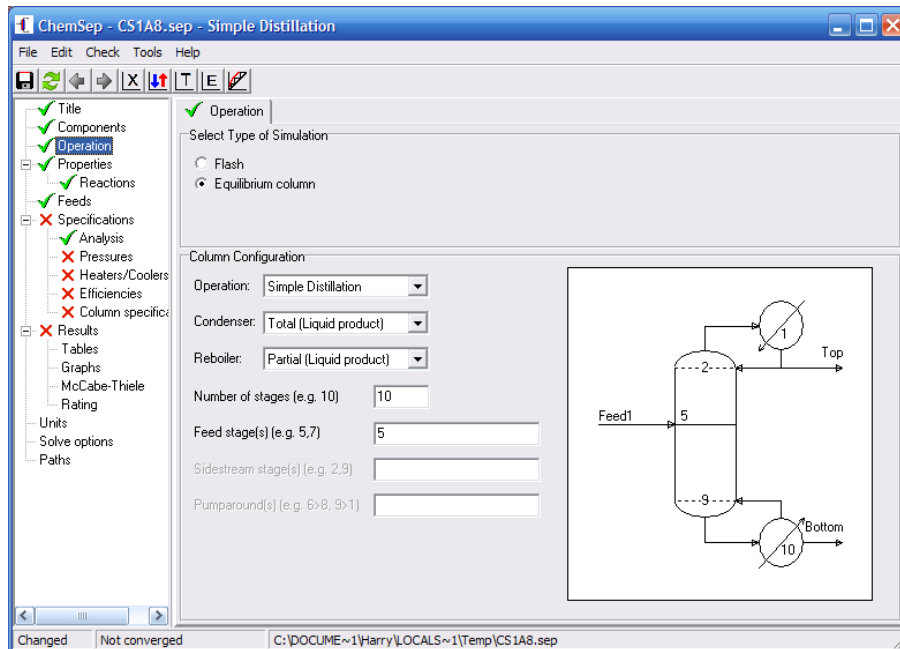


Figure 3.20: Setting simple distillation column configuration.

Once the problem is setup, exit the GUI and save the column specifications. As we now have defined that the column has a feed and two products, we can also start connecting them in the flowsheet. Add our feed to the column and add top and bottom streams. You'll notice that the color of the column changes from brown to black when you do this. COCO noticed that the problem is specified and enabled also the solve button. You can now solve the BTX column. Do so by clicking the solve button.

This makes COCO call the wrapper which in turn calls *ChemSep*'s simulators. During the time that the column simulator actually solves the column, the icon is displayed with a orange background. Once the BTX column is solved, it is displayed as green. If there are warnings there will be a warning label attached where you can click on to see what was wrong. If solving failed, the column will be shown in red.

After the column has been solved, go an edit it again by double clicking while holding the **Alt**-key. Note that this skips COCO's editor and jumps straight to the GUI of the *ChemSep* wrapper. The simulator's output is shown in the status window, Figure 3.22, in green when the column is solved.

Now the column is solved you can click the status tab and look at the results report. This will give a summary of all the tabular output that can be seen in the *ChemSep* GUI. The same

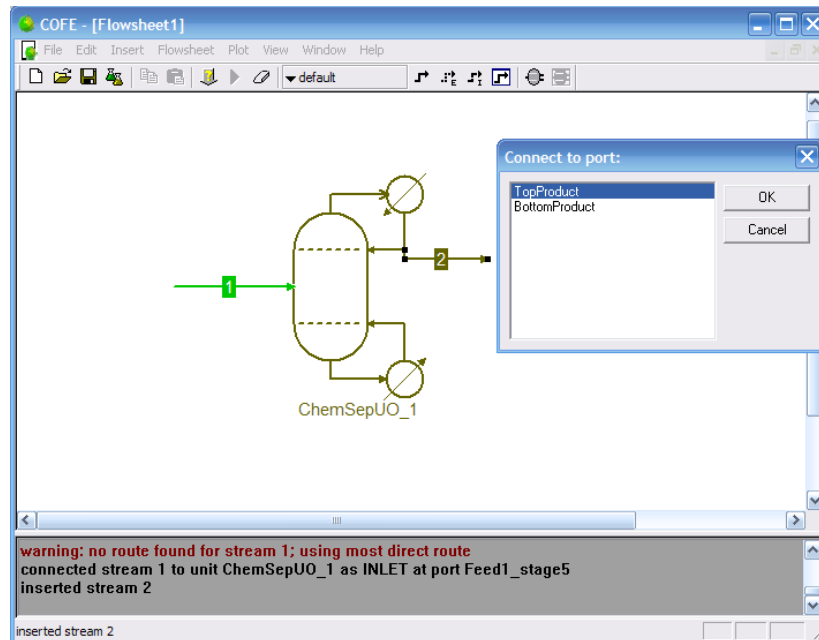


Figure 3.21: Connecting top and bottom product streams.

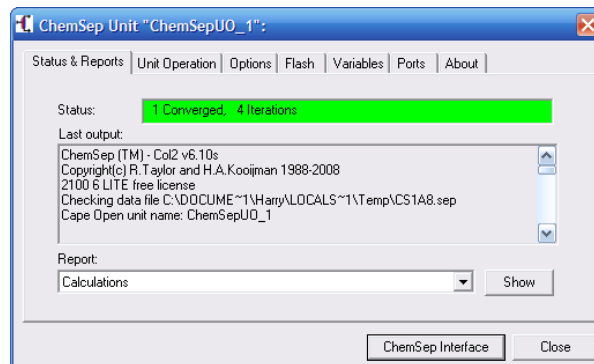


Figure 3.22: Solved unit operations with last output.

information can be inserted as the report of a unit operation on the flowsheet. Similarly, you can add stream tables for all the streams in a converged flowsheet.

Add a stream table now by selecting **Insert-Stream report** which shows the mole fractions of the three components in our BTX example. We can even edit the table by double clicking it and adding the density as property to report to the stream table, see Figure 3.23.

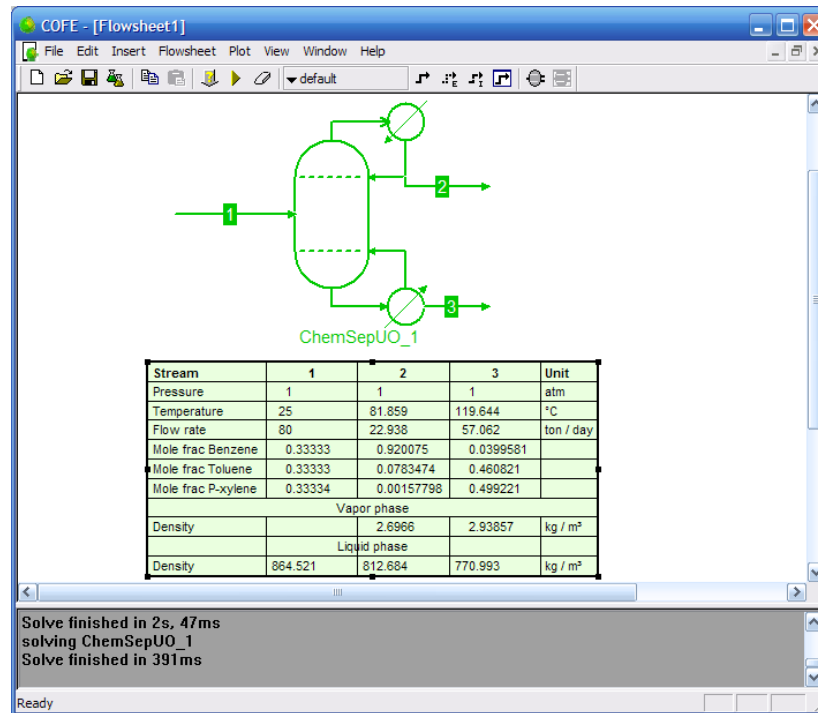


Figure 3.23: Flowsheet with densities.

Notice that the separation isn't good enough to produce 99% pure Benzene as top product. To reach this purity we can add more stages in the column, here 20 stages (with the feed in the middle) suffices to reach the required purity. Of course, we could have also increased the reflux ratio we specified. These are process considerations, of course, since more stages increases the height of the column whereas more reflux increases its diameter and energy usage. To determine these aspects you must rate the column (e.g. by means of the **Rating** tab in the ChemSep GUI).

Now we have successfully separated Benzene from the ternary BTX, we still need to separate the bottoms of this column into pure Toluene and p-Xylene. To do so we must add a second column which will take as a feed the bottoms of the first column. Go through the same steps as above but while specifying the second column use more stages (30) and a higher reflux ratio (3). Connect the bottoms of the first column to this second column, and attach top and bottom product streams. Resolve the flowsheet and add the products to our stream table. The result is shown in Figure 3.24.

Note that in this separation train there was no recycle and thus the flowsheet does not need to iterate to obtain the final solution. When you re-solved the first column you might also notice a significant speed-up. The reason is that COCO remembers the old state of the column and switches the column simulator automatically to a mode where it re-uses the old results as initialization. This generally saves the user a lot of time but sometimes these old results can lead to the convergence to a wrong solution. Therefore, the wrapper has a button to delete them under the Options tab.

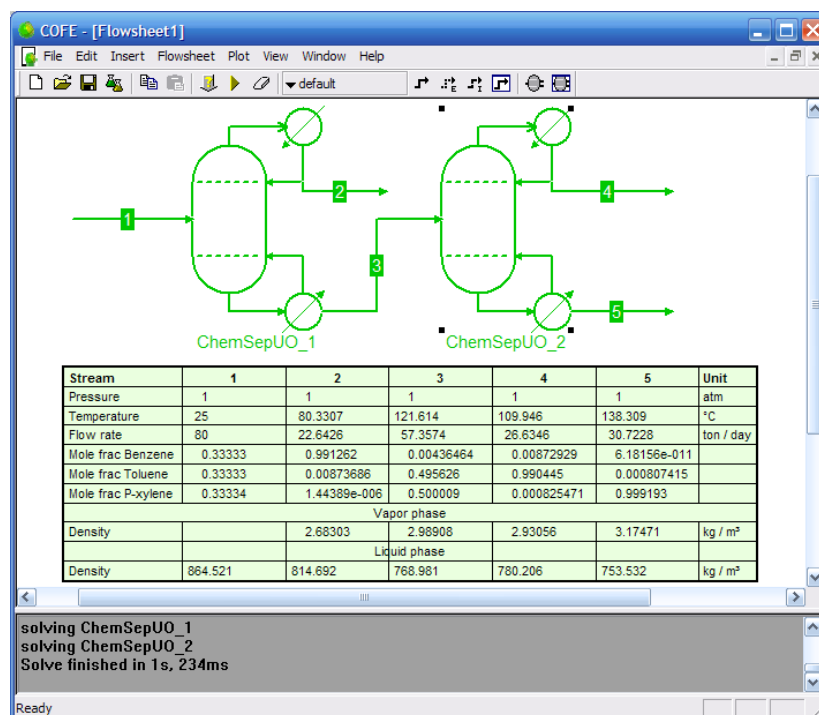


Figure 3.24: Separation train for BTX with two columns.

3.2 Flowsheet Examples

Here we discuss several examples that have been modelled with COCO and *ChemSep*. We will not supply step by step setup as the previous BTX example but instead just discuss the necessary details to specify and simulate the problem. The flowsheets are distributed on the download page *ChemSep*, <http://www.chemsep.org/downloads/>. Note that here we introduce the flowsheets using equilibrium column models. Of course, better descriptions of actual column performance are obtained with the nonequilibrium column model. We show this

in a bit more detail with one particular example where we illustrate that *ChemSep's* integrated design method enables us to use the nonequilibrium column model while still changing the flowsheet. The design method eliminates the need to know the column layout *a priori*: it is continuously adapted while the column model is solved. In most of these examples *ChemSep* solves the columns without needing specific guesses with respect to temperatures or flows: its *automatic initialization* suffices. Note that while iterating to find the solution of the flowsheet, *old results* are used as initialization. This speeds up the convergence when the flowsheet close to being solved (but can sometimes result in a column not being solved).

3.2.1 Extractive Distillation of MCH and Toluene with Phenol (PH)

We need to separate an equimolar mixture of methylcyclohexane (MCH) and Toluene which form an azeotrope together. We can still separate the two components by means of extractive distillation using Phenol as the solvent. The flowsheet is shown in Figure 3.25.

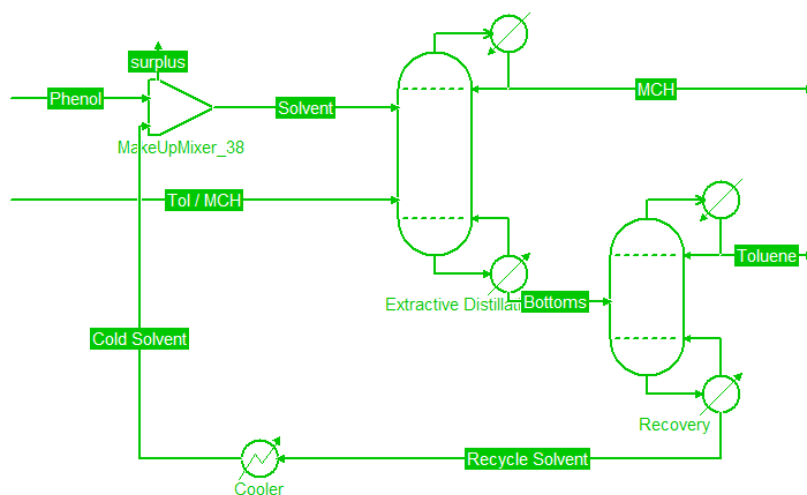


Figure 3.25: Extractive Distillation.

In the first column we introduce the 50/50mole% feed of MCH and Toluene. Above this feed the solvent is introduced into the column on a stage close to the top. The feedstage for this depends on the difference in boiling point between the solvent and that of the light component, here MCH and the feed stage is set to stage 5.

will be seen that the process involves two distillation columns, a heat exchanger, and a make-up stream. Valve trays are used for both the columns using the design mode nonequilibrium model. The phenol recycle is cooled to 100°C. For a high purity of the products the solvent

feed to MCH/toluene feed ratio as well as the reflux ratio needs to be sufficiently high (for the extractive column). We need the make-up unit to regulate the amount of phenol in the feed to the first column.

3.2.2 Distillation with a Heterogeneous Azeotrope (BW)

Water and n-butanol form an azeotrope and cannot be separated by conventional distillation. However, at not too high temperature, the liquid mixture splits into two liquid phases; an aqueous phase with very little butanol and a butanol phase containing a substantial amount of water. We can use a decanter to separate the two liquid phases. A feed that is predominantly water but contaminated with butanol (1 *mole%*) can be separated into two pure products using such a decanter. The flowsheet is shown in Figure ???. The first column produces a bottom product that is water, and the vapor overhead goes to a condenser/cooler and then to the decanter. There we obtain water- and a butanol-rich streams that get recycled to the columns. In a second distillation column we produce butanol as the butanol-rich recycle contains much more butanol than the butanol-water azeotrope. Note that we use flash units to simulate the coolers of the vapor overheads of the columns (which, therefore, do not have any condensers). The liquid-liquid decanter is simulated using a single equilibrium-stage extraction unit. Since we need to estimate the water and butanol-rich recycle flows, it is best to simulate these columns first with a condenser. The solutions so obtained can be used as a basis for estimating the recycle flows.

3.2.3 Distillation of a Pressure Sensitive Azeotrope (MA)

Methanol and acetone form a minimum temperature azeotrope but the composition of this azeotrope is sensitive to the pressure. We can make use of this to separate the two components into pure products by operating two columns at different pressures.

here show an example of the azeotropes of methanol-acetone in two overlapping Txy diagrams at two different pressures using chemsep

This type of azeotropic distillation is rare as the azeotrope composition needs to be sensitive to the pressure in order to obtain a recycle stream that is not unreasonably large, and that the pressures are such that no special equipment is required. The Methanol and Acetone system is such a system and the separation shown in Figure 3.26 consists of two columns, the first operating at 1 *bar* and the second at 6 *bar*. The recycle flow is 59% of the feed flowrate (when the feed is equimolar). Note that the feed coming from the second column at higher pressure can be fed directly back to the first column as *ChemSep* takes care of flashing the feed at 1 *bar*. In the actual process this stream would be pumped and flashing would occur over the feed flow control valve.

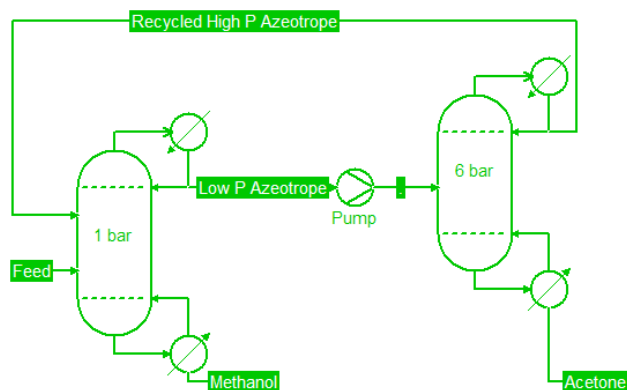


Figure 3.26: Azeotropic Distillation at Two Pressures.

3.2.4 Petlyuk Columns (PETYLUK)

Three different species can be separated using two simple distillation columns (as long as there are no azeotropes to complicate things). An alternative configuration that lowers the energy consumption of a conventional separation train is to use the so-called Petlyuk column shown in Figure ?? that involves only one condenser and reboiler.

The feed to be separated goes to the first column. Since the first column has no condenser or reboiler it needs a vapor feed to the bottom and a liquid feed to the top. These streams are provided by the second column from which they are, in fact, withdrawn as sidestreams. The products of the first column go into the second column at the the same sidestream stages.

In the example that comes in the flowsheet directory of *ChemSep* we included a Petlyuk column combination for the separation of n-propanol, isobutanol, and n-butanol. The *ChemSep* model involves just two columns; there are no extra flashes, make-up streams, or stream splitters. To get this flowsheet to run requires that you first solve both the columns separately; this is not easy. The second column needed a user supplied initialization information in order to obtain a solution. Subsequent convergence of the flowsheet (or of this column by itself) can be obtained more easily by using the old results as initialization.

3.2.5 Liquid-Liquid Extraction with Solvent Recovery (BP)

Extraction can be used to separate aromatics from paraffins. This is a common type of separation in the refining of crude oil. Figure ?? shows a simplified process flowsheet in which

we separate benzene from n-pentane, using a liquid-liquid extraction column with sulfolane as solvent. The extractor is a rotating disk contactor (RDC) operating at $50^{\circ}C$. The solvent is recovered from the extract by simple distillation. The UNIQUAC model is used for the activity coefficients in the extractor; parameters for this model are given in Table 3.1. The extraction is sensitive to the temperature as can be seen from the interaction parameters; this fact can be used to influence the separation. Note that sulfolane only melts at around $27^{\circ}C$ and this limits the temperature for the extraction.

Table 3.1: UNIQUAC LLE interaction parameters (K)

Components i-j	A_{ij}	A_{ji}
25°C:		
pentane-benzene	295.38	-171.63
pentane-sulfolane	532.04	187.84
benzene-sulfolane	151.02	-36.08
50°C:		
pentane-benzene	179.86	-95.70
pentane-sulfolane	375.93	247.77
benzene-sulfolane	131.51	-6.28

The RDC extraction column does not achieve such a good separation as *ChemSep* suggests, largely because backflow in the RDC has been ignored. In practice, the feed would not be to the bottom of the extractor but more towards the middle. The aromatics concentration would be increased by recycling some of the distillate of the second column to the extractor.

3.2.6 The Separation of Air

Distillation has long been used for air separation. The three major components of air - nitrogen, oxygen and argon - do not form any azeotropes and can be separated by simple distillation, albeit carried out at cryogenic temperatures because these compounds boil at very low temperatures ($77.5 K$, $90.3 K$, and $87.3 K$ respectively, at atmospheric pressure). In fact, the relative volatility between oxygen and argon is close to unity, especially in the pure argon region. Cooling is achieved through the Joule-Thomson effect: if we expand higher pressure air it cools down and can be partially liquefied. To simulate this process we use the Peng Robinson EOS. Interactions parameters can be obtained from DECHEMA where various data sets are at hand, each with a different fit. Here we used the following binary parameters: N2-O2 = -0.0133, N2-Ar = -0.0026, and O2-Ar = 0.0089. Although these parameters are small, they have a profound effect on the design of air separation columns. For example the separation of Oxygen and Argon is difficult due to the reduced relative volatility in the pure Argon region.

The cryogenic air separation process is a tight integration of heat exchangers and separation columns which is completely driven by the compression of the air at the inlet of the unit. The

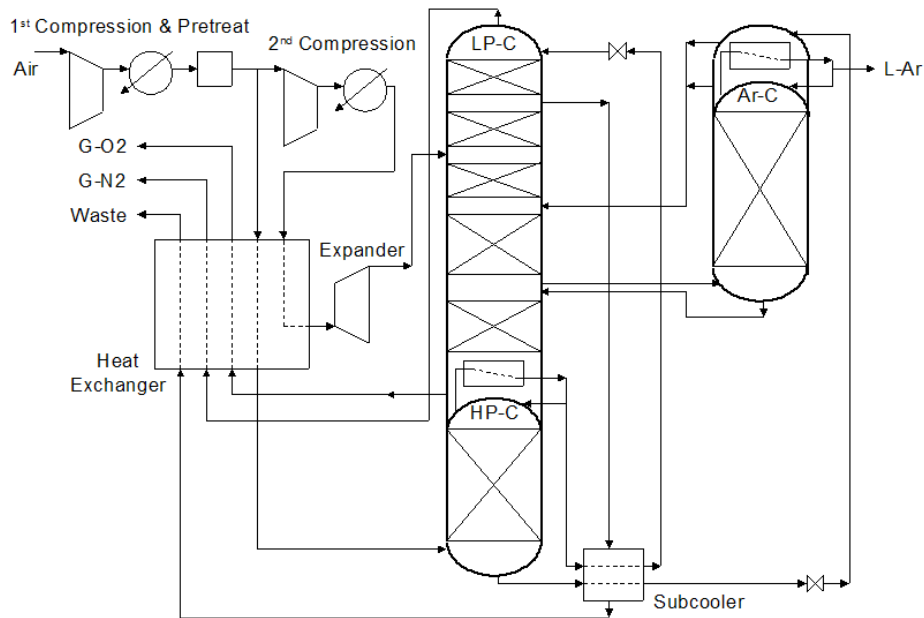


Figure 3.27: Cryogenic Distillation of Air.

air inlet stream is cooled cooled to ambient temperatures with cooling water and further cooled against product and waste streams leaving the plant until partially liquefied. The liquefied fraction varies between 25 to 35 mol%) depending on cooling water temperatures and compressor outlet pressure. This stream enters the bottom of a column that produces pure liquid nitrogen (impurities ; 1ppm) as overhead. The column has roughly 40 equilibrium stages including condenser and operates at "high" pressure. The cooling medium in the condenser is the oxygen bottom product produced in a second column. Thus, this second column must operate at a lower pressure to make the reboiler colder than the condenser of the high pressure column (typically by 1 Kelvin only). Thus, the pressures of both columns are linked mainly by the vapour pressures of Nitrogen and Oxygen and the temperature approach (heat exchanger size) of the combined condenser/reboiler. In order for the oxygen to get out of the plant the bottom of the low pressure column is typically around 1.3 bar which sets a pressure of at least 5.7 bar for the high pressure column. This requires the inlet compressor to deliver air at slightly higher pressures (here we chose 6.1 bar). These two columns share the same column shell to minimize the temperature difference between the condensing nitrogen and evaporating oxygen.

The liquid bottom product of the High Pressure Column (HPC) is rich in oxygen. When this Rich Liquid (RL) is reduced in pressure the Joule-Thomson (JT) effect causes this rich liquid to cool further. The rich liquid is fed to the second Low Pressure Column (LPC). If Argon is also produced this refrigeration is used to create reflux for a side-rectifier that separates Argon from

Oxygen. Because Argon is the intermediate boiler it builds up between the RL feed and the oxygen produced in the reboiler. Argon concentration increases from 1 mol% in the RL to up to 13 mol% at the vapor draw to the side rectifier. The side rectifier operates with a high reflux rate to further distill the Argon. This liquid reflux of this side rectifier is returned at the same location as the draw into the low pressure column. When the side rectifier is equipped with structured packing enough equilibrium stages can be included to produce pure Argon ($x_{O_2} < 1 \text{ ppm}$) while the condenser still operates above atmospheric to ensure there is no ingress of air through leaks.

When no Argon is produced the refrigeration produced by the JT effect of both the liquid nitrogen and the RL can be suffice run the whole plant. However, while producing Argon additional refrigeration will be needed. This can be obtained by expanding an additional flow of air in via an expander, using the JT effect, that feeds directly into the low pressure column. To obtain a more effective refrigeration this air is compressed in a second stage to a higher pressure, where effectively the extra refrigeration is obtained via the cooling water (but at a higher temperature). Of course, since this air is poorer in oxygen than the Rich Liquid (RL) fed to the Low Pressure Column (LPC) is, it enters the LPC at a higher feed location. Important to realize is that a certain part of the compressed air cannot be separated and has to leave the plant as a waste stream. This waste stream leaves the low pressure column as a side vapor draw. Finally, the low pressure column is refluxed with with liquid nitrogen from the high pressure column, after having been flashed and cooled by the JT effect. The liquid oxygen bottoms is gasified and heated against the incoming air, just as the pure gaseous nitrogen of the top of the LPC and the waste air stream are. The products of the Air Separation Unit (ASU) are thus Gaseous Nitrogen (GN₂) and Oxygen (GO₂), as well as liquid Argon (LAr).

With the application of structured packings in ASU's (in 1980-1990) it became possible to "pack" enough separation in the columns without incurring a too high a column pressure drop. It enabled the production of high purity Argon without the need to use a reactive oxygen removal from the crude Argon with Hydrogen. With 70 to 80 equilibrium stages in the LP column the oxygen product purity ranges from 97.5 to 99.5 mol%. For both Nitrogen and Argon purities are typically 99.999% (impurities less than 1 ppm). The Argon column is equipped with 100 to 120 equilibrium stages (see also *Comp. Chem. Eng.*, Vol. **30** (2006) pp. 1436-1446). The low pressure high capacity structured packings first introduced in 1999 by Sulzer further eased the design of these packed columns with over 200 combined equilibrium stages.

Cryogenic air separation by distillation involves tightly (thermally) linked cycles of material as well as energy where the cost of heat integration is balanced against the cost of air compression (heat integration shows itself in capital cost whereas compression cost is a production cost). This is further complicated by the many two phase flows between the various (tall!) columns and heat exchangers. Since pumping of pure Oxygen is something typically avoided and the flow regime determines the pipe pressure drop controlling the flow regime during design is very important. The tightness of the coupling of the HPC and LPC via the combined condenser/reboiler largely sets the operating pressures. With temperature differences in the heat exchangers of just one or a few degrees Kelvin there is significant interaction such that any disturbance directly affects one of the column controlled variables.

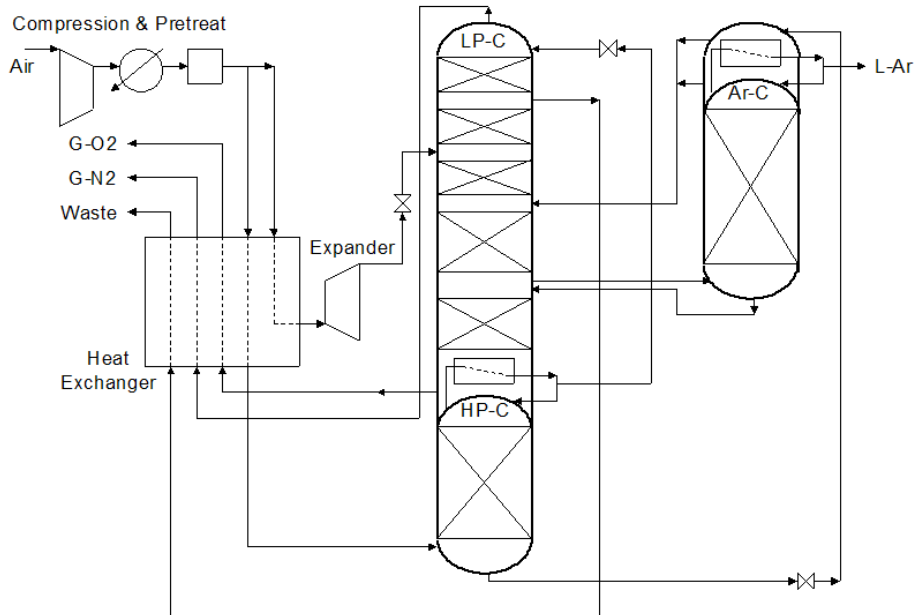


Figure 3.28: Simplified ASU.

The ASU is simulated with an air feed at 30 degrees Celcius and 1 atmosphere. Air is taken to consist of Nitrogen (N2), Oxygen (O2), and Argon (Ar) only (normally there is also moisture in the air as well as traces of other gases such as Carbon dioxide and other Noble gases such as Neon and Krypton). The gas composition is taken as 0.7812 N2, 0.2095 O2, and 0.0093 Ar. This Air is compressed to 6.1 bara in three stages using a compression ratio per stage of 1.9. The compressors are only 80% efficient and water coolers are needed to cool the inflowing air to 30 degrees C. This air flow is then split into two: 85% is cooled to a temperature of 99.4 Kelvin against streams leaving the ASU where 31 mole% gets liquified. The other 15% is cooled only to 140 K to be generating power and refrigeration in an expander. When this stream is entering the low pressure column it has cooled down to about 103 K. The split-ratio of these two streams is used to control the refrigeration in the ASU. To stabilize the plant the percentage of liquification in the main air stream to the high pressure column is controlled. That percentage is a direct function of the temperature as at that point the air still has the atmospheric condition and the splitter can be regulated with by measuring the exit temperature directly.

The high pressure column is to produce liquid nitrogen that is used to provide reflux in the low pressure column and hence it must as pure enough as the GN2 product needs to be. Hence the impurity of Argon is used to control the reflux ratio. In the simulation we set the Argon composition to 0.7ppm and the top pressure to 5.6 bar. In reality, the top pressure is controlled by the reboiler temperature of the low pressure column. As we fix the composition of that

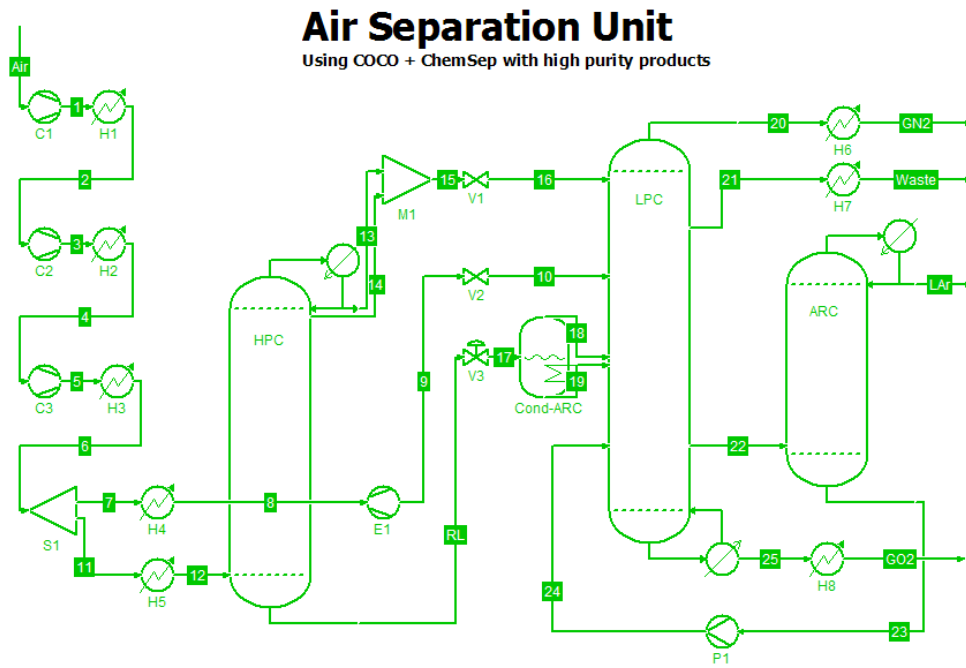


Figure 3.29: ASU in COCO.

column needs to be 98.5 mole% Oxygen, the temperature in the reboiler is largely a function of the top pressure and allowed pressure drop in that column. The top of the low pressure column is specified at 1.2 bar since the GN2 overhead still needs to be able to pass through the heat exchangers out the ASU. To obtain the desired oxygen purity we need 70-80 stages in the low pressure column and with a typical pressure drop of 2 mbar per HETP we get a bottom pressure of at least 1.35 bar. This allows for a comfortable 2.6 degrees temperature difference in the condenser/reboiler of the two columns.

In this example we'll use a simplified process where the second compression stage and the subcooler are omitted at the expense of lower recoveries. The flowsheet was simulated with one tear stream: the returning liquid stream from the Argon column back to the low pressure column. Normally, the Argon column bottoms liquid flows back to the low pressure column by means of gravity. This is simulated by using a pump that raises the pressure of the bottoms liquid, as otherwise COCO warns us that the bottoms liquid would flow against a pressure gradient (the Argon column operates at a lower pressure than the low pressure column!). Also note that there is a vapor and a liquid drawn from the condenser of the high pressure column. This is done for control purposes and allows the HPC condenser duty to be matched by a controller to LPC reboiler duty. The combined stream is fed to the top of the low pressure column. The rich

liquid bottom product of the high pressure column is flashed where the heat input is set equal to that of the condenser duty of the Argon side rectifier. The gaseous product streams are all heated up against the inlet air flow to recover heat. Heat leaks are added to the columns from the environment as well as pressure drops in the columns over the column internals, which raises the required inlet pressure for the ASU.

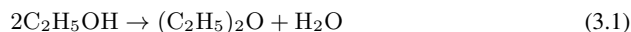
Here the three heat integrations in the ASU is shown: the cooling of the feed against the product streams in the main heat exchanger, the LP condenser against the HP reboiler, and the condensation of Argon against evaporation of rich liquid.

Easy McCabe-Thiele plots let the user quickly evaluate the process and the feed and draw locations to/from the low pressure column.

3.3 Di-Ethyl Ether from Ethanol-Water Feed

We consider the production of diethylether from an ethanol-water (85%-15%) feed, as shown in Figure ???. The input file for this flowsheet, *ep.fs*, can be found in the **fs** directory of the *ChemSep* distribution. Use the load option to load this file, after which you can peruse it using the edit option.

In this process ethanol is converted to diethylether in the *Reactor* which has a conversion of 50%. The reaction is:



Diethylether (99.5 mol%), is recovered in a distillation column *Sep-1*. The ethanol is recovered in a second distillation column and recycled *Sep-2*. We also obtain water (containing 1 mol% ethanol) as a byproduct.

The flowsheet input file defines the unit operations and streams in the flowsheet and consists of four parts. The first part consists of one section where all the components, units, streams, feeds, and stream estimates are declared. It also contains the output filename, executable directory, and the method and accuracy used in solving the flowsheet. The section starts with the **[Flowsheet]** identifier.

```
[Flowsheet]
Comment=Ethylether Production
Components=Water,Ethanol,Diethylether
Units=Mixer,Reactor,Sep-1,Sep-2
Streams=1,2,3,4,5,6,7,8
Feeds=1
Estimates=7
Output=ep.fs
ExeDir=c:\cs\exes\
```

```
MaxIter=20
Method=Direct
Accuracy=0.001
! this is a comment
```

The different sections of the input file are separated by empty lines. You can add (non empty) comment lines in the input file by, for example, using any punctuation character to start the line. The flowsheet program looks for specific keywords and if none are found the line is considered to be a comment. These comment lines will not get copied to the output file. The keywords are the identifiers left of the equal signs in the above flowsheet section. They are not case-sensitive and you may enter them in any order (some restrictions do apply, though). The keywords can be different in each section.

Comma's are used to separate the various components, streams, units, feeds, or stream estimates, and therefore, they can not be part of a name. This is important as IUPAC component names sometimes use comma's. In that case omit the comma's in the component name. **It is essential that the component names appear in the same order in the sep files for each unit!** The case of the names you specify is not important but the names in the flowsheet section are used in the rest of the report.

In our example the unit operations are a *mixer*, *reactor*, and two columns (*Sep-1* and *Sep-2*) where we separate the products. We have identified the streams that connect these units using numbers from 1 to 8 (names may also be used). There is only one feed that must be specified, 1, and we must estimate the flow and composition of the recycle stream (7). Stream 2 does not actually exist in this process, but is needed here since the flash program that simulates the mixer produces a vapor product stream (with no flow). Our output file is *ep.fs* but the flowsheet program **fs2** writes the output to the edit window. To save your results, you would need to use the save option in the main menu.

The maximum number of flowsheet iterations allowed here is fixed at 20. *Method* refers to the method used to converge the flowsheet calculations. In this case *Direct* substitution is used, with a convergence tolerance of 0.001. To test for convergence the differences between stream variables (temperature, pressure, flows, compositions, enthalpies) are compared between flowsheet iterations. The absolute differences between these variables or their relative differences

$$c = \left| \frac{\text{new value} - \text{old value}}{\text{old value}} \right| \quad (3.2)$$

are compared to the convergence criterion given in the input file. Absolute differences are normalized otherwise the large values for the enthalpies or pressures would cause convergence problems. The normalization factors are 10^2 for temperatures, 10^5 for pressures, and 10^7 for enthalpies.

The next part of the input file describes each of the units in turn. There are four units in our example. Each unit section starts with the **[Unit]** identifier and has five keywords: Name, File, Type, Inlets, and Outlets.


```
[Unit]
Name=Mixer
File=ep-mix.sep
Type=go-col2.exe
Inlets=1,7
Outlets=2,3
```

```
[Unit]
Name=Reactor
File=ep-reac.sep
Type=reactors.exe
Inlets=3
Outlets=4
```

```
[Unit]
Name=Sep-1
File=ep-sep1.sep
Type=go-neq2.exe
Inlets=4
Outlets=5,6
```

```
[Unit]
Name=Sep-2
File=ep-sep2.sep
Type=go-neq2.exe
Inlets=6
Outlets=7,8
```

The specified unit name must match the one given in the flowsheet section. The unit file is the associated data file, for most units it is the Sep-file that was created by *ChemSep*. The unit type is the program that must be run to simulate this specific unit. For a *ChemSep* flash that is **go-col2.exe**, for an equilibrium-stage column **go-col2.exe**, and for a nonequilibrium column **go-neq2.exe**. This allows you to make your own unit simulation program that reads a data file with feed and product section as in a Sep-file. In the case of the reactor this is handled by the **reactors.exe** program (which is described below).

The units are connected by inlets and outlet streams that were declared in the flowsheet section above. Be sure to use the same names (case is not important) as otherwise the flowsheet will be incorrect or incomplete. If you do not want to run the simulator for a specific unit, omit the unit file name and its type. The flowsheet program will consequently not execute anything for this unit.

```
[Feed]
```

```
Name=1
Temperature=40C
Pressure=1.01325bar
Rate=20mol/s
Z=0.15,0.85,0
```

The next part of the input file consists of the **[Feed]** sections where the feed streams are defined. The specified feed name must match the one given in the flowsheet section (again, case is not important). Furthermore, the feed stream pressure, temperature, flow rate, and composition must be specified. Append units when specifying temperatures, pressures, and flows. The default units (which can be omitted) are temperatures in $^{\circ}C$, pressure in *bar* (absolute), and flows in *mol/s* (see Section ?? how to change the default units). The flowrate and compositions can also be set by specifying the component flows, for example for the definition of the feed above we could use as well

```
[Feed]
Name=Feed
Temperature=40C
Pressure=1atm
F=3,17
```

F is the list of component flows (default units *mol/s*). If all the component flows are specified, the total flow rate and compositions are computed. Other possible feed specifications are *vapor fraction* or *enthalpy* (in *J/kmol*). The flowsheeting program uses the pressure and enthalpy specification internally. Only this type of specification guarantees proper handling of the streams. For this reason one must be careful when including reactive distillation columns, for the enthalpy in these column simulations includes the heats of formation (all units must switch these on, see Section ??).

All streams are reset at the start of the simulation. Only the specifications supplied in the input files are written to the unit file. If stream values are reset the values already present in the unit-file will be used.

The last part consists of the **[Estimate]** section where estimates of stream variables (flow, temperature, pressure, or composition) can be provided. This part is required only if you specified streams under the estimates keyword in the flowsheet section. Stream estimates have the same input format as feeds (except for the different identifier, of course). In our case we estimate the recycle stream 7 to start with a better value of the flowrate to the *reactor*:

```
[Estimate]
Name=7
Temperature=80C
```

```
Pressure=1.01325bar  
Rate=10mol/s  
Z=0.85,0.15,0
```

Stream estimates are required for any recycles in the flowsheet because otherwise one of the unit operations will have undefined feed specifications. However, information on the recycle streams is likely to be unavailable at the start of a simulation. By specifying *Rate=0* as an estimate for the recycle stream the user will be able to start simulating the flowsheet. Upon completion or interruption of the calculations, a new estimate will be written by the flowsheet simulator.

The files that make up our example all start with 'ep' (ep.fs, ep-mix.sep, ep-reac.sep, ep-sep1.sep, ep-sep2.sep) to group them together. In the main menu of the flowsheet program you will also find a *units* option which shows a listing of the units in the flowsheet (Figure ??). Selecting one of the units will start the *ChemSep* interface and load the associated datafile as sep-file. This facilitates quick editing of the unit's specifications.

Chapter 4

Analysis of Separation Processes

In this chapter we will use the various tools *ChemSep* provides to analyze the specified problem: for example parametric studies to compare experimental data with simulations or thermodynamic model predictions with measured VLE data. Or plotting stream properties as function of temperature, pressure, or heat content as well as the rating of the column simulated using vendor tools. These analysis tools are only available when either the components and models have been defined or the problem has been solved before.

4.1 Plot Data File

Here we will explain how you can plot any data, for example experimental data that was published in a journal article or a book. As an example we will use data on Sulzer BX packing which was published in *Chemical Engineering Progress*, January 1998 (p.60). This paper was written by by Mike Lockett and the data originated from the Fractionation Research Institute FRI. The article showed "measured" HETP's as a function of the packing F-factor at four different pressures. In particular, we use a scan for Lockett's plot of HETP's for the o/p-Xylene system at 1 atmosphere, Figure 1a. We could read the values from this graph by hand but to get a little more precise numbers we use the tool "ScanIt" to obtain the datapoints. ScanIt (available for free) lets you pick the origin and two axis end points of the Figure to determine the scale of the scan. After this you can select each point with the mouse and ScanIt will mark it and compute the x,y values of the original datapoint.

With the "Copy data" button we can copy the data to the clipboard. We open a Notepad text

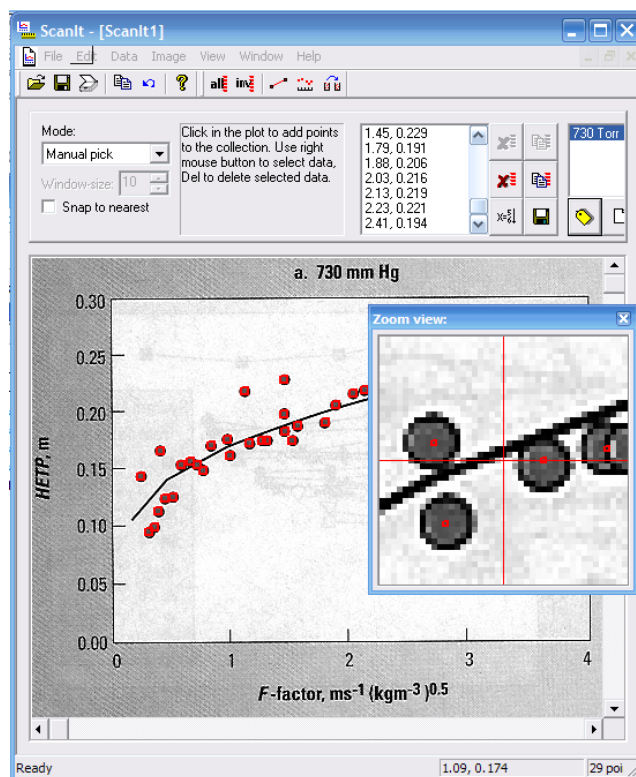


Figure 4.1: Using ScanIt to obtain the data points in the Figure.

editing window and paste the data in there. We also add some descriptions in this file:

```
# fullref= Chemical Engineering Progress, January 1998, p. 60, Fig. 1a
# author= Michael J. Lockett
# title= Easily predict structured-packing HETP
# url= http://www.aiche.org/CEP/Issues/1998-01/
# fileref= cep/cep1998-01p60.pdf
# subref= fig1a
# type= structured
# vendor= Sulzer
# name= BX
# system= xylenes
# pressure= 96 [kPa]
# location= FRI
```

```

# diameter= 1.22 [m]
# bedheight= 3 [m]
# xbot_ave= 0.3
# data_y= ave.HETP [m]
# data_x= F-factor [(Pa.s)^0.5]
# data_type= experiments
# copleft= Harry Kooijman, ChemSep.org

```

this basically gives a complete reference to the article. If possible, we refer to the publishers URL of the paper. Note that this is not always possible, as in this case, as CEP does not have an electronic versions of their older publications. In this case we make a link in the same way as it works for the current publication of the journal. Important to realize is that any datalines starting with a "#" is not a data line. All these entries are providing meta information about the data that follows. Does the data represent experimental values or model predictions? What was measured as a function of what, in what units? What chemical system was used, at what pressure, and in how large a column? What was the (average) molefraction in the bottom of the column of the more volatile compound? We can add a title and a label with:

```

# plottitle=Sulzer BX o/p-Xylene 730 Torr @ FRI (1.22m ID) CEPjan1998p60fig1a
# plotstage=Vertical
# plotcommands=set label "ChemSep" at 3,0.3 right

```

The title gives packing type, test system and pressure, column ID, and reference. The label is just an example of text which is right aligned. There is also another ways of making labels, as we will see later. We can now also define the X- and Y-axis with

```

# x1-title=F-factor
# x1-start=0
# x1-end=3
# x1-ticint=1
# x1-stics=4
# x1-log=Off
# y1-title=Average HETP (m)
# y1-start=0
# y1-end=0.3
# y1-ticint=0.05
# y1-stics=4
# y1-log=Off

```

Note that ChemSep can draw 4 axes, and that the default x- and y-axis are x1 and y1. This is the notation used by GNUplot, the tool ChemSep uses to make its plots. To make define the color and style of the datapoints we can use:

```
# label= exp.  
# x-axis=1  
# y-axis=1  
# x-units=  
# y-units=  
# color=1  
# style=1  
# thick=1  
# point=6  
# flipp=0
```

which defines the datapoints as red filled circles labeled as "experimental" points. Next we paste the content of the buffer with the data collected by ScanIt:

```
0.242 0.145  
0.306 0.0955  
0.345 0.1  
0.385 0.115  
0.398 0.168  
0.444 0.126  
0.512 0.126  
0.581 0.155  
0.659 0.158  
0.707 0.155  
0.765 0.15  
0.825 0.172  
0.97 0.178  
0.989 0.163  
1.12 0.219  
1.15 0.173  
1.26 0.176  
1.3 0.176  
1.45 0.199  
1.45 0.229  
1.45 0.183  
1.52 0.175  
1.55 0.188  
1.78 0.191  
1.87 0.206  
2.03 0.216  
2.13 0.219  
2.22 0.222  
2.41 0.194
```


We save this as a text file with a descriptive name, for example cep1998janp60Fig1a.txt, where the name uses the abbreviated journal name, followed by the volume number, a small "p" for page, and then the page number. This usually generates a unique name. When the journal does not use volume numbers but year and month instead, we use those, with the month as three small letters. For conferences we use an abbreviated conference name, year, and a session number or a page number of the proceedings.

The resulting text file can directly be plotted with *ChemSep* using **Analysis - Plot Data File** menu-item. The plot has the same axes as it was originally published with as well as labels in the plot. When multiple experimental data sets are shown in one graph it is better to separate them into separate files and call them for example fig1a, fig1b, fig1c etc. This will allow them to be used separately in model comparison studies as we will see below. You can always combine them into one figure by adding another text file, for example as "fig1", by appending the files into that file. Make sure you give each set a different label and color/point-type so that they remain distinguishable! Our example will look like Figure 4.2.

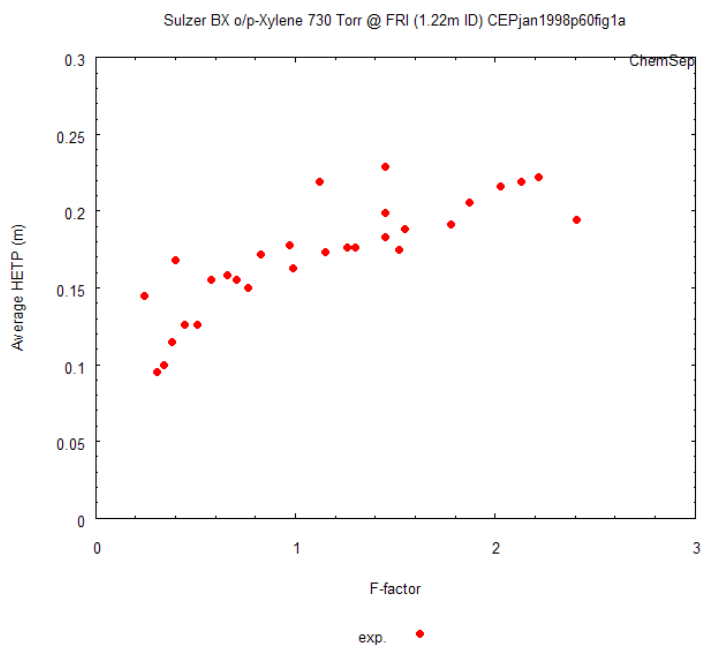


Figure 4.2: Showing the distillation test data with the "Plot data file".

4.2 Parametric Study

The Sulzer BX packing data as plotted with the Plot Data File option above can be compared with predictions of mass transfer coefficient (MTC) correlations in *ChemSep*. Lockett also made such a comparison with the mass transfer model of Bravo, Rocha, and Fair (Hydrocarbon Processing, January 1985, p. 91). Here we discuss how to do the same using the *ChemSep* **Analysis - Parametric Study** option. To make this comparison you need to setup a simulation of the distillation tests in *ChemSep* for the system in question. We like to use the same filename as the dataset and add to it the model name (abbreviation), for example for the BRF85 model `cep1998janp60Fig1a_BRF85.sep`. The easiest way to get this done is by searching our online database for a simulation (sep) file with the same distillation test mixture (and pressure) and adapt it for the right packing, bed height, and MTC model.

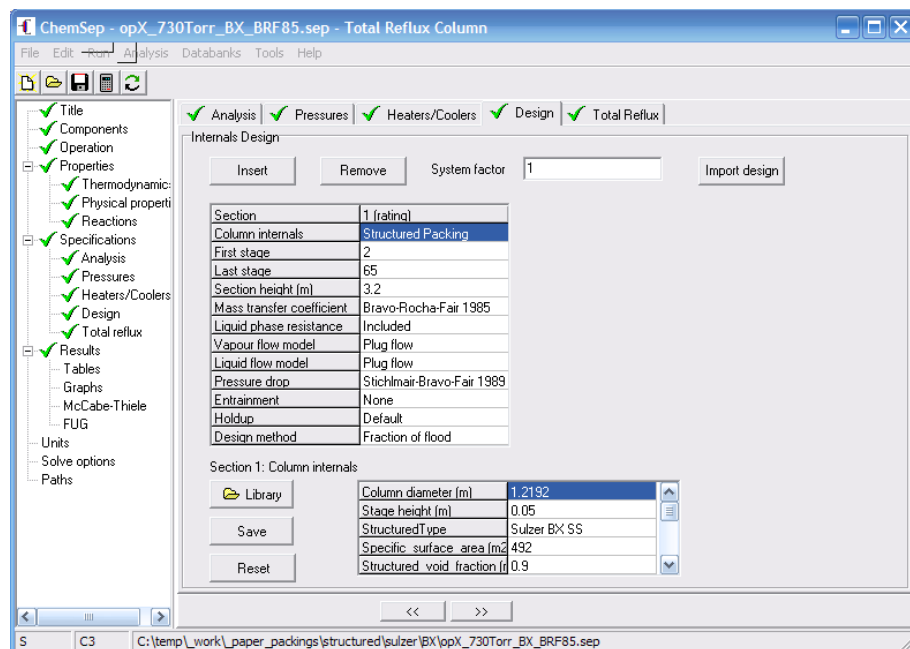


Figure 4.3: Setting up the o/p-xylene simulation with the BRF'85 MTC model.

It is important to realize that "an HETP" can not measured as such but that it is computed by simulating the distillation tests with an equilibrium stage simulator while matching measured bottoms and tops compositions. To get a good match one stage is to be taken to have a fractional stage efficiency (between 0 and 1). The HETP is then resulting from the bed height divided by the number of stages in the simulation (of course, excluding condenser and reboiler. If the vapor velocity in the bed changes it is important to use the location with the highest flows for computing the F-factor. When we have a converged simulation and verified it is correct we need

to simulate it at various flow rates using **Analysis - Parametric Study** option in *ChemSep*. We vary the reflux flow to the column to simulate the tests at different vapor loadings. You might need to play a little with the liquid reflux rates to obtain the right range for the test data. Note that we chose to plot the average HETP, by using AHETP1. This computes an average HETP over the whole bed of section number 1. We plotted it as function of the F-factor on stage 35, which resides in the middle of our bed.

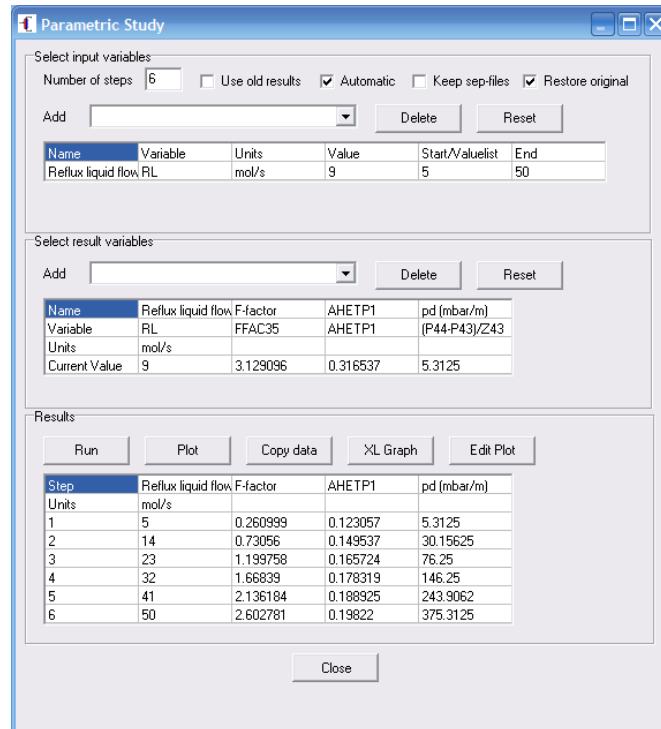


Figure 4.4: Using Parametric Study to simulate the column at different vapor loadings (and hence F factors) and calculating the average HETP.

Then plot the data points again with the "Plot data file" menu item and switch on the "Graph settings". This will allow you to add a data-set with the "Add set" button. Here we plot the second and third column in the Parametric Study, by using the identifiers PS2 and PS3, on axes x1 and y1 for the 6 points generated in the Parametric Study (adapt this if you use a different number and indices for the variables).

When we now click the "Display" button we obtain the experimental HETP and the model predicted HETP in one graph. We see that there is quite a good agreement with Lockett's published model predictions for the BRF'85 model. Deviations stem from using the average HETP, different physical properties and the use of a full nonequilibrium model. It is important

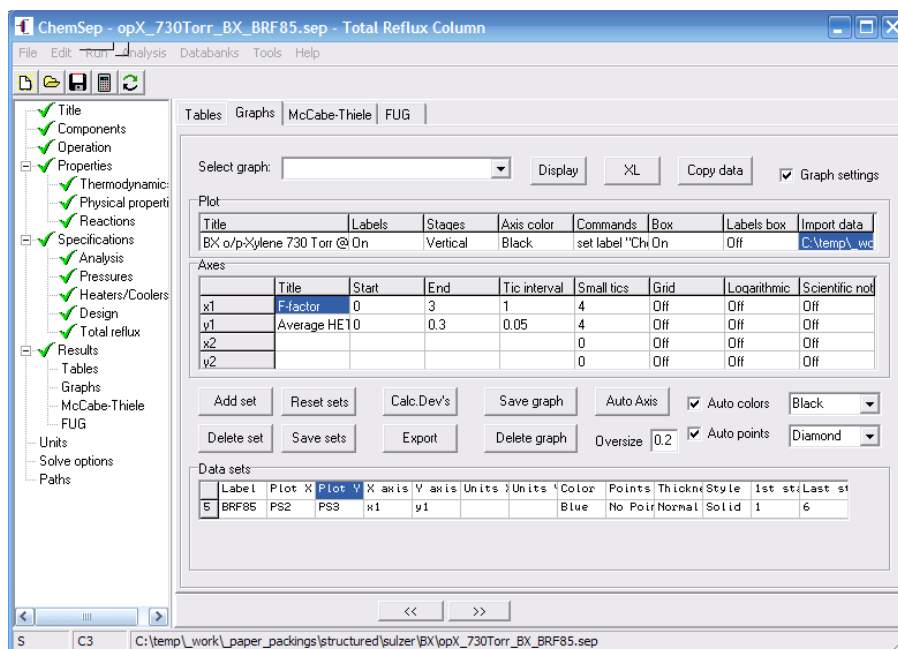


Figure 4.5: Use "graph settings" to add the BRF'85 prediction next to the experimental data.

that we use a sufficient number of stages to integrate the packed bed and the correct flow models are chosen in the simulation, of course.

To compare various MTC models we need to be able to also compute the standard deviations in this plot. It would also be nice if we can visualize the model deviations. We can! Simply click the "Calc Dev's" button and these deviations will be computed for all data points. Note that the range of the parametric study must be large enough for this to work. ChemSep uses rational inter- and extra-polation to compute the model HETP's for each data point. The overall standard deviation in HETP and relative HETP is also added as label to the graph. This is all written to a new data file which includes the deviations as individual line pieces, and ChemSep asks you to specify a new name for this file. Typically we append a digit for the text file, for example: Structured/Sulzer/BX/cep1998janp60Fig1a1.txt. When you inspect this file you see also that ChemSep appends a label with the deviation and the relative error:

```
# string 0 0.3 s=0.024 r=0.143 b=+0.012
```

This allows you to further add other labels to the graph. The label may contain spaces but

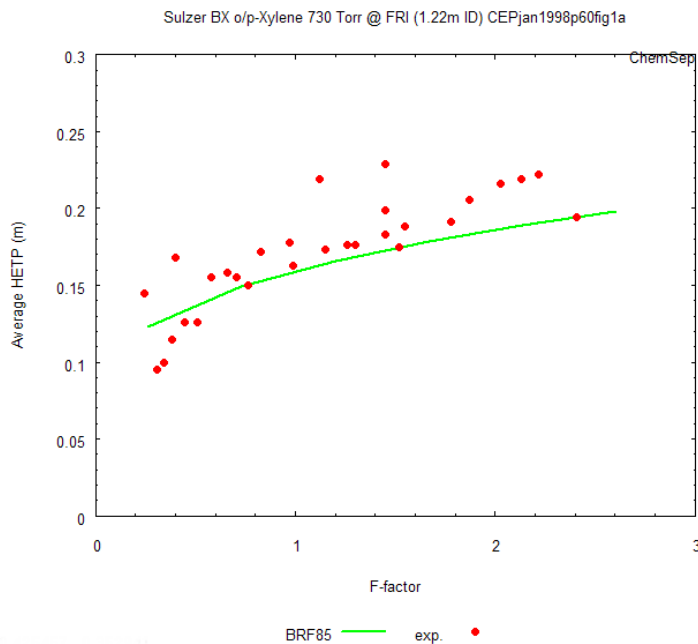


Figure 4.6: The BRF'85 prediction with the FRI experimental data for Sulzer BX packing with o/p-Xylene at 1 atm shows a good agreement.

special characters might not print properly.

We are now ready to compare the results for different models and select the best MTC model for this packing (and distillation system). We welcome anybody to contribute to our online database of distillation tests. We try to cover all industrially relevant packings. Obviously, this only works if the vendor makes its test data available or someone publishes measured distillation tests on the packings.

4.3 Column Rating

ChemSep can perform a simple column tray rating to determine tray and overall column cost estimates (using tray correlations from Economopoulos, 198x, and cost correlations by Douglas, 1988). This preliminary rating can then be used to initialize software of industrial column internals vendors, for example SulCol by Sulzer Chemtech, that do more detailed column internals rating using the vendors own (proprietary) capacity correlations.

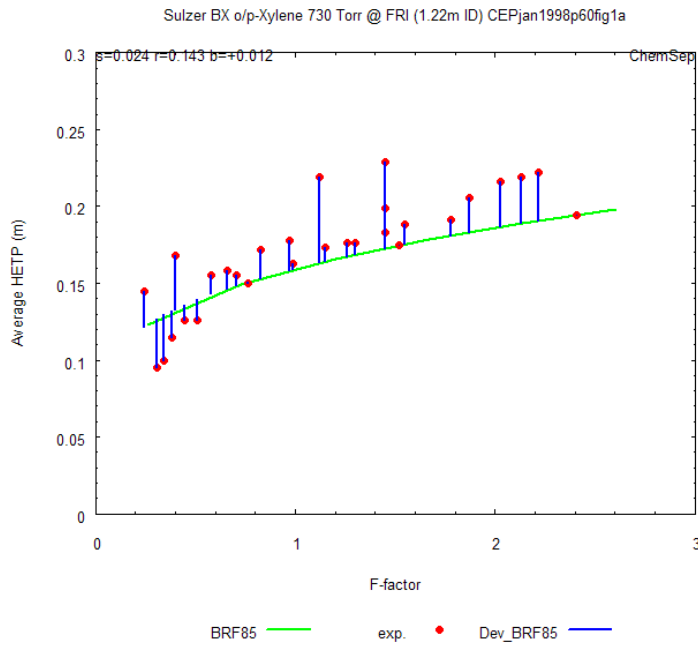


Figure 4.7: The BRF'85 model shows a relative standard deviation of 14 per cent or 24mm on an HETP that increases from 120mm at low vapor loadings to 200mm at high vapor loadings. Note that the bias is positive (+12mm), meaning that the BRF'85 model is 7 per cent optimistic in its prediction of the HETP!.

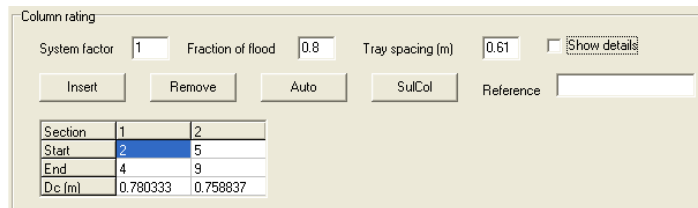


Figure 4.8: Column rating.

Note that in Figure 4.8 the column is split into two sections automatically by *ChemSep*, as the feed to the column is used to automatically split up the column in a rectification and a stripping section, that due to their different internal traffic have different diameters. You can insert or remove any section, in case this automatic division is not sufficient.

4.4 Stream Curves

Stream curves are often used to perform sizing of heat exchangers for condensers, reboilers, and heaters/coolers for feeds and product streams. To size these exchangers it is very important to determine the temperature and physical properties (especially densities, viscosities, and thermal conductivities) as function of the heat content of the stream flowing through the exchanger.

4.5 Phase Diagrams

In *ChemSep* it is quick and easy to analyse the VLE of a mixture at hand. It is even possible to make comparisons with measured VLE data. We'll use binary VLE data on Acetone and Benzene measured 25 degrees Celcius as an example. It was published by A. Tasic, B. Djordjevic, D. Grozdanic, N. Afgan, D. Malic, in Chem. Eng. Sci. (1978), 33, pp. 189-197. Many binary VLE data sets published in literature are collected in the thermodynamic libraries. The most well known of these libraries is the DECHEMA data series of which there is also an online version, called Detherm. This library includes the data of the Dortmund Daten Bank (DDB).

We will start with a new file in ChemSep and enter the VLE Data in the comments section of the sep file. You can find the VLE data in our set of binary vle (bvle) files using the library indices of acetone, 1051, and that of benzene, 501. The data file is then called "bvle_1051_501_t25.txt". The order of the components is set to increasing normal boiling points. Since acetone boils at a lower temperature than benzene it comes first. The component indices are separated with an underscore since the uses of spaces in file names makes the use of commandline tools difficult. As a last part of the file we add an indicator for the conditions at which the measurements were taken, here at constant temperature of 25 C. We can use copy-paste for entering the data as text in the comments field:

```
# file=bvle_1051_501_t25.txt
# ref=ces33p189
# title=P-xy data for acetone (1051) / benzene (501) at 25 C in torr
# fullref=A.Tasic, B.Djordjevic, D.Grozdanic, N.Afgan, D.Malic, Chem.Eng.Sci. (1978), 33,
p x y
95.05 0 0
116.65 0.082 0.248
137.2 0.185 0.416
156.3 0.307 0.55
169.3 0.401 0.632
182.1 0.505 0.703
191.2 0.59 0.755
203.2 0.704 0.824
213.4 0.8055 0.882
```

```
220.95 0.8955 0.936
222.35 0.909 0.944
225.05 0.932 0.958
230.4 1 1
```

Note that we separated the data here by tabs. ChemSep's import option for binary VLE data also allows spaces, comma's, or semicolons as separators. Important is that there is a line describing the type of data in each column. When data is only available in graphical form you can use the tool ScanIt to obtain the datapoints. ScanIt lets you pick the origin and two axis end points to determine the scale of the scan. After this you can select each point with the mouse and ScanIt will mark it and compute the x,y values of the original datapoint. With the "Copy data" button you can copy the data to the clipboard and from there to a sep file or a text file. Be sure to always include the complete reference to the original article besides a descriptive title. If possible, we refer to the publishers URL of the publication. We use specific set of journal abbreviations and refer to the volume and first page of the publication, in this case "ces33p189". The title and reference are put on lines that start with a "#" to indicate the line does not contain data. By including lines with the file name, short and full reference, and title, it is easier to automatically scan a lot of files for specific information. It takes a little effort to do this systematically for each file but the pay-back is huge when many of these small files are compiled into free online resources. If you send us your files without references they are of little use to us as we can not make them available to others.

When we want to compare the VLE data to predicted values by thermodynamic models we need to define the components as well as make a model selection and provide the interaction parameters for the models. Select "Components" in the tree of the input navigator on the left hand side. This will open a pane where we can select the compounds. ChemSep shows the compounds in the default component library (a "PCD" file) and you can select any of these by double-clicking a compound. Typing a (part of a) name in the "find" box will narrow down the list of components to only those with the particular string in the name (clicking the "advanced" checkbox allows for more sophisticated searched). We add the components in the order of their boiling point, first acetone and then benzene:

Next we click on the "properties" in the input navigation tree to select the thermodynamic models. To describe Vapor-Liquid equilibria ChemSep uses K-values, which define the ratio of the vapor to the liquid compositions ($K=y/x$). For ideal mixtures the K value equals the ratio of the component vapor pressure to that of the system. For nonideal systems such as the system at hand, this is insufficient, and an activity coefficient is needed to describe the non-ideality of the liquid mixture fugacity. The resulting model for the K-values is called the DECHEMA model, since it is also used in the DECHEMA data book series. As activity model we will first select a predictive model, UNIFAC, which will require no further input. As vapor pressure model we select the Antoine model.

To display our calculations in the same units of measure, select "units" in the navigator. Our experimental data was in degrees Celcius and torr (mm Hg). Since torr is not part of the standard list of pressure units you must type it in the edit box. ChemSep allows you to do so and thus

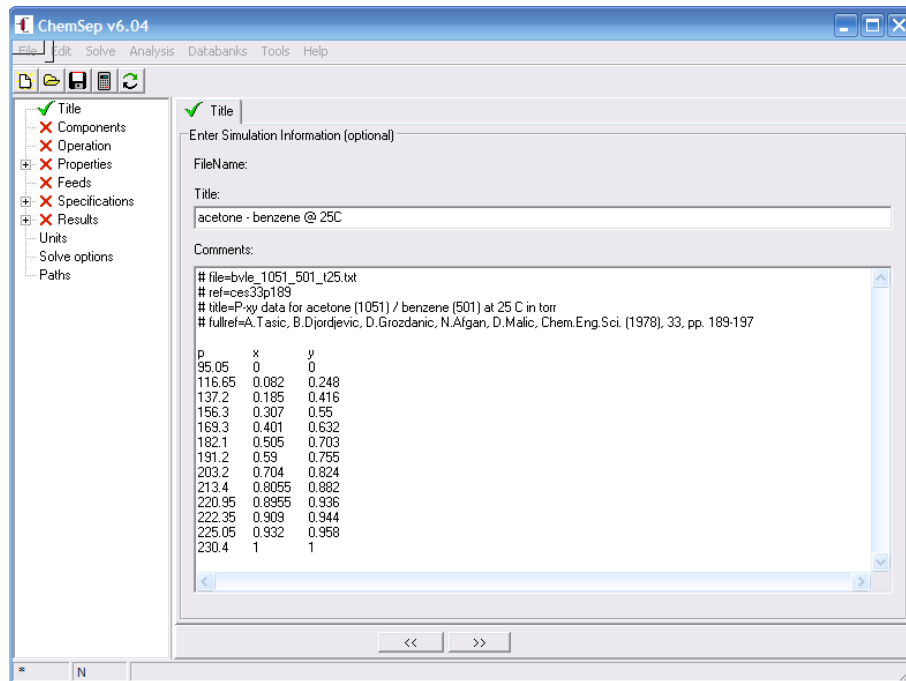


Figure 4.9: Entering VLE data in a new file.

compose all kinds of personal units of measure. ChemSep knows most chemical engineering units and all the prefixes to scale them.

Now save the file under a descriptive name, for example "bvle_1051_501_t25.sep". Then go to Analysis - Phase diagrams - Binary menu option. This opens a window to generate our pxy diagram. Select the type of diagram (pxy) and phase equilibrium (VLE) as well the condition (temperature = 25C) and click on calculate to generate the phase diagram:

Now we want to make a comparison of the model predictions with the experimental data. For this we must load the data either from a file or from the comments section. For both you must click the "file open" icon next to data. In our case we must select the "read from comments field" file type and select to open any file. This loads the following information onto the data line (all in one line):

```
pxy;
p=95.05, x=0, y=0;
p=116.65, x=0.082, y=0.248;
```

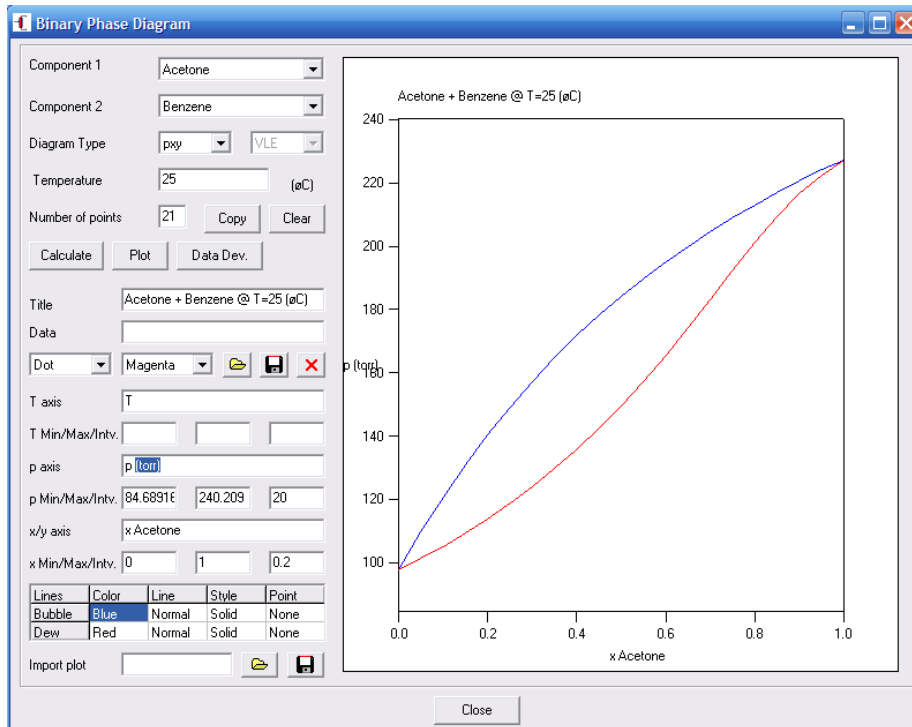


Figure 4.10: VLE model predictions.

$p=137.2, x=0.185, y=0.416;$
 $p=156.3, x=0.307, y=0.55;$
 $p=169.3, x=0.401, y=0.632;$
 $p=182.1, x=0.505, y=0.703;$
 $p=191.2, x=0.59, y=0.755;$
 $p=203.2, x=0.704, y=0.824;$
 $p=213.4, x=0.8055, y=0.882;$
 $p=220.95, x=0.8955, y=0.936;$
 $p=222.35, x=0.909, y=0.944;$
 $p=225.05, x=0.932, y=0.958;$
 $p=230.4, x=1, y=1;$

We see the data is processed: the pxy identifier is repeated and then used for identifying each data item, separated by comma's. The data points themselves are separated by semicolons. Note that the original data did not have units, so no conversions are done. The diagram now has the points shown as entered:

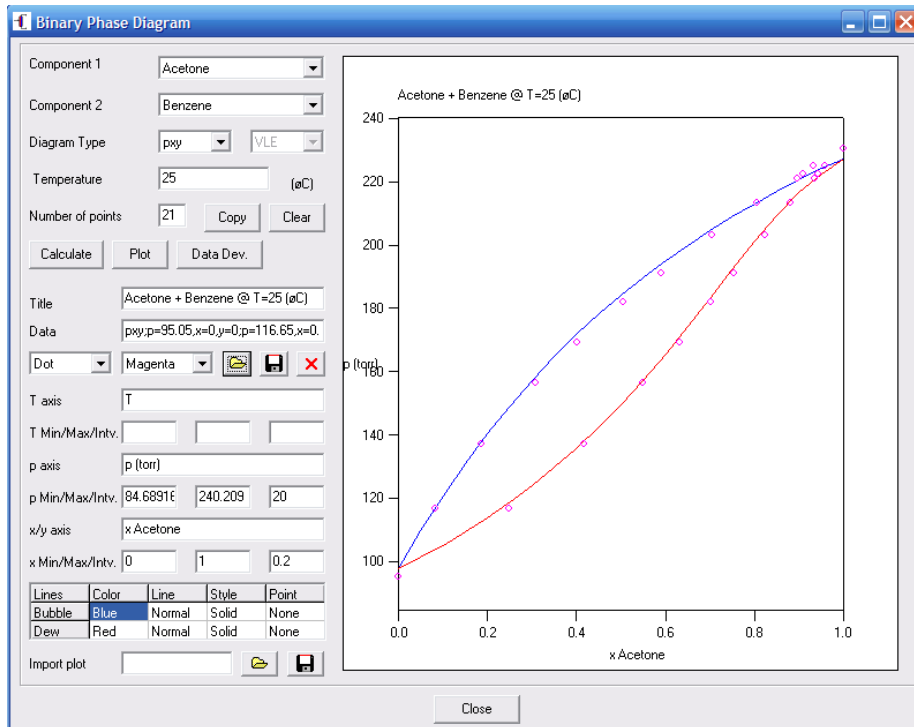


Figure 4.11: Comparing experimental data with UNIFAC/Antoine model predictions.

To quantify the deviations between the model and the experimental points we can compute the standard deviations by clicking the "Data Dev." button. This computes for each point the pressure q and vapor fraction z that is in equilibrium with the measured liquid composition x and the specified temperature 25 C. ChemSep adds these to the data line:

```

pxy;
p=95.05, x=0, y=0;
p=116.65, x=0.082, y=0.248, q=116.8901, z=0.231314;
p=137.2, x=0.185, y=0.416, q=137.7293, z=0.414939;
p=156.3, x=0.307, y=0.55, q=158.3805, z=0.557008;
p=169.3, x=0.401, y=0.632, q=171.8393, z=0.636984;
p=182.1, x=0.505, y=0.703, q=184.7208, z=0.708714;
p=191.2, x=0.59, y=0.755, q=193.9699, z=0.759586;
p=203.2, x=0.704, y=0.824, q=204.9321, z=0.822478;
p=213.4, x=0.8055, y=0.882, q=213.554, z=0.877809;

```

$p=220.95, x=0.8955, y=0.936, q=220.3915, z=0.930066;$
 $p=222.35, x=0.909, y=0.944, q=221.3489, z=0.938402;$
 $p=225.05, x=0.932, y=0.958, q=222.9353, z=0.95301;$
 $p=230.4, x=1, y=1;$

It also computes the relative standard deviations (only for the binary points) and adds them to the title of the diagram. Here we see that the UNIFAC+Antoine models predict vapor compositions (sy) with 0.72% and the pressure (sp) with 1.01%. This is actually very good. Actually most of the deviation in the pxy diagram is caused by the Antoine model not capable of predicting the exact pure component bubble pressures.

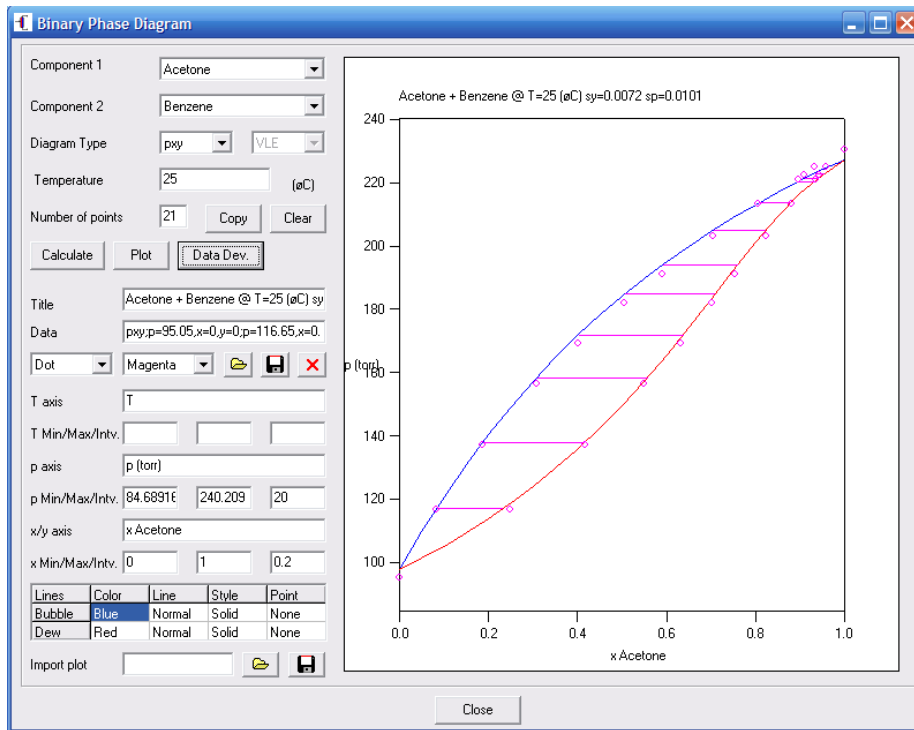


Figure 4.12: Calculating standard deviations in y and p.

Once you are happy with the way your VLE diagrams looks like use the Plot button to generate a GNUplot diagram. From GNUplot you can bring the plot into any other windows program supporting MS meta-graphics, such as Word and powerpoint, by pressing Alt-Space in GNUplot and selecting Options - Copy to Clipboard.

Though the fit of the predictive UNIFAC model looks very good, we see that close to pure acetone the dew and bubble point lines start to curve at a decreasing slope, indicating that we

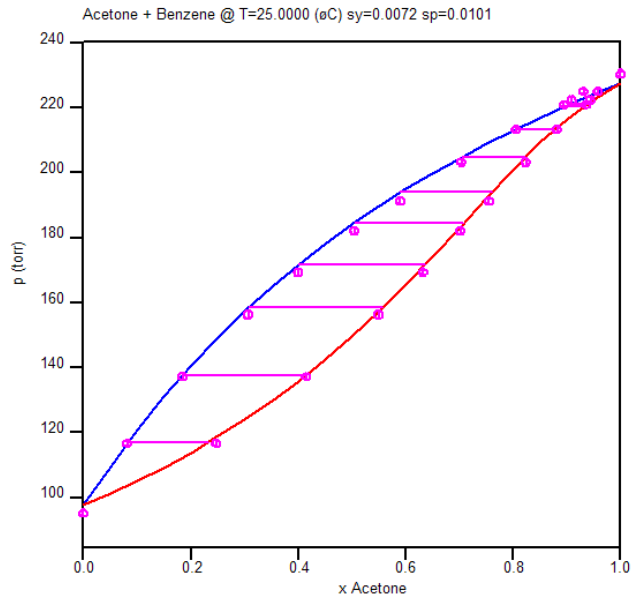


Figure 4.13: VLE diagram in GNUplot.

are close to an azeotrope. Actually, the data does not seem to have this tendency. When we go back to the thermodynamic model selections and select the UNIQUAC model and load the parameters from the library,

we will obtain a fit that is less good - deviations in pressure increase by more than a factor two - but the curvature of the dew and bubble point lines is correct:

Thus, lower relative deviations of pressure and mole fractions aren't necessarily indicative for a qualitative better description of the binary VLE. This is due to the errors in the pure component vapour pressures. Errors over half of the concentration range are introduced when a pure component bubble point temperature is off by just one or two degrees! We leave it to the reader as an exercise to select different models for the vapor pressure to see how the predictions can vary.

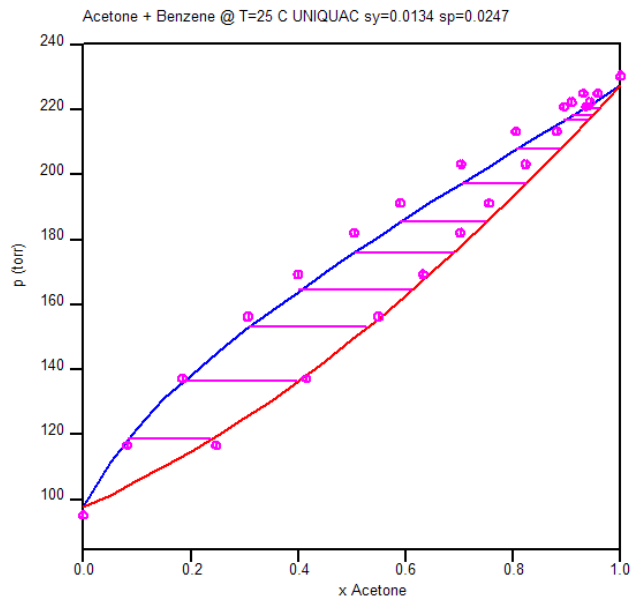


Figure 4.14: UNIQUAC fit of the VLE data.

4.6 Residue Curve Maps

In ternary or multi-component mixtures, the distillation path in the column can be analysed by means of ternary Residue Curve Maps (RCM).

4.7 Property Diagrams

Just as *ChemSep* allows an easy and quick way of plotting VLE diagrams, it also allows you to do the same with any physical property. Here we will show how to plot binary diffusivities as they would occur in a distillation column. We add the following data to the comments area in the GUI:

```
# binary Diffusivities at 40 C
xA
```

```

0.000 1.700; 0.024 1.510; 0.100 1.000; 0.144 0.780; 0.200 0.680
0.254 0.635; 0.300 0.610; 0.400 0.640; 0.500 0.730; 0.590 0.850
0.600 0.865; 0.680 1.020; 0.700 1.060; 0.792 1.260; 0.800 1.275
0.880 1.440; 0.900 1.475; 0.960 1.570; 1.000 1.640

```

```
# binary Diffusivities at 58 C
```

```
xA
```

```

0.000 2.400; 0.024 2.143; 0.100 1.500; 0.144 1.220; 0.200 1.020
0.254 0.931; 0.300 0.930; 0.400 1.020; 0.500 1.185; 0.590 1.360
0.600 1.380; 0.680 1.550; 0.700 1.590; 0.792 1.792; 0.800 1.810
0.880 2.010; 0.900 2.060; 0.960 2.240; 1.000 2.360

```

Note that these represent data for property "A" where everytime the data is presented as pair of x and A values. Multiple values can be entered per line, they are separated with semicolons. The "_" denotes the end of the data. It is good to realize that property A will be the property plotted with the left axis, and B using the right axis.

```
# Activity coefficients at 1 atm using Antoine vapor pressure
```

```
# Antoine Ethanol 8.1122, 1592.864, 226.184
```

```
# Antoine Water 8.07131, 1730.63, 233.426
```

```
xAB
```

```

0.015 6.192 0.997
0.040 3.564 1.042
0.045 4.218 1.032
0.121 2.830 1.065
0.200 2.202 1.091
0.201 2.215 1.061
0.223 1.956 1.174
0.235 2.002 1.149
0.274 1.777 1.170
0.319 1.606 1.231
0.376 1.440 1.313
0.393 1.423 1.310
0.496 1.227 1.503
0.548 1.148 1.629
0.650 1.085 1.781
0.826 1.018 2.148
0.849 1.013 2.185

```

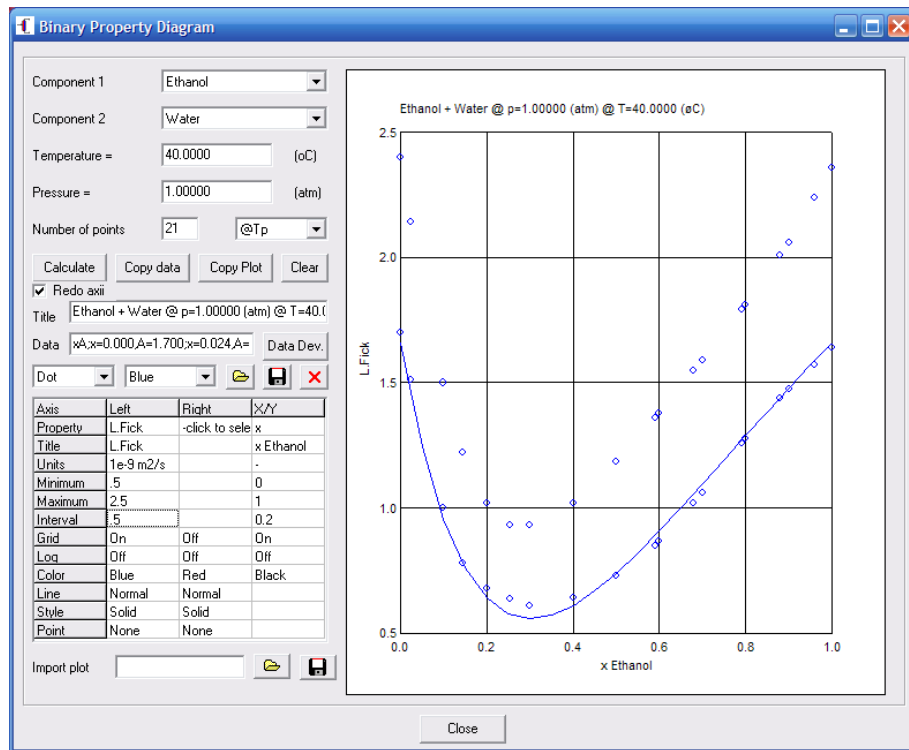


Figure 4.15: Liquid Fick diffusivities of ethanol-water at 40 degrees Celcius.

4.8 References

Economopoulos, 198x,

J.M. Douglas, *Conceptual Design of Chemical Processes*, McGraw-Hill, New York, (1988).

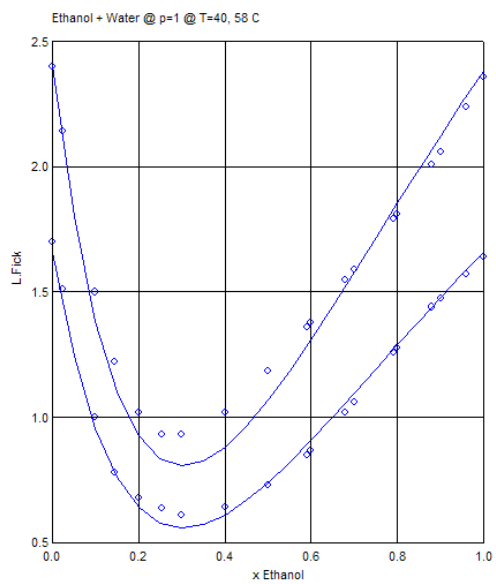


Figure 4.16: Liquid Fick diffusivities of ethanol-water at 40 and 58 degrees Celcius described with the Kooijman (2002) method.

Chapter 5

Databanks

5.1 Pure Component Data (PCD)

Pure component data can be stored in binary or in text formatted files. The interface can inspect and add/change/remove any components data. Of course, the binary data files are much smaller and as they have a fixed format file the changes in an individual compound can be directly written to the file, whereas text files require a sequential writing process than will require more time when the data files become large. The default PCD library supplied with *ChemSep* is chemsep1.pcd.

5.2 Pure Component Libraries (LIB)

Polynomial K-value and enthalpy correlation coefficients as well as extended Antoine coefficients are stored as component LIBraries (LIB files), which are ASCII files as well. The default LIB files are

- EANTOINE.LIB (Extended Antoine)
- H-POLY.LIB (Example polynomial enthalpy coefficients)
- K-POLY.LIB (Example polynomial K-value coefficients)

Here the interface starts reading the file after the [LIB] keyword. Again a comment is read from the next line (after the '=') and then the data starts with a line for each component (first the library index followed by the coefficients). For example the extended Antoine file (with data from Prausnitz et al., 1980) starts like:

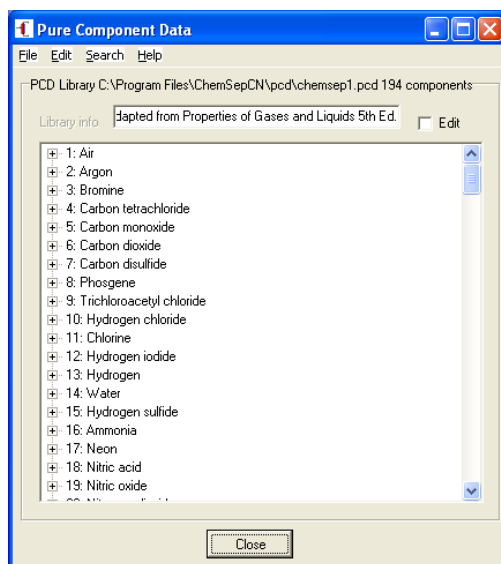


Figure 5.1: PCDman.

```
[LIB]
Comment=Extended Antoine Prausnitz et al.
#
# ID      A          B          C      D          E          F          G
902  3.15799e+01 -3.2848e+2  0.    0.    -2.5980e+0  0.    2.0 Hydrogen
905  3.48329e+01 -9.5124e+2  0.    0.    -2.5980e+0  0.    2.0 Nitrogen
901  3.57639e+01 -1.1660e+3  0.    0.    -2.5980e+0  0.    2.0 Oxygen
```

5.3 Group Contribution Data (GCD)

Group Component Data (GCD) files are binary files containing data for group contribution method such as UNIFAC or ASOG. There are three UNIFAC files (UNIFACRQ.GCD, UNIFACVL.GCD and UNIFACLL.GCD) for Vapor-Liquid and Liquid-Liquid systems. These files need not be changed, unless for example the UNIFAC group tables have changed. Several GCD-files are used for the estimation of pure component data in *ChemSep*. There are also text versions of the GCD-files, they have the GCT file extension. They can be converted from binary to text format and vice versa with the **GClib** utility. This program also allows editing of the data, see Figure 5.2.

# Sub	Group	# Main	Group	par-1	par-2	par-3
1	CH3	1	CH3	0.9011	0.848	15.034
2	CH2	1	CH2	0.6744	0.54	14.026
3	CH	1	CH	0.4469	0.228	13.018
4	C	1	C	0.2195	0	12.01
5	CH2=CH	2	CH2=CH	1.3454	1.176	27.044
6	CH=CH	2	CH=CH	1.1167	0.867	26.036
7	CH2=C	2	CH2=C	1.1173	0.988	26.036
8	CH=C	2	CH=C	0.8886	0.676	25.028
9	C=C	2	C=C	0.6605	0.485	24.02
10	ACH	3	ACH	0.5313	0.4	13.018
11	AC	3	AC	0.3652	0.12	12.01
12	ACCH3	4	ACCH3	1.2663	0.968	27.044
13	ACCH2	4	ACCH2	1.0396	0.66	26.036
14	ACCH	4	ACCH	0.8121	0.348	25.028
15	OH	5	OH	1	1.2	17.008
16	CH3OH	6	CH3OH	1.4311	1.432	32.042
17	H2O	7	H2O	0.92	1.4	18.016
18	ACOH	8	ACOH	0.8952	0.68	29.018

Figure 5.2: Editing the UNIFAC R and Q group contributions.

The GCD manager allows one to change the R and Q group contributions for each group. Note that there is a third column which contains the molecular weights. Many calculations require the molecular weight of the groups so this was added as a group parameter. UNIFAC uses subgroups and main groups, for which both numbers can be edited. Note that the group names are not used in the calculations. After exiting the GCD manager will prompt you whether you want to save the group contribution data (in the same type of file format as it was read).

For UNIFAC, there are also the group interaction parameters to calculate the residual terms of the liquid activity coefficients. Thus, a second type of GCD file can be edited; this file contains the interaction parameters for all of the main groups. Figure 5.3 shows the GCD manager edit window for this file. Note that interaction parameters that are unknown are set to zero so that the residual part also becomes zero and that these interactions do not contribute to liquid activity coefficients.

5.4 Interaction Parameter Data (IPD)

The Interaction Parameter Data (IPD) files are ASCII text files with the interaction parameters for activity coefficient models and equations of state. Currently we have the following IPD files for liquid activity coefficient models:

i	Name	j	Name	A-ij	A-ji	.
1	CH2	2	C=C	86.02	-35.36	15.034
1	CH2	3	ACH	61.13	-11.12	14.026
1	CH2	4	ACCH2	76.5	-69.7	13.018
1	CH2	5	OH	986.5	156.4	12.01
1	CH2	6	CH3OH	697.2	16.51	27.044
1	CH2	7	H2O	1318	300	26.036
1	CH2	8	ACOH	1333	275.8	26.036
1	CH2	9	CH2CO	476.4	26.76	25.028
1	CH2	10	CHO	677	505.7	24.02
1	CH2	11	CCOO	232.1	114.8	13.018
1	CH2	12	HC00	507	329.3	12.01
1	CH2	13	CH2O	251.5	83.36	27.044
1	CH2	14	CNH2	391.5	-30.48	26.036
1	CH2	15	CNH	255.7	65.33	25.028
1	CH2	16	(C)3N	206.6	-83.98	17.008
1	CH2	17	ACNH2	920.7	1139	32.042
1	CH2	18	Pyridine	287.7	-101.6	18.016
1	CH2	19	CCN	597	24.82	29.018

Figure 5.3: Editing the UNIFAC group interaction parameters.

- MARGULES.IPD
- VANLAAR.IPD
- WILSON.IPD
- NRTL.IPD
- UNIQUAC.IPD
- UNIQUACP.IPD (UNIQUAC Q')

and for equations of state:

- PR.IPD (Peng-Robinson)
- SRK.IPD (Soave-Redlich-Kwong)
- HAYDENO.IPD (Hayden O'Connell Virial)

These files are in plain text format file so that the user can add more data (it is probably better to backup the original files or to use a new name for the extended files). The files may be edited using any ASCII editor, or the built-in editor in *ChemSep*.

The actual data in these files comes after the line with the [IPD] keyword. The next line contains the file-comment after the '=', which will be used as a header in the interface. Next comes a line (only for activity coefficient models) with the interaction parameters units. Then comes the

actual data, one line per binary pair. The first two numbers are the component library indices, then the interaction parameters. Appended text is optional but will also be displayed. As an example we include the first relevant lines of the NRTL.IPD file:

```
[IPD]
Comment=DECHEMA NRTL data @ 1atm.
Units=cal/mol
#
1101 1921 -189.0469 792.8020 0.2999 Methanol/Water p61 1/1a
```

Lines starting with a '#' are comment lines which may appear anywhere. Since the interface will only start reading the file from the [IPD] keyword on, you can start the file with some text describing where the data was obtained and remarks on who/when/how changed the file. Most of the IPD files contain data from the DECHEMA series, a very extensive collection of interaction parameters. The Hayden O'Connell virial and the UNIQUAC Q' parameters are from Prausnitz *et al.* (1980).

5.5 Column Internals Parameter Libraries (INP)

Chapter 6

Tools and Settings

This chapter discusses the various tools and options *ChemSep* has. This includes discussions of the files that *ChemSep* uses to define menu options, units of measure, nonequilibrium model column internals and models, design mode defaults, etc. This allows the user to tailor the interface and the program to the users specific needs.

6.1 Commandline Options

The following commandline options exist for the GUI (WinCS.exe):

- /debug=logfile** switch debugging on to logfile
- /memdebug** switch memory logging on
- /granu=XX** set memory granularity to XX bytes
- oFilename** load options from filename. cnf extension is added if needed
- install** generates the chemsep.cnf in the directory where winsc resides and registers ChemSepUO.dll when present
- hintsfile** saves a hintsfile in 'chemsep.hnt.' after loading the existing hintsfile. Can be used to extract all widgets in the GUI by name
- pPath** override paths. When path contains 'bin' then smart substitutions are made for the help, pcd, ipd, ild paths and the temporary path is set to the 'WINTMPDIR' as well as default synonyms file name is selected
- phlpPath** override help path with the specified path string
- ppcdPath** override pure component data path with the specified path string

- pipdPath** override property data path with the specified path string
- pildPath** override internals layout path with the specified path string
- pbinPath** override binary executables path with the specified path string
- ptmpPath** override temporary path with the specified path string
- pscrPath** override scripts path with the specified path string
- LPCDfile** sets the PCD file for displaying components to select from
- Cape-Open** switches GUI into CO mode
 - NN** match CO components by name instead of CAS number
 - CC** component check & save the sep file w/o starting the GUI
 - CHECK** perform check on supplied sep-file and save under name.chk w/o starting GUI
 - new=name** generate new sep-file with specified name
- getvar=name** write value of a results variable in the specified sep-file w/o starting GUI
- out=name** specify output filename
- setvar=name** set value of an input variable in the specified sep-file w/o starting GUI
- value=value** string with value to set
- units=units** string with units of value to set
- getvardescriptions=name** output results variable names of the specified sep-file w/o starting GUI
- setvardescriptions=name** output input variable names of the specified sep-file w/o starting GUI
- getCOvars=name** write CAPE-OPEN in- and output variables names of the specified sep-file w/o starting the GUI
- plot=XX** generate plot number XX as would show in the GUI but w/o starting GUI
- gout=gif/png/emf/svg/ps** select file format of graphical output
- bvleplot** generate binary phase vle diagram
- bvledev=name** calculate and write binary vle deviations from data to file
- cs** do not show about box or startup screen but start GUI immediately

6.2 Interface Settings and Files

The chemsep interface has a general settings screen, Fig. 6.1. This screen allows the user to set the following defaults:

Numerical Input Number of significant digits, E notation, trailing zeroes, automatic units conversions, and the color of the input field that needs to be specified or is incorrect.

Hints The file with descriptions of the fields as well as all buttons, menu options, etc. When the user want to use a different language he can select the english, german, or dutch versions. The user may specify a different time that hints remains visible or select them to not be shown.

Component selection Components can be stored in the sep file instead of loading them from the binary PCD which is the default behavior (to shorten the sep-file). Also the default identified can be selected (Name, Structure, or Formula).

File The user can specify his/her own file viewer and editor to be used

Printing A left margin and a line spacing may be used for printing tables

Definitions Here the user can inspect the various nonequilibrium models that are loaded at startup of the interface as well as defined in the *ChemSep* Model development environment (see below). Only the nonequilibrium model requires these definitions which are stored in the CHEMSEP.DEF file. The main definion types are Operations, Modeltypes, Internaltypes, Models, and Internals. The interface can reload the definitions from a different file. The user can opt for automatic assignment of column sections or a mode where (s)he needs to set them by hand.

Sep-files When you solve a problem *ChemSep* saves the new specifications to a file with the .sep extension that is read by the simulations programs. By default *ChemSep* creates a backup of the old file using the backup extension specified here. If you want to keep more than one backup change the number of backups, the maximum is nine. These are numbered 1 through 9 using the name of the sep-file together with a special character to separate the name from the backup number.

Output The user can select the number of columns in the stream table or the number of columns in the table with the column sections. When these are set to zero all the streams and column sections are put next to eachother, instead of wrapping them. The user can also switch the internal computation of the physical properties on (the values are also supplied by the simulator). Furthermore, the GUI font can be set here and program which is used for producing the graphs. By default this is GNUplot in a normal window. Alternatively, you can let GNUplot generate a GIF file and view that file in the browser. A delay for the running of the external graph tool can be set so that the files needed for the graphing are not deleted before the tool has finished running. There is also a possibility to not generate a graph but to write an XML file instead that later can be imported under "imported data". This allows the user to make overlapping graphs.

The CHEMSEP.CNF file is used to store the default configuration, such as the tools definitions, directory structure, selected models, and solve options. CHEMSEP.CNF is the default configuration file which will be loaded upon startup of the interface. If none is found in the current directory, the file supplied in the original distribution is used. This allows one to have multiple CHEMSEP.CNF files in different directories, which automatically configure the interface to (a) specific problem(s).

The CHEMSEP.SYN is a file containing synonyms for over 1000 compounds. While searching for a special component name you can use synonyms if you have selected to do so in the options interface spreadsheet. You must select this file as your synonyms file. The synonym search does

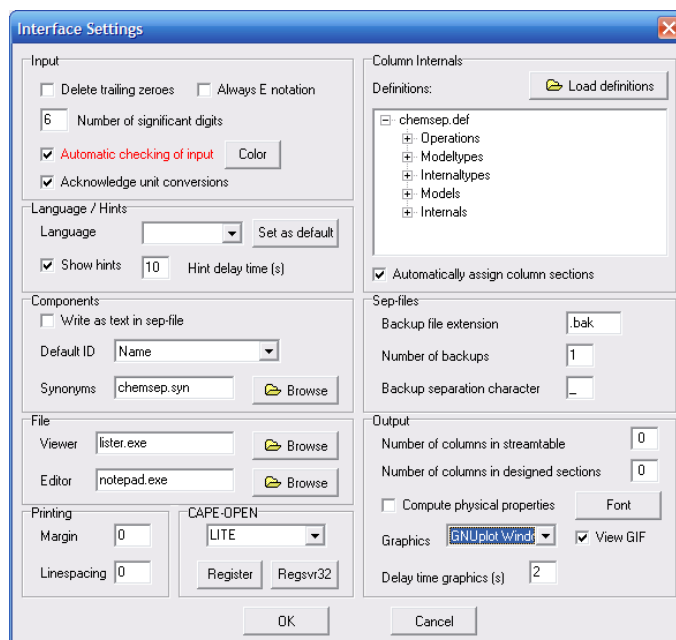


Figure 6.1: Interface settings.

not work while typing in a searchlist for a synonyms name. You will have to issue a search under the synonyms name! The synonyms file is an ASCII file you can modify to your needs.

The CHEMSEP.UDF file contains the definitions for the units of measure and the unit conversions in *ChemSep*. The first line must have the number of following lines with on each a unit definition. Such a unit definition consists of 15 characters (from column 1) with the unit abbreviation (take care, these are case sensitive!) followed by 15 characters (from column 16) with the full unit name (not case sensitive). Then, from column 31 the offset-factor (fo) and multiplication factor (fm) come and finally the reference unit. The conversion is done according the following formula:

$$\text{Number (in Reference units)} = \text{fm} * (\text{Number (in Units)} - \text{fo})$$

For example $22^{\circ}\text{C} = 1.0(22 - -273.15) = 295.15\text{K}$. You can inspect ASCII text files with *ChemSep*'s file viewer (F7) or make simple modifications with edit-file. However, we strongly suggest you do not change the original data files that come with *ChemSep*. We carefully selected the data in these files and other users might use your changed data and obtain erroneous results. We urge you first to copy the file to another name before you change or add anything to these files. Errors in the unit conversion can be very frustrating to find so it is good to check some results if you have changed or added a unit definition. To discourage people to make changes, CHEMSEP.UDF is set to be a read-only file!

6.3 Tools and Tool Configuration

ChemSep allows the configuration of the tools menu to include any auxiliary tool to be directly called from within the interface. Likewise the toolbar is fully configurable. You can move actions down or up the list and any *ChemSep* menu option with an icon can be added or removed from the toolbar. Fig. 6.2 shows the tools and toolbar configuration.

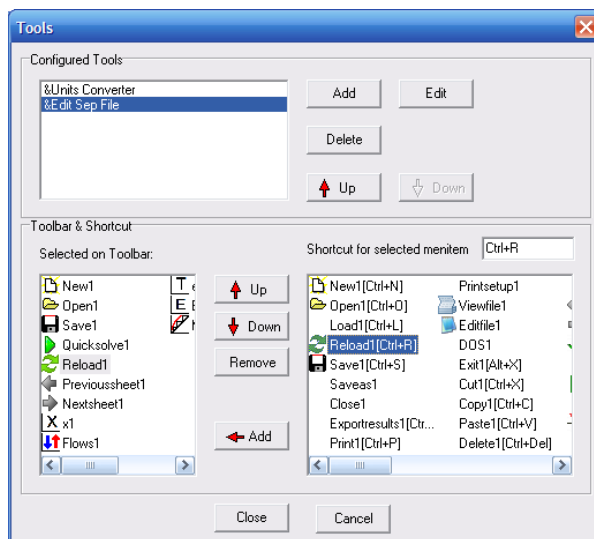


Figure 6.2: Tools.

Each menu item can have a shortcut key, which can be edited to personalize the GUI to your needs. Items frequently used could be programmed as function keys. Note that no checking is done on multiple entries of the same shortcut key. The GUI will only activate the first menu item it finds for such a shortcut.

External tools can be extremely handy for processing the files *ChemSep* creates as well as to start utility programs (plotting tools, editors, etc.). By default *ChemSep* configures the separate unit conversion tool and notepad as external tools. The latter is set to load the sep-file with the specifications. Each external tool appears as a menu item under the tools menu. Tools can be added, edited, moved, or deleted.

The editing of the configuration for an external tool is shown in Fig. 6.3. Each tool has a title that is used in *ChemSep*'s Tools menu and it must be associated with an external executable file. The user can specify a shortcut key if so desired. Also the directory where the tool should be started from can be specified. If omitted, the starting directory equals that of the sep-file. Finally a list of arguments can be specified. For this argument list macro's can be inserted to fill in the name of the currently loaded sep-file, or a specific path.

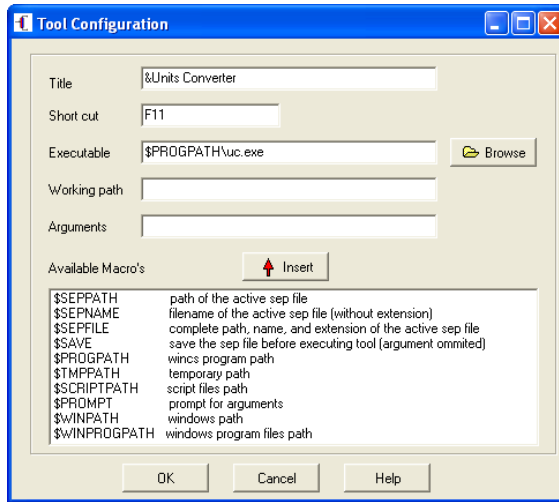


Figure 6.3: Tool Configuration of Tools.

The use of the \$\$SAVE macro can cause the interface to save the current problem prior to the execution of the tool, and the \$\$LOAD macro can cause the interface to load the sep file again after the executable has finished. the \$\$PROMPT macro causes a window to popup and ask the user for input which will be inserted as arguments at the position of the macro.

6.4 Design Mode Settings

Tray design is done using many default values for the parameters that determine the tray layout, all these values are set in the design mode. However, they also can be loaded from a configuration file that resides in the directory where the sep file resides. The latter allows designs tailored to specific operations (distillation or absorption, for example). The nonequilibrium design mode checks the existence of a local TDESIGN.DEF definitions file. If it is there, the tray design parameters are read from this file in a fixed format. TDESIGN.DEF files can be generated and altered by selecting "Tools — Design Mode Settings — Save" option, see Fig. 6.4. The settings of different DEF files can also be loaded into the interface.

Here we'll briefly discuss each of the screens that define the settings for the built-in design method for trays. Care must be taken in setting the design parameters as the layout design process is an iterative process which needs termination with a useful design in order to guarantee the convergence of the simulation problem.

Each TDESIGN.DEF has a general description line. The tray design algorithm adapts the layout

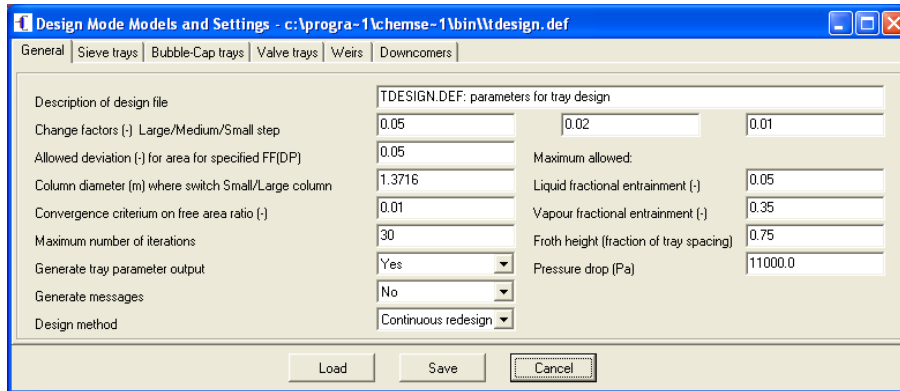


Figure 6.4: General design mode settings.

in steps of certain sizes to obtain either a certain fraction of flood or the pressure drop to be less than a certain maximum. Three different step sizes are used: large, medium, and small. Each are defined as a fractional stepsize. The stepsizes applied to the tray layout and the required number of iterations for the design method to come to an acceptable tray design are correlated: the smaller the steps, the more iterations are required.

The simulator only triggers a re-design when the flows on a tray change by a certain amount. Basically this allows the design to deviate from the selected fraction of flood or maximum pressure drop. By default this is 5%. Be careful not to set this allowed deviation too small otherwise the method will always exit after the maximum number of iterations is reached with a layout that is most likely to be unbalanced. The re-design can be done inbetween each iteration or, alternatively, only after the simulation has converged. The default method is the "continuous redesign" as this provides the best performance/convergence. If a simulation does not converge in design mode, switching this to the discontinuous redesign mode might help.

The design method varies the column diameter, active area (and thus implicitly the downcomer area), open ("free") area ratio, weir height, and tray spacing to meet the specified fraction of flood as well as a minimum turn down ratio. It uses a double iteration loop with the free area ratio in the inner loop, using a convergence criterion. If the design method doesn't converge this criterion can be loosened. In addition to the targetted fraction of flood some other limits must be met: a maximum liquid fractional entrainment, a maximum fractional vapor entrainment, a maximum ratio of froth height over tray spacing (jet flood), and a maximum allowed pressure drop over the tray. To prevent the design mode from hanging in an endless loop only a certain maximum number of iterations is allowed.

For bubble cap and valve tray columns two different default settings are used for the bubble caps and valves: one for small columns, another for large columns. The column diameter where this switch is made can be set as well. Various switches can be set: whether to generate

tray parameter output and to generate warnings and error messages

Capacity Factor (CF) method	Ogboja & Kuye		
Tray spacing (m), Lower/Default/Upper	0.1	0.5	2.0
Free area ratio (-), Lower/Default/Upper	0.05	0.15	0.20
Hole diameter (m), Lower/Default/Upper	0.0025	0.0047625	0.1
Tray thickness (m), Lower/Default/Upper	0.0005	0.00254	0.01

Figure 6.5: Sieve tray design settings.

Several correlations for predicting sieve tray capacity can be selected: Treybal, Ward, Economopolous, Lygeros and Magoulas, or Ogboja and Kuye. The latter is the default. Lower limits, default values, and upper limits are set for the sieve tray hole diameter, tray thickness, tray spacing, and the free area ratio can be given to provide the most suitable starting point of the tray design and to adhere to company guidelines.

Capacity Factor (CF) method	Ogboja & Kuye		
Disk diameter (m) where we switch from small to large caps	0.0889		
Disk diameter (m), Default for Small/Large column	0.0762	0.1016	
Hole diameter (m), Lower/Default/Upper	0.6	0.7	0.71
Slot height (m), Lower/Upper	0.0127	0.0381	
Slot height (m), Default for Small/Large cap	0.0254	0.03175	
Skirt height (m), Lower/Default/Upper	0.0127	0.0254	0.0381
Pitch as fraction (-) of disk diameter Lower/Default/Upper	1.25	1.25	0.5

Figure 6.6: Bubble-cap tray design settings.

As tray capacity correlation only the Ogboja and Kuye method can be used bubble-cap trays. We have a different default bubble cap disk diameter in small and large columns a value where we switch from small to large bubble-caps. Furthermore the user can specify a lower limit, default value, and upper limit can be set for the bubble cap hole diameter, bubble cap skirt height, slot height, and the pitch between caps.

There are three correlations for predicting valve tray capacity: Koch, Glitsch (default) and Ogboja and Kuye. Defaults can be given for the coefficients K for a closed and open valve, the valve eddy loss coefficient, the valve weight ratio legless/with legs, the default valve density, the default valve thickness, and the default valve diameter (for both small and large columns).

For the tray weirs a lower limit, default value, and upper limit can be set for the weir height. The weir height is also limited to be less than a certain fraction of the tray spacing. The design method employs a maximum liquid weir load to determine the number of passes on a tray. Liquid flow passes are added until the calculated liquid load is less than this limit. For northed

General	Sieve trays	Bubble-Cap trays	Valve trays	Weirs	Downcomers
Capacity Factor (CF) method	Glitsch				
Default diameter (m), Default for Small/Large column	0.0254	0.0508			
Default thickness (m)	0.003				
Default density (kg/m ³)	7849				
Default weight ratio (-) legless/with legs	1.29				
Default eddy loss coefficient (-)	1.3				
Default valve open K (-)	0.448				
Default valve closed K (-)	3.077				

Figure 6.7: Valve tray design settings.

General	Sieve trays	Bubble-Cap trays	Valve trays	Weirs	Downcomers
Weir height (m), Lower/Default/Upper	0.005	0.0508	0.1		
Maximum weir load (US-Gal/min/ft)	144.0				
Maximum weir height as fraction (-) of trayspacing	0.2				
Weir serration angle (rad)	0.7854				
Weir serration depth as fraction (-) of weir height	0.3333				
Weir notch depth as fraction (-) of weir height	0.3333				
Weir circular diameter as fraction (-) of weir length	0.9				

Figure 6.8: Tray weirs design settings.

weirs a default notch depth can be specified and for serrated weirs a default serration depth and serration angle. For circular weirs a diameter as fraction of the weir length can be defined.

General	Sieve trays	Bubble-Cap trays	Valve trays	Weirs	Downcomers
Downcomer area method	Glitsch				
Downcomer velocity method	Glitsch				
Average liquid fraction (-)	0.5				
Seal (m), applied when weir < clearance	0.0127				
Minimum residence time downcomer (s)	1.5				
Clearance (m) Lower/Default/Upper	0.01	0.0381	0.1		

Figure 6.9: Downcomer design settings.

Downcomer area is sized according to Koch or Glitsch rules or is taken as a fixed 12% of the cross-sectional area. A maximum downcomer velocity check according to Koch or Glitsch can be selected, or none. The downcomer average liquid fraction is specified as well as the minimum downcomer residence time. When the weir height is less than the downcomer clearance, the downcomer clearance is adjusted to maintain a certain downcomer liquid seal. A lower limit,

default value, and upper limit can be set for the downcomer clearance.

6.5 ChemSep Internals and Model Development

ChemSep allows the development of models for the proper nonequilibrium simulation of separation processes via dynamically linked libraries (DLL's). The user is free to program these models in C or in Fortran and can specify up to 50 model parameters which can consist of model switches, for example type selections, or model coefficients. Models can be written to compute mass transfer rates, pressure drops, liquid mixing and holdup, as well as entrainment or backmixing. Similarly, the user can generate new types of column internals that are completely compatible with the existing column internal types. Up to 50 hardware parameters can be supplied to describe the column internal. A design method can be added (just like any model) that enables the engineer to simulate non-existing new columns still to be designed. The following types of models can be added:

- Mass Transfer Coefficient (MTC)
- Pressure Drop (PD)
- Liquid Holdup (HL)
- Entrainment/backmixing (ENTR/BACK)
- Design method (DSGN)
- Heat transfer Coefficient (HTC)

The interface offers a step by step process that guides the user through the process of:

1. defining the model/internal parameters
2. writing the model code,
3. compiling the model code,
4. writing code to test and validate the model,
5. compiling the test-code,
6. running the test-code to test in a debugger,
7. compiling the code as a DLL,
8. compiling the test-code to run with the DLL,
9. running the test-code with the DLL in a debugger,
10. enable the model definitions in the interface,
11. load an example test sep-file that uses the model,
12. running the example test sep-file with/without running the debugger.

The writing of the DLL is simplified by use of templates which can be used to create a new model or internal. The C/Fortran templates contain the required routines as empty routines. The actual code only needs to be added inside these routines. The develop environment can contain several models/internals next to eachother that can be selected by means of a selector, see Figure 6.10.

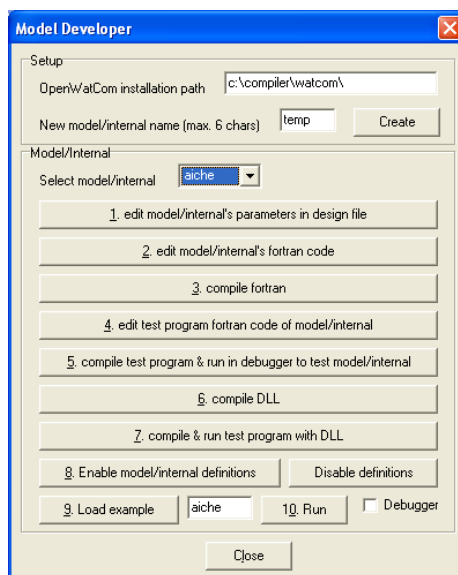


Figure 6.10: ChemSep Model Development.

We will discuss here how define model/internal parameters, what routines are required (and what their function), and then illustrate the model development with several examples.

6.5.1 Model Definitions Files

ChemSep reads a definitions file (CHEMSEP.DEF) at startup, where models for the mass transfer coefficients, pressure drop, flow models, entrainment, and holdup are defined. This alleviates the problem of having to adapt the *ChemSep* interface upon any addition of or modification to a model. The definition file also includes the definitions of any non-standard type of internal. In case no definitions file is found, the nonequilibrium part of *ChemSep* is disabled.

The definitions file must start with '[ChemSep Definitions]' followed by a line which defines the version (the current version is 3). Only after these two lines content may appear. Everything after a '#' is regarded as comment. Comments may appear on any content line, and may also be appended after any content. There are *five* different definitions types that are read from the defi-

inition file: **[InternalType]**, **[Operation]**, **[ModelType]**, **[Internal]**, and **[Model]**. Other entries are ignored. The five different types of definitions may be mixed throughout the definitions file. A definition may have no blank lines in it as *entries are ended by a blank line!*

The **[InternalType]** defines the nature of an internal, that is whether it is continuous or discrete. The **[Operation]** defines the nature of the operation, that is whether we are contacting a vapor and a liquid (VL) or two liquids (LL). The **[ModelType]** defines the type of models, that is whether the model describes mass transfer coefficients, pressure drops, etc. Each of these has the following fields: ID, Name, and Short, for example:

```
[Operation]
ID=1
Name=Vapor-Liquid
Short=VL
```

The **[Internal]** definitions link these types together: its 'Type' must be one of the defined internal types, and its 'Operation' one of the defined operation types. It also defines which 'Models' need to be specified for the internal as well as which layout 'Parameters' the internal has. These parameters and models are separated by semicolons (;) or comma's (.). The parameters might include a units string in parentheses and a default value (with different units) after a colon (:). Note that multiple 'Parameters' entries are allowed but that its total length is limited to 255 characters (these rules also apply for the 'Models' and 'Internals' entries).

A parameter entry may also contain a list in square brackets, separated by vertical lines. In such cases the interface will prompt the user to select the parameter from the list of items. The first item in the list will result in the value 1, the second in 2, etc. A default item may also be specified using the colon and the numeric value of the parameter. For the model entries, either ID numbers or short notation may be used, as long as they are defined ModelTypes. For example the definition for a baffle tray may be:

```
[Internal]
ID=10
Name=Baffle tray
Type=Discrete
Operation=VL
Short=Baffle
Models=MTC,PD,VF,LF,DSGN
Parameters=Column diameter (m):120cm;Baffle area (%):50
Parameters=Baffle spacing (m)
Parameters=Baffle type[Disk & donut|Single flow|Multiflow]:2
# default baffle type = Single
```

The **[Model]** definitions has instead of a 'Models' entry the 'Internals' entry, where the types of

internals for which the model can be used are defined. It also has the 'Parameters' where model parameters can be defined. For the definition of a baffle tray above the parameters are a column diameter (with meters as units and a default value of 1.2m), the baffle area (as percentage, default 50%), the baffle spacing (in meters, without any default), and the baffle type (with a pick list between 3 different types where the default is the single flow). Internals parameters can be stored and reloaded to/from test files (*.inp) to enable the selection of standard types of trays and/or packings.

The models that are required for the baffle tray are a mass transfer coefficient model (MTC), a pressure drop model (PD), models for the vapor and liquid flow (VF and LF), as well as a design method (DSGN). Note that the design method is also handled as a model. Thus, more than one design method can be implemented. A sample design method definition is:

```
[Model]
ID=1
Name=Fraction of flood
Type=DSGN
Operation=VL
Short=%flood
Internals=Bubble cap;Sieve;Valve
Parameters=Fraction of flooding:75%;Fraction of weeping:70%
```

Model parameters can be loaded from model parameter files (*.mop). Note that a model definition may also be repeated (with the same id, name, type, and operation) so to add another internaltype to the list to which the model applies. Short fields are optional, and have a maximum length of ten characters, used for displaying selected models etc. ID fields associate a unique number (in 1 ...255) to the definition. Only for the internal and model definitions non-unique numbers are allowed. When the interface reads the definitions file it uses the Short descriptions to assign the ID's for the Operations, InternalTypes and ModelTypes, see Table 6.1 (the Short descriptions used by the interface are in parenthesis). Thus, you will have to use these Short descriptions but are free to change the ID numbers or names.

There are standard definitions for internal types that are built into the interface. These do not require the parameters entry to be specified. If they are specified, they -override- the internal built-in types. For example, the definition of the bubble cap tray internal is:

```
[Internal]
ID=1
Name=Bubble cap tray
Type=Discrete
Short=Bubble cap
Operation=VL
Models=MTC, PD, VF, LF, ENTR
```

Table 6.1: Interface Assigned Types

Operation	InternalType	ModelType
Vapor-Liquid (VL) Liquid-Liquid (LL)	Discrete (DIT) Continuous (CIT)	Mass transfer coefficient (MTC) Pressure drop (PD) Vapor flow (VF) Liquid flow (LF) Entrainment (ENTR) Holdup (HOLD) Light liquid flow (LLF) Heavy liquid flow (HLF) Backmixing (BACK) Design method (DSGN)

The built-in types are listed in Table 6.2. This table also lists defined vapor and liquid flow models and models for entrainment.

Table 6.2: Assigned Internals and Vapor flow, Liquid flow, and Entrainment Models

Internal	Vapor Flow / Light liquid	Liquid Flow / Heavy liquid	Entrainment / Backmixing
Bubble cap tray (1)	Mixed (1)	Mixed (1)	None (1)
Sieve tray (2)	Plug flow (2)	Plug flow (2)	Estimated (2)
Valve tray (3)			
Dumped packing (4)			
Structured packing (5)			
Equilibrium stage (6)			
RDC compartment (7)			
Spray column stage (8)			

An example of a model definition is:

```
[Model]
ID=1
Name=AIChE
Type=MTC
Operation=VL
Internals=Bubble cap,Sieve tray,Valve tray
```

Assigned models for Mass Transfer Coefficient and Pressure Drop models are listed in Table 6.3. Model parameters may be read from model parameter libraries (*.mop).

Table 6.3: Assigned MTC and PD Models

Mass Transfer Coefficient	Pressure Drop
AIChE (1)	Fixed (1)
Chan Fair (2)	Estimated (2)
Zuiderweg (3)	Ludwig 1979 (3)
Hughmark (4)	Leva GPDC (4)
Harris (5)	Billet-Schultes 1992 (5)
Onda et al. 1968 (6)	Bravo-Rocha-Fair 1986 (6)
Bravo-Fair 1982 (7)	Stichlmair-Bravo-Fair 1989 (7)
Bravo-Rocha-Fair 1985 (8)	Bravo-Rocha-Fair 1992 (8)
Bubble-Jet (9)	
Bravo-Rocha-Fair 1992 (10)	
Billet-Schultes 1992 (11)	
Nawrocki et al. 1991 (12)	
Chen-Chuang (13)	
Handlos-Baron-Treybal (14)	
Seibert-Fair (15)	
Kronig-Brink-Rowe (16)	
Rose-Kintner-Garner-Tayeban (17)	
Sherwood (20)	

6.5.2 Required DLL Routines

The DLL used by the simulator is always called MODINT.DLL and the development environment copies the selected DLL in and out of the binary executables directory. The DLL must contain the following routines:

UsrFap Appends output to unit to specified filename

UsrFcl Close output to unit

UsrDin Assigns default column internal parameters and sets the number of model parameters and defines which represent the internals diameter (D) and height (H)

UsrDot Outputs the column internal's parameters

UsrDes Designs a column internal type

UsrRed Re-designs a section of one specific column internal type

UsrAt Computes internal interfacial area and residence times

UsrMTC Computes mass transfer coefficients (MTCs) for a column internal/model

UsrRep Report/calculate pressure drop, entrainment, fraction of flooding/weeping

UsrHTC Computes heat transfer coefficients (HTCs) for a column internal/model

The first two routines, *UsrFap* and *UsrFcl*, are used for opening and closing file output and do not need to be modified. The *UsrDin*, *UsrDot*, *UsrDes*, and *UsrRed* routines are only needed for programming a new type of column internal. As the simulator cannot know what the internal parameters represent the user must supply his/her own routines to set default values, how the parameters are named in the sep-file, and also how the internal is to be designed as well as how different internals sitting in one column section are to be rationalized into one design (i.e. to have the same diameter and parameter values). The *UsrAt* and the *UsrMTC* routines can compute interfacial area and residence times as well as the mass transfer rates. The *UsrRep* routine can compute pressure drop, liquid holdup, or entrainment and expresses the hydraulic behavior of a column internal in terms of fraction of flood and fraction of weep (for trays. For other internals this represents of lowest operation without impairing the mass transfer performanc of the device).

The use of these routines is differently. To make a new mass transfer coefficient (MTC) model for existing column internals requires just implementation of the *UsrAt* and the *UsrMTC* routines. Below we discuss such an example with how we can program the AIChE MTC model which can employ different models for the liquid on the tray. To make a new pressure drop (PD) model we just need to implement a new *UsrRep* routine. We will show this in an example that models the pressure drop of Sulzer I-Rings. To make a new internal we need all routines. We show this by implementing baffle trays with design and performance models as published by Fair *et al.*.

6.5.3 AIChE MTC model

The model defintions for our new AIChE model are:

```
# BEGIN INTERNALS TEMPLATE
# MTC (new definition)
[Model]
ID=150
Name=AIChE model
Type=MTC
Operation=VL
Short=MTC
Internals=Bubble cap;Sieve;Valve
Parameters=Hydraulic model[Original|Bennett]
```



```

parameters=Tray type[Bubble-Cap|Sieve|Fit]
Parameters=Debug[Off|On]
Parameters=Cv:1;Cl:1
# END TEMPLATE

```

The lines starting with # are comment lines. In this case we use them to mark the start and end of the definitions template. Here we defined our new model of type MTC, to be used for columns with vapor and liquid phases, and that can be used for the existing column internals Bubble-Cap, Sieve, and Valve trays. There are 5 model parameters: the hydraulic model, the type of tray, a debug flag, and two model coefficients, Cv and Cl. The hydraulic model is a selector with two choices: Original and Bennett. The tray type is a selector with three choices and the debug flag a selector with two choices (off/on). The model coefficients have the default values of unity.

The *UsrAt* routine is used here to compute the interfacial area used in the mass transfer calculations. we will discuss it in detail here. it starts with the call declaration that may not be changed:

```

      subroutine UsrAt (InType, iMTC, j, n, m, Params, PMTC,
+                    Vmf, Lmf, RhoV, RhoL, VisV, VisL, MolwtV, MolwtL, Sigma,
+                    Atot, Avap, Aliq, Tvap, Tliq, iColD, iColH)
      integer InType, iMTC, j, n, m, iColD, iColH
      double precision Params (n, m), PMTC (n, m),
+                    Vmf, Lmf, RhoV, RhoL, VisV, VisL, MolwtV, MolwtL, Sigma,
+                    Atot, Avap, Aliq, Tvap, Tliq
c -----

```

Next we declare the local variables and common blocks we are going to use:

```

c   Local variables:
      double precision Dc, Ts, Z, Active, Ahol, Pitch, W, hw, free, Xarea, Vol,
+                    Vm, V, Uv, Lm, L, Ul, Ql, cc, Flv,
+                    Aint, alfa, Hcl, Eps, Hf,
+                    psi, HclHZ, Term, C, aeB, HclB, aeG, HclG
c   Common block:
      double precision A_t, A_l, A_v, T_v, T_l, Fs, Qlw
      common /areat/   A_t, A_l, A_v, T_v, T_l, Fs, Qlw
c -----

```

The routine starts with making sure we execute the model only for the correct types of column internals (bubble cap, sieve, and valve trays have the identifiers 1, 2, and 3) and the right mass

transfer model (our id is 150). If we do not filter for the right id's we might overwrite calculations by other models! To signal we handled the call we set the iColD and iColH variables that identify which parameters are the diameter and height of the column's internal. If we do not do this the simulator halts the execution.

```
c
  if (InType .ge. 1 .and. InType .le. 3) then
    if (iMTC .eq. 150) then
c
      set {iColD,iColH} to signal we handled call
c
c
      iColD = 1
      iColH = 2
```

Next we convert the tray dimensions we need for the computations. We take them from the *Params* array and put them into local variables. The tray dimensions are ordered by the parameters list for the sieve trays in the chemsep.def file. Strictly speaking this conversion is not necessary as we can also access the tray dimension by directly referring to the *Params* array but this technique helps to prevent programming mistakes with the array indices. For example if the order of the tray parameters were to change it is easier to also propagate such changes when we concentrate the conversion into local variables in blocks like this.

```
c
c      Get tray layout parameters:
c      (Params from Sieve tray definition in chemsep.def):
c
c      Parameters=Column diameter (m)
c      Parameters=Tray spacing (m)
c      Parameters=Number of flow passes
c      Parameters=Liquid flow path length (m)
c      Parameters=Active area (m2)
c      Parameters=Total hole area (m2)
c      Parameters=Downcomer area (m2)
c      Parameters=Hole diameter (m)
c      Parameters=Hole pitch (m)
c      Parameters=Weir length (m)
c      Parameters=Weir height (m)
c      Parameters=Weir type[Segmental|Notched|Serrated|Circular]
c      Parameters=Notch depth/Weir diameter (m)
c      Parameters=Serration angle (rad)
c      Parameters=Downcomer clearance (m)
c      Parameters=Deck thickness (m)
```

```

c      Parameters=Downcomer sloping (%)
c
      Dc      = ParmS(j,iColD)
      Ts      = ParmS(j,iColH)
      Z       = ParmS(j,4)
      Active  = ParmS(j,5)
      Ahol   = ParmS(j,6)
      Pitch  = ParmS(j,9)
      W      = ParmS(j,10)
      hw     = ParmS(j,11)

```

Next we start to compute some tray geometry values such as the cross-sectional area and tray volume, mass and superficial velocities, and vapor load factor (cc) and flow parameter (Flv).

```

c
c      computed tray geometry
c
      free   = Ahol / Active
      Xarea  = 3.141d0 * Dc**2 / 4.0d0
      Vol    = Ts * Xarea
c
c      compute mass, mass/area and superficial velocities, liquid volumetric flow
c
      Vm     = Vmf * MolwtV
      V      = Vm / Active
      Uv     = V / RhoV
      Lm     = Lmf * MolwtL
      L      = Lm / Active
      Ul     = L / RhoL
      Ql     = Lm / RhoL
      Qlw    = Ql / W
c
c      C and F factor, flow parameter
c
      cc     = Uv * dsqrt(RhoV / (RhoL-RHoV))
      Fs     = dsqrt(RhoV) * Uv
      Flv    = (Lm / Vm) * dsqrt(RhoV / RhoL)

```

Next we compute the specific interfacial area (m^2/m^3) using a correlation by Zuideweg and the liquid holdup and residence time by either the Bennett or the Colwell/Gerster models, dependent on selection set by the hydraulic model parameter, PMTC(j,1):

```

c
c      Zuiderweg correlation for clear liquid (m) and interfacial area (m2/m3)
c
Psi    = Qlw / Uv * dsqrt (RhoL / RhoV)
HclHZ  = 0.6d0 * (Psi * Pitch * hw * hw) ** 0.25D0
Term   = Uv * Uv * RhoV * HclHZ * Flv / sigma
Aint   = 43.d0 / free**0.3d0 * Term**0.53d0
c
      if (PMTC(j,1) .eq. 2.d0) then
c
c          Bennett's liquid holdup model
c
C      = 0.5d0 + 0.438d0 * dexp( -137.8d0 * hw )
aeB   = dexp( -12.55d0 * cc**0.91d0 )
HclB  = aeB*(hw + C * (Qlw/aeB)**0.67d0)
alfa  = aeB
Hcl   = HclB
Tliq  = Hcl * Active / Ql
c
      else
c
c          Gerster et al. liquid height
c
HclG  = 0.040d0 + 0.29d0*hw - 0.0135d0*Fs + 2.45d0*QlW
c
c          Colwell model for liquid density
c
aeG   = cc**2 / (9.81d0 * hclG)
aeG   = 12.6d0 * aeG**0.4d0 / free**0.25d0
aeG   = 1.d0/ (1.d0+aeG)
alfa  = aeG
Hcl   = HclG
Tliq  = Hcl * Z * W / Ql
      endif
c
c      Vapour holdup fraction (eps) and froth height, residence times
c      based on the Selected hydraulic model
c
Eps   = 1.d0 - Alfa
Hf    = Hcl / Alfa
Tvap  = Eps * Hf / Uv
c

```

Next we compute the specific interfacial area per volume liquid and per volume vapor as well as the total interfacial area. The results are stored in a common block so we can pick them up

again in the mass transfer rates calculation. Finally, dependent on the debug flag PMTC(j,3) we write out some debug messages to standard output (STDOUT = unit 6) if this is requested:

```

c
c      Interfacial Area's
c
c      Aliq = Aint / (Alfa * Hf)
c      Avap = Aint / (Eps * Hf)
c      Atot = Aint * Active
c
c      Store area's & residence times in common block for MTC calcs
c
c      A_t = Atot
c      A_l = Aliq
c      A_v = Avap
c      T_v = Tvap
c      T_l = Tliq
c
c      Debug messages
c
c      if (PMTC(j,3) .gt. 1.d0) then
c          write (6,*) 'Area/residence times AICHe ',j
c          write (6,203) Atot,' Atot'
c          write (6,203) Avap,' Avap'
c          write (6,203) Aliq,' Aliq'
c          write (6,203) Tvap,' Tvap'
c          write (6,203) Tliq,' Tliq'
c          write (6,203) Hf,' Hf'
203      format (1H , 1PE12.5,(A))
c          endif
c      endif
c
c      return
c      end

```

For the routine that computes the actual mass transfer coefficients we do the start with the declaration of the routine, the local variables and the common block, and then verify again that the calculations are to be done only for the correct internals and models type (on the basis of the id's in InType and iMTC). We set the iColD and iColH to signal we compute the MTC's and obtain the internals dimensions we need from the Params array:

```

      subroutine UsrMTC(InType, iMTC, j, n, m, ParmS, PMTC,
+                    Vmf, Lmf, Dv, Dl, RhoV, RhoL, VisV, VisL, ScV, ScL,
+                    CpV, CpL, ConV, ConL, MolwtV, MolwtL, Sigma,
+                    KKV, KKL, iColD, iColH)
      integer InType, iMTC, j, n, m, iColD, iColH
      double precision ParmS(n, m), PMTC(n, m),
+                    Vmf, Lmf, Dv, Dl, RhoV, RhoL, VisV, VisL, ScV, ScL,
+                    CpV, CpL, ConV, ConL, MolwtV, MolwtL, Sigma,
+                    KKV, KKL
c -----
c Local variables:
      double precision Xarea, Vol, Uv, ql, cc, C, ae, hl, Fss, tll, tv
+                    Dc, Ts, hw, W, Z, Ahol, Activ
c Common block:
      double precision A_t, A_l, A_v, T_v, T_l, Fs, Qlw
      common /areat/   A_t, A_l, A_v, T_v, T_l, Fs, Qlw
c -----
c
      if (InType .ge. 1 .and. InType .le. 3) then
         if (iMTC .eq. 150) then
c
c           set {iColD, iColH} to signal we handled call
c
c           iColD = 1
c           iColH = 2
c
c           hw = ParmS(j, 11)

```

Next we compute the number of transfer units (NTU) from the AIChE correlations. Note that a different equation is used for the liquid MTC dependent on the tray type parameter, PMTC(j,2). The number of transfer units are scaled by the model parameters PMTC(j,4) (for the vapor) and PMTC(j,5) (for the liquid). Mass transfer coefficients are computed by dividing the NTU's by the product of the interfacial area and the residence time (of each particular phase). Again, debug messages are written when requested by the debug flag parameter, PMTC(j,3):

```

c
c   Number of Transfer Units: NTUv, NTUl
c
      KKV = (0.77d0 + 4.57d0*hw -0.238d0*Fs + 104.8d0*Qlw)
      KKV = KKV / dsqrt(ScV)
      if      (PMTC(j,2) .eq. 1.d0) then
c         Bubble-Cap
         KKL = 20320d0*dsqrt(Dl)*(0.21d0*Fs+0.15d0)*t_l
      else if (PMTC(j,2) .eq. 2.d0) then

```

```

c      Sieve
      KKL = 19700d0*dsqrt(Dl)*(0.4d0*Fs+0.17d0)*t_1
else if (PMTC(j,2) .eq. 3.d0) then
c      Fit with coefficients
      KKV = PMTC(j,4) * dsqrt(T_v/ScV)
      KKL = PMTC(j,5) * dsqrt(T_v/ScL)
endif

c
c      Mass Transfer Coefficients: Kv, Kl in (m/s)
c
      KKV = KKV / (A_v * T_v)
      KKL = KKL / (A_l * T_l)

c
c      Debug messages
c
      if (PMTC(j,3) .gt. 1.d0) then
        write (6,*) 'MTC AIChE ', j
        write (6,203) KKV,' kv'
        write (6,203) KKL,' kl'
203      format (1H , 1PE12.5, (A))
      endif

c
      endif
endif

c
      return
end

```

The routine *UsrHTC* from the template we do not need to adapt as we do not want to change the heat transfer coefficients *ChemSep* computes itself from the average mass transfer coefficients.

When running through the debugger:

6.5.4 I-Rings Pressure Drop Model

-here the model must come-

6.5.5 Baffle Tray Internal

In this example we define a complete new column internal type, the baffle tray. We start defining a new internal. Watch out to use a unique identifier number for the baffle tray (here we used 20) that's not been used before. Indicate what type of internal it is (a discrete tray internal for

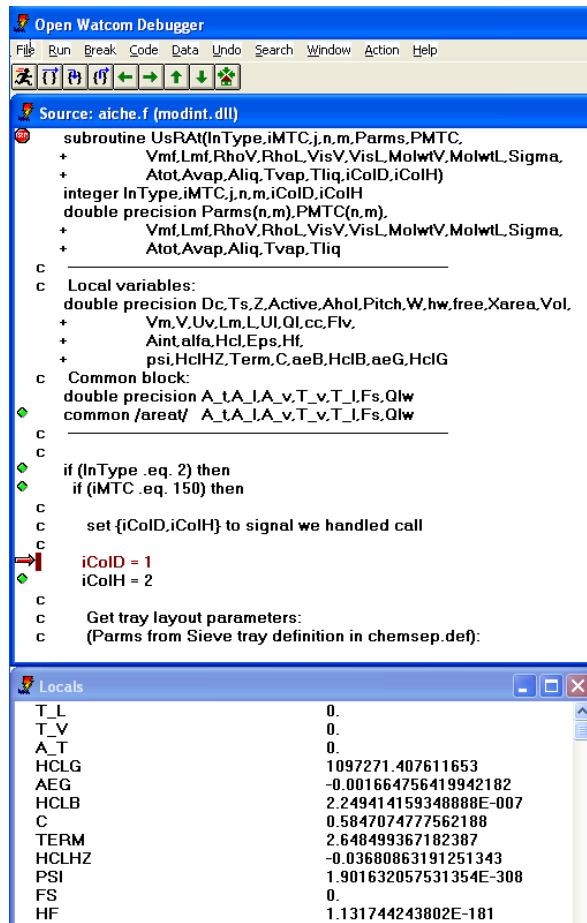


Figure 6.11: Stepping through the code for the AICHe MTC model in the debugger allows you to see each calculation. All the variables can be inspected in the "Locals" watch window..

vapor and liquid operation only) as well as what kind of Models the internal requires: mass transfer coefficients (MTC), pressure drop (PD), vapor and liquid flow (VF, LF) and design mode (DSGN) models.

```
# BAFFLE-v1.0-BaffleTrays
# implementation Baffle trays
```



```
# INTERNAL (new definition)
[Internal]
ID=20
Name=Baffle tray
Type=Discrete
Operation=VL
Short=Baffle
Models=MTC,PD,VF,LF,DSGN
Parameters=Column diameter (m):1;Baffle spacing (m):0.5
Parameters=Baffle area (%):50;Baffle flow type[Single|Double|Multi]:1
Parameters=Debug[Off|On]:1
```

Then for each of the required models, specify these:

```
# PD (add to existing definition)
[Model]
ID=1
Name=Fixed
Type=PD
Operation=VL
Internals=Baffle
```

```
# VF (add to existing definition)
[Model]
ID=1
Name=Mixed flow
Type=VF
Operation=VL
Internals=Baffle
```

```
# LF (add to existing definition)
[Model]
ID=1
Name=Mixed flow
Type=LF
Operation=VL
Internals=Baffle
```

```
# DSGN (new definition)
[Model]
ID=20
Name=Fraction of flood
Type=DSGN
Operation=VL
```

```

Short=%flood
Internals=Baffle
Parameters=Fraction of flood (%):75
Parameters=Debug[Off|On]:1

# MTC (new definition)
[Model]
ID=210
Name=Baffle
Type=MTC
Operation=VL
Short=Baffle
Internals=Baffle
Parameters=C_v:0.0025;C_l:1.0
Parameters=Debug[Off|On|Extra|Wait]

# BAFFLE-v1.0-BaffleTrays

```

Note that here we have assigned pressure drop and flow models that use existing models (for example the fixed pressure drop model where the pressures are specified in the interface).

We then write our *UsrDin* routine that sets the number of parameters of the baffle tray as well as assigns default values (so to make it work out of the box — most of the time)

```

subroutine UsrDin(InType,nParms,ParmsD,j,n,m,iColD,iColH)
integer          InType, nParms, j, n, m, iColD, iColH
double precision ParmsD(n,m)
c -----
c include 'baffle.i'
c
c if (InType .eq. iTbaf1) then
c   nParms=5
c   Parameters:
c   Column diameter (m):1
c     iColD=jColD
c     ParmsD(j,jColD)=1.d0
c   Baffle spacing (m):0.6096
c     iColH=jColH
c     ParmsD(j,jColH)=0.6096d0
c   Baffle area (%):50
c     ParmsD(j,jArea)=0.5d0
c   Baffle flow type[Single|Double|Multi]:1
c     ParmsD(j,jFtype)=1.d0
c   Debug[Off|On]:1

```

```

        ParmSD(j, jDebug)=1.d0
    endif
    return
end

```

Note how we used an "include file" called 'baffle.i' where we defined constants such as the ID of the baffle tray, ITbafl, that we use in the code. We also defined constants jColD, jColH, jArea, jFtype, and jDebug to access the baffle tray parameters from the array with internal parameters. This makes it easier to keep track of things and to move the sequence of the baffle tray parameters (we'd only need to make changes in the include and definitions files). Next we must write an *UsrDot* routine that outputs the internal to our sep-files. This is done by writing a line with the tray type ID followed by a description and then an equal sign and the two numbers, the operation type and the number of parameters the new internal has. In our case the operation type is VL (=1) and we have five parameters that describe the tray. Thus "= 1 5" is appended to the description. Each next line contains a parameter value followed by a description (which is ignored but enhances the readability of the sep-file):

```

subroutine UsrDot(iUnit, InType, j, n, m, ParmS)
integer          iUnit, InType, j, n, m
double precision ParmS(n, m)
c -----
c include 'baffle.i'
c
c if (InType .eq. iTbafl) then
c   write "tray-type description = operation-type number-internal-parms"
   write (iUnit,201) InType, ' Discrete Baffle tray = 1 5'
   write (iUnit,203) ParmS(j, jColD), ' Column diameter (m)'
   write (iUnit,203) ParmS(j, jColH), ' Baffle spacing (m)'
   write (iUnit,203) ParmS(j, jArea), ' Baffle area (-)'
   write (iUnit,203) ParmS(j, jFtype), ' Baffle flow type'
   write (iUnit,203) ParmS(j, jDebug), ' Debug'
201  format (1H , i5, 7X, (A))
203  format (1H , 1PE12.5, (A))
endif
return
end

subroutine UsrDes(InType, iDesgn, Dparms, j, n, m,
+              VmF, Lmf, RhoV, RhoL,
+              VisV, VisL, MolwtV, MolwtL,
+              Sigma, SF,
+              ParmS, iColD, iColH)

```

```

integer          InType, iDesgn, j,n,m,
+               iColD, iColH
double precision Vmf,    Lmf,    RhoV,    RhoL,
+               VisV,    VisL,    MolwtV, MolwtL,
+               Sigma, SF,
+               Dparms(n,m), Parms(n,m)
c -----
c include 'baffle.i'
c Local variables:
double precision foot, one, two, Hbdef, Csbdef, Abdef
parameter        (foot=0.3048d0, one=1.d0, two=2.d0)
parameter        (Hbdef=two*foot, Csbdef=0.65d0*foot, Abdef=0.5d0)
double precision Csbw,Sqrtrr,Uw,Ac
c
c   if (InType .eq. iTbaf1) then
c     if (iDesgn .eq. iDbaf1) then
c       iColD = jColD
c       iColH = jColH
c       Flow type must be single baffle
c       Parms(j,jFtype) = one
c       Set default open area if not set properly
c       if (Parms(j,jArea) .lt. 0.1d0 .or.
+         Parms(j,jArea) .gt. 0.9d0) then
c         Parms(j,jArea) = Abdef
c       endif
c       Set default baffle spacing to
c       if (Parms(j,jColH) .lt. 0.01d0 .or.
+         Parms(j,jColH) .gt. 1.0d1) then
c         Parms(j,jColH) = Hbdef
c       endif
c       Calculate vapor flooding velocity based on open area and
c       scaled by baffle spacing to the power one half
c       Csbw = Csbdef * ( Parms(j,jColH) / Hbdef )**0.5d0
c       Sqrtrr = (RhoV/(RhoL-RhoV))**0.5d0
c       Uw = Csbw / Sqrtrr
c       Derate vapor velocity with fraction of flood and system factor
c       if ((Dparms(j,jFF) .gt. 0.01d0) .and.
+         (Dparms(j,jFF) .le. one ) ) then
c         Uw = Dparms(j,jFF) * SF * Uw
c       else
c         Uw = 0.75d0 * SF * Uw
c       endif
c       Calculate open, cross sectional area, and column diameter
c       Ac = (Vmf * MolwtV / RhoV) / Uw
c       Ac = Ac / Parms(j,jArea)

```

```

       Parms(j,jColD) = sqrt(4.d0*Ac/3.141d0)
c      When Debugging show results
        if (Parms(j,jDebug) .eq. 2.d0) then
            write (io,*) 'Designing baffle tray ',j
            write (io,*) Dparms(j,jFF),' Fraction of flood @ Design'
            call UsrDot(io,InType,j,n,m,Parms)
        endif
    endif
endif
return
end

subroutine UsrRed(InType,n,m,jstart,jend,Parms,iFlow,iColD,iColH)
integer InType,n,m,jstart,jend,iFlow,iColD,iColH
double precision Parms(n,m)
c -----
c include 'baffle.i'
c Local variables:
integer          jd,jh,k
double precision dmax,hmax
c
c if (inType .eq. iTbaf1) then
c     find the stages with largest diameter and spacing
        j=jstart
        jd=j
        jh=j
        iColD=jColD
        iColH=jColH
        dmax=Parms(j,iColD)
        hmax=Parms(j,iColH)
        do j=jstart+1,jend
            if (Parms(j,jColD) .gt. dmax) then
                dmax=Parms(j,jColD)
                jd=j
            endif
            if (Parms(j,jColH) .gt. hmax) then
                hmax=Parms(j,jColH)
                jh=j
            endif
        enddo
c     set design to stage with largest diameter
        do j=jstart,jend
            do k=1,5
                Parms(j,k)=Parms(jd,k)
            enddo
        enddo
end

```

```

        enddo
    enddo
c    When Debugging show results
    if (Parms(jd,5) .eq. 2.d0) then
        write (io,*) 'Rationalizing baffle trays ',jstart,'-',jend
        call UsrDot(io,InType,jd,n,m,Parms)
    endif
endif
return
end

subroutine UsRAt(InType,iMTC,j,n,m,Parms,PMTC,
+             Vmf,Lmf,RhoV,RhoL,VisV,VisL,MolwtV,MolwtL,Sigma,
+             Atot,Avap,Aliq,Tvap,Tliq,iColD,iColH)
integer InType,iMTC,j,n,m,iColD,iColH
double precision Parms(n,m),PMTC(n,m),
+             Vmf,Lmf,RhoV,RhoL,VisV,VisL,MolwtV,MolwtL,Sigma,
+             Atot,Avap,Aliq,Tvap,Tliq
c -----
include 'baffle.i'
c Local variables:
double precision Aint,Xarea,Vol,Uv,Ul,Rev
c
if (InType .eq. iTbaf1) then
    if (iMTC .eq. iMbafl) then
c
        iColD = jColD
        iColH = jColH
c
        Aint = 25.d0
        Xarea = 3.141d0 * Parms(j,jColD)**2 / 4.0d0
        Vol = Parms(j,jColH) * Xarea
        Atot = Aint * Vol
        Avap = Aint
        Tvap = Vol * (RhoV/MolwtV) / Vmf
        Aliq = Aint
        Tliq = Vol * (RhoL/MolwtL) / Lmf
c
        if (PMTC(j,3) .gt. 1.d0) then
            write (io,*) 'Area/residence times Baffle tray ',j
            write (io,203) Atot,' Atot'
            write (io,203) Avap,' Avap'
            write (io,203) Aliq,' Aliq'
            write (io,203) Tvap,' Tvap'
        endif
    endif
endif

```

```

        write (io,203) Tliq,' Tliq'
203      format (1H , 1PE12.5, (A))
        endif
c
        endif
        endif
c
        return
        end

subroutine UsrMTC(InType,iMTC,j,n,m,Parms,PMTC,
+              Vmf,Lmf,Dv,Dl,RhoV,RhoL,VisV,VisL,ScV,ScL,
+              CpV,CpL,ConV,ConL,MolwtV,MolwtL,Sigma,
+              KKV,KKL,iColD,iColH)
integer InType,iMTC,j,n,m,iColD,iColH
double precision Parms(n,m),PMTC(n,m),
+              Vmf,Lmf,Dv,Dl,RhoV,RhoL,VisV,VisL,ScV,ScL,
+              CpV,CpL,ConV,ConL,MolwtV,MolwtL,Sigma,
+              KKV,KKL
c
-----
include 'baffle.i'
c
Local variables:
double precision Aint,Xarea,Vol,Uv,Ul,Rev
c
if (InType .eq. iTbafl) then
  if (iMTC .eq. iMbafl) then
c
    iColD = jColD
    iColH = jColH
c
    Xarea = 3.141d0 * Parms(j,jColD)**2 / 4.0d0
    Vol   = Parms(j,jColH) * Xarea
c
    Uv = Vmf * MolwtV / Xarea / RhoV
    Ul = Lmf * MolwtL / Xarea / RhoL
c
    Rev = 0.0001d0 * Parms(j,jColD) * Parms(j,jArea)
    ReV = ReV * RhoV * Uv / VisV
    ShV = PMTC(j,jCv) * Rev**0.8d0 * ScV**0.33d0
    ShL = PMTC(j,jCl) * ScL**0.5d0
c
    HTU in ft
c
    KKV = ( (CpV/4.1868d3) /

```

```

+          (PMTC(j,jCv) * (7.3734d2*RhoL*U1)**0.44d0 ) ) *
+          ( ScV / (CpV*VisV/ConV) )** (2.d0/3.d0)
c
c      Kv in m/s
c
c      KKV = Uv / (0.3048d0 * KKV * Aint)
c
c      What this is set to doesn't matter as the liquid transfer
c      model should be ignored! So let set it to a LARGE number
c      in case this wasn't done!
c
c      KKL = 1.d0
c
c      if (PMTC(j,jDbg) .gt. 1.d0) then
c          write (io,*) 'MTC Baffle tray ',j
c          write (io,203) KKV,' kv'
c          write (io,203) KKL,' kl'
203      format (1H , 1PE12.5, (A))
c          if (PMTC(j,3) .gt. 2.d0) then
c              write (io,*) 'PMTC ',PMTC(j,1),PMTC(j,2),PMTC(j,3)
c              call UsrDot(io,InType,j,n,m,Parms)
c              if (PMTC(j,3) .gt. 3.d0) then
c                  read(*,*) ii
c              endif
c          endif
c      endif
c
c      endif
c
c      return
c      end

```

The *UsrRep* routine computes the performance of the tray. In this case we set the pressure drop to 1 Pascal for the baffle tray. Of course, this is incorrect and needs to be replaced by actual code computing the pressure drop as a function of the vapor and liquid traffic on the tray.

```

subroutine UsrRep(iOut,iReprt,InType,j,n,m,iColD,iColH,
+          Parms,Pmtc,Ppd,Phold,Pvf,Plf,Pentr,
+          Vmf, Lmf, RhoV, RhoL,
+          VisV, VisL, MolwtV, MolwtL,
+          CpV, CpL, ConV, ConL, Sigma, sf,
+          Pdrop,HoldUp,Phil,FloodF,WeepF,
+          QQv,QQl,TTv,TTi,TTl)

```



```

integer iOut, iReprt, InType, j, n, m, iColD, iColH
double precision Params(n, m), Pmtc(n, m), Ppd(n, m), Phold(n, m),
+          Pvf(n, m), Plf(n, m), Pentr(n, m),
+          Vmf, Lmf, RhoV, RhoL, VisV, VisL,
+          CpV, CpL, ConV, ConL, MolwtV, MolwtL, Sigma,
+          Pdrop, HoldUp, Phil, FloodF, WeepF,
+          QQv, QQl, TTv, TTi, TTL
c -----
c include 'baffle.i'
c
c double precision a
c
c if (InType .eq. iTbaf1) then
c
c   Baffle
c   -----
c
c   iColD = j1ColD
c   iColH = j1ColH
c
c   Pdrop = 1.d0
c
c   if (iReprt .eq. 2) then
c
c     write(iOut, *) 'TestOut ', 0.01d0*DBLE(j)
c
c   endif
c
c endif
c
c return
c end

```

The *UsrHTC*, *UsrFap*, and *UsrFcl* routines we leave unchanged.

Chapter 7

Reactive Distillation

Chapter 8

Extraction

Chapter 9

Three Phase Distillation

Part B

Technical Reference

Chapter 10

Property Models

This chapter contains a catalog of all of the thermodynamic and physical property models available in *ChemSep*. The formulae for most methods are given here but additional reading is available in the following sources:

A: R.C. Reid, J.M. Prausnitz and B.E. Poling, *The Properties of Gases and Liquids*, 4th Ed., McGraw-Hill, New York (1988).

B: S.M. Walas, *Phase Equilibria in Chemical Engineering*, Butterworth Publishers, London (1985).

References to these works are between parentheses, by combining the letter A or B with the page number, for example (A43). Some of the methods described below are referred to by the name given to them in Danner and Daubert (1983).

10.1 Thermodynamic Properties

These properties are required for both the equilibrium and the nonequilibrium models. They determine the equilibrium mole fractions of the phases that are present in the column as well as the terms in the energy balance. Typically, they have a profound influence on the simulation. Selecting the right models will require insight from the user; there are no models that cover all systems and operations.

10.1.1 K-value models

The distribution ratio of component i , the K-value, is defined as the ratio of the mole fraction in the vapor over the mole fraction in the liquid phase:

$$K_i = \frac{y_i}{x_i} \quad (10.1)$$

Of course, for extraction columns, where two liquid phases are present the K-value equals the ratio of the liquid mole fractions in each liquid.

The available K-models for VLE are:

Ideal (A251, B548) K-values for ideal mixtures are given by Raoult's law:

$$K_i = \frac{P_i^*}{P} \quad (10.2)$$

where P_i^* is the vapor pressure of component i and P the system pressure.

EOS (A319, B301) K-values are calculated from the ratio of fugacity coefficients:

$$K_i = \frac{\phi_i^L}{\phi_i^V} \quad (10.3)$$

where the fugacity coefficients (ϕ) are calculated from an equation of state. This model is recommended for separations involving mixtures of hydrocarbons and light gases (hydrogen, carbon dioxide, nitrogen, etc.) at low and high pressures. It is not recommended for nonideal chemical mixtures at low pressures. The EOS must be able to predict vapor as well as liquid fugacity coefficients.

Gamma-Phi (A250, B301) K-Values are calculated using an activity coefficient model for the liquid phase and an equation of state for the vapor:

$$K_i = \frac{\gamma_i \phi_i^* P_i^* P F_i}{\phi_i^V P} \quad (10.4)$$

This option should be used when dealing with nonideal fluid mixtures. It should not be selected for separations at high pressures.

DECHEMA (B301) K-values are calculated from a simplified form of the complete Gamma-Phi model in which the vapor phase fugacity coefficient and Poynting correction factor are assumed equal to unity:

$$K_i = \frac{\gamma_i P_i^*}{P} \quad (10.5)$$

This is the form of the K-value model used in the DECHEMA compilations of equilibrium data (Hence the name given to this menu option). DECHEMA uses the Antoine equation to compute the vapor pressures but *ChemSep* allows you to choose other vapor pressure models if you wish. This option should be used when dealing with non-ideal fluid mixtures. It should not be selected for separations at high pressures.

Chao-Seader (B303) The Chao-Seader method is widely used for mixtures of hydrocarbons and light gases. It is not recommended for nonideal mixtures. The method uses the Regular solution model for the liquid phase and the Redlich Kwong EOS for the vapor phase. An alternative choice would be the Equation of State option.

Polynomial (B11) Calculate K-value as function of the absolute temperature (Kelvin):

$$K_i^{1/m_i} = A_i + B_i T + C_i T^2 + D_i T^3 + E_i T^4 \quad (10.6)$$

You must supply the coefficients *A* through *E* and the exponent *m*.

There is currently only one LLE model available in *ChemSep*. In this model the K-value model is equal to the ratio of the activity coefficients in the two liquid phases:

$$K_i = \frac{x_i^I}{x_i^{II}} = \frac{\gamma_i^{II}}{\gamma_i^I} \quad (10.7)$$

For supercritical extractors an alternative is to use cubic equation of states and define the K-value as the ratio of the fugacity coefficients, just as described above.

10.1.2 Activity coefficient models

Here we discuss the activity coefficient models available in *ChemSep*. For an in depth discussion of these models see the standard references. For the calculation of activity coefficients and their derivatives (for diffusion calculations) see Kooijman and Taylor (1991). Most of these models require interaction parameters that can be loaded from libraries.

Ideal For an ideal system the activity coefficient of all species is unity ($\ln \gamma_i = 0$).

Regular (A284, B217) The regular solution model is due to Scatchard and Hildebrand. It is probably the simplest model of liquid mixtures. The model uses the

Flory-Huggins modification. The activity coefficient is given by:

$$\theta_i = V_i / \sum_k^c x_k V_k \quad (10.8)$$

$$\ln \gamma_i^* = \frac{V_i}{RT} \left(\delta_i - \sum_j^c x_j \delta_j \theta_j \right)^2 \quad (10.9)$$

$$\ln \gamma_i = \ln \gamma_i^* + \ln \theta_i + 1 - \theta_i \quad (10.10)$$

where δ_i is called the solubility parameter and V_i the molal volume of component i (both available in the PCD-file). This model also is incorporated into the Chao-Seader method of estimating K-values.

Margules (A256, B184) The ‘three suffix’ or two parameter form of the Margules equation is implemented in *ChemSep*:

$$\ln \gamma_i = (A_{ij} + 2(A_{ji} - A_{ij})x_i) x_j^2 \quad (10.11)$$

It can only be used for binary mixtures ($i=1, j=2$ and $i=2, j=1$).

Van Laar (A256, B189) The Van Laar equation is

$$\ln \gamma_i = \frac{A_{ij}}{\left(1 + \frac{A_{ij}x_i}{A_{ji}x_j}\right)^2} \quad (10.12)$$

It can only be used for binary mixtures ($i=1, j=2$ and $i=2, j=1$).

Wilson (A274, B192) The Wilson equation was proposed by G.M. Wilson in 1964. It is a ‘two parameter equation’. That means that two interaction parameters per binary pair are needed to estimate the activity coefficients in a multicomponent mixture. For mixtures that do NOT form two liquids, the Wilson equation is, on average, the most accurate of the methods used to predict equilibria in multicomponent mixtures (Reference B). However, for aqueous mixtures the NRTL model usually is superior.

$$\Lambda_{ij} = (V_j/V_i) \exp(-(\mu_{ij} - \mu_{ii})/RT) \quad (10.13)$$

$$S_i = \sum_{j=1}^c x_j \Lambda_{ij} \quad (10.14)$$

$$\ln \gamma_i = 1 - \ln(S_i) - \sum_{k=1}^c x_k \Lambda_{ki}/S_k \quad (10.15)$$

The two interaction parameters per binary pair of components: $(\mu_{ij} - \mu_{ii})$ and $(\mu_{ji} - \mu_{ii})$.

NRTL (A274, B201) The NRTL equation due to Renon and Prausnitz is a three parameter equation. Unlike the original Wilson equation it may also be used for liquid-liquid equilibrium calculations.

$$\tau_{ij} = (g_{ij} - g_{ii})/RT \quad (10.16)$$

$$G_{ij} = \exp(-\alpha_{ij}\tau_{ij}) \quad (10.17)$$

$$S_i = \sum_{j=1}^c x_j G_{ji} \quad (10.18)$$

$$C_i = \sum_{j=1}^c x_j G_{ji} \tau_{ji} \quad (10.19)$$

$$\ln \gamma_i = C_i/S_i + \sum_{k=1}^c x_k G_{ik} (\tau_{ik} - C_k/S_k)/S_k \quad (10.20)$$

The interaction parameters per binary are $a_{ij} = (g_{ij} - g_{ii})$, $a_{ji} = (g_{ji} - g_{ii})$, and α_{ij} (only one α is required as $\alpha_{ij} = \alpha_{ji}$). The temperature dependent version computes the interaction parameters with a reference temperature of $T^o = 25^\circ C = 298K$ and:

$$a_{ij} = a_{ij}^o \left(1 - \frac{T}{T^o}\right) \quad (10.21)$$

UNIQUAC (A274, B205) UNIQUAC stands for Universal Quasi Chemical and is a very widely used model of liquid mixtures that reduces, under certain assumptions, to almost all of the other models mentioned in this section. Like the Wilson equation, it is a two parameter equation but is capable of predicting liquid-liquid equilibria as well as vapor-liquid equilibria. Two versions of the UNIQUAC model are available in *ChemSep*: Original and q-prime. Original is the default option and is to be used if you have obtained interaction parameters from DECHEMA. The q-prime (q') form of UNIQUAC is recommended for alcohol mixtures. An additional pure component parameter, q' , is needed. Both versions are identical if $q' = q$.

$$r = \sum_{i=1}^c x_i r_i \quad (10.22)$$

$$q = \sum_{i=1}^c x_i q_i \quad (10.23)$$

$$\phi_i = x_i r_i / r \quad (10.24)$$

$$\theta_i = x_i q_i / q \quad (10.25)$$

$$\tau_{ji} = \exp(-(\lambda_{ji} - \lambda_{ii})/RT) \quad (10.26)$$

$$S_i = \sum_{j=1}^c \theta_j \tau_{ji} \quad (10.27)$$

$$\ln \gamma_i^C = \left(1 - \frac{z}{2} q_i\right) \ln \left(\frac{\phi_i}{x_i}\right) + \frac{z}{2} q_i \ln \left(\frac{\theta_i}{x_i}\right) - \frac{r_i}{r} + \frac{z}{2} q \left(\frac{r_i}{r} - \frac{q_i}{q}\right) \quad (10.28)$$

$$\ln \gamma_i^R = q_i \left(1 - \ln(S_i) - \sum_{k=1}^c \frac{\theta_k \tau_{ik}}{S_k}\right) \quad (10.29)$$

$$\ln \gamma_i = \ln \gamma_i^C + \ln \gamma_i^R \quad (10.30)$$

Here, ϕ_i is the segment fraction (similar to the volume fraction) and θ_i is the area fraction of component i . The interaction parameters per binary are $(\lambda_{ij} - \lambda_{ii})$ and $(\lambda_{ji} - \lambda_{ii})$. The parameters r_i and q_i are read from the component database (PCD file).

UNIFAC (A314, B219) UNIFAC is a group contribution method that is used to predict equilibria in systems for which NO experimental equilibrium data exist. The method is based on the UNQUAC equation, but is completely predictive in the sense that it does not require interaction parameters. Instead, these parameters are estimated from group contributions of all the molecules in the mixture. UNIFAC uses a set of main groups, each of which have their own interaction parameters a_{ij} (and a_{ji}) with which the τ_{ij} are computed:

$$\tau_{ij} = \exp(-a_{ij}/T) \quad (10.31)$$

These main groups are further divided into subgroups (k) that have their own R_k and Q_k values (these parameters are obtained from the van der Waals group volume and surface areas). The r_i and q_i for component i are computed by summing the product of the number of group k in i with R_k , respectively Q_k :

$$r_i = \sum_k \nu_k^i R_k \quad (10.32)$$

$$q_i = \sum_k \nu_k^i Q_k \quad (10.33)$$

The combinatorial part can then be calculated using the UNQUAC equations. In the residual part of the activity coefficient the segment fraction (θ) is replaced by:

$$Q = \sum_{i=1}^c X_i Q_i \quad (10.34)$$

$$\theta_i = X_i Q_i / Q \quad (10.35)$$

where X_i is the mole fraction of group i in the *mixture*. Summations and residual parts are then computed for group k instead of component i (using Eqs. 10.27, 10.29). The residual part of the activity coefficient for component i is then computed with:

$$\ln \gamma_i^R = \sum_k \nu_k^i \left(\ln \gamma_k^R - \ln \gamma_k^{R,i} \right) \quad (10.36)$$

where the last term, $\ln \gamma_k^{R,i}$, represents the residual part in a reference solution containing only molecules of type i . This normalization makes $\ln \gamma_i^R$ become zero when the mixture contains only molecules i .

If you select one of the other models but fail to specify a complete set of the interaction parameters, then UNIFAC is used to compute any unspecified parameters.

ASOG (A313, B219) ASOG is a group contribution method similar to UNIFAC but based on the Wilson equation. It was developed before UNIFAC but is less widely used because of the comparative lack of fitted group interaction parameters.

10.1.3 Vapor pressure models

Antoine (A208, B11) The Antoine Equation is:

$$\ln P_i^* = A_i - \frac{B_i}{T + C_i} \quad (10.37)$$

Note the natural logarithm. This option should be selected if you are using activity coefficient models with parameters from the DECHEMA series. Antoine parameters are available in the *ChemSep* data files and need not be loaded. If you wish to use different parameters then select the *Extended Antoine* model and enter the parameters yourself.

Extended Antoine (B11) An equation with more parameters that is sometimes known as the extended Antoine equation is incorporated in *ChemSep*

$$\ln P_i^* = A_i + \frac{B_i}{C_i + T} + D_i T + E_i \ln T + F_i T^G \quad (10.38)$$

The parameters A through G must be supplied by the user. A library of parameters for some common chemicals is provided with *ChemSep* in the file EANTOINE.LIB.

DIPPR (B11) The Design Institute for Physical Property Research (DIPPR) has published a correlation for the vapor pressure that takes the form:

$$\ln P_i^* = A_i + \frac{B_i}{T} + C_i \ln T + D_i T_i^E \quad (10.39)$$

Riedel (B523) The Riedel equation is best suited to nonpolar mixtures:

$$\zeta_T = 36/T_r + 96.7 \log T_r - 35 - T_r^6 \quad (10.40)$$

$$\zeta_{Tb} = 36/T_{rb} + 96.7 \log T_{rb} - 35 - T_{rb}^6 \quad (10.41)$$

$$\phi = 0.118\zeta_T - 7 \log T_r \quad (10.42)$$

$$\psi = 0.0364\zeta_T - \log T_r \quad (10.43)$$

$$\alpha = \frac{0.136\zeta_{Tb} + \log P_c - 5.01}{0.0364\zeta_{Tb} - \log T_{rb}} \quad (10.44)$$

$$\log P_r^* = -\phi - (\alpha - 7)\psi \quad (10.45)$$

Lee-Kesler (A207, B69) Lee and Kesler used a Pitzer expansion to obtain:

$$\ln P_i^* = f^{(0)} + \omega_i f^{(1)} \quad (10.46)$$

$$f^{(0)} = 5.92714 - \frac{6.09648}{T_r} - 1.28862 \ln T_r + 0.169347T_r^6 \quad (10.47)$$

$$f^{(1)} = 15.2518 - \frac{15.6875}{T_r} - 13.4721 \ln T_r + 0.43577T_r^6 \quad (10.48)$$

where $T_r = T/T_{Ci}$. Both the Riedel and Lee-Kesler models are recommended for hydrocarbon mixtures in particular.

10.1.4 Equations of State (EOS)

Three types of equations of state may be selected in *ChemSep*; *Ideal Gas*, *Virial*, and *Cubic EOS*. The ideal gas EOS is, of course:

$$P = \frac{RT}{V} \quad (10.49)$$

The fugacity coefficient of an ideal gas mixture (B3) is unity.

The Virial and cubic EOS are discussed in the sections below.

10.1.5 Virial EOS

The two-term virial equation:

$$P = \frac{RT}{V} + \frac{BRT}{V} \quad (10.50)$$

where B is the second virial coefficient. Different methods are available for estimating this coefficient.

Hayden-O'Connell (B39) Hayden and O'Connell have provided a method of predicting the second virial coefficient for multicomponent vapor mixtures. The method is quite complicated (see Prausnitz *et al.*, 1980) but is well suited to ideal and nonideal systems at low pressures. You must input the association parameters. A library of association parameters is provided with *ChemSep* in the file HAYDENO.IPD.

Tsonopoulous (B45) The method of Tsonopoulous for estimating virial coefficients is recommended for hydrocarbon mixtures at low pressures. It is based on a correlation due to Pitzer.

$$B = \sum_{i=1}^c \sum_{j=1}^c y_i y_j B_{ij} \quad (10.51)$$

$$B_{ij} = RT_{c,ij} P_{c,ij} \left(B_{ij}^{(0)} + \omega_{ij} B_{ij}^{(1)} \right) \quad (10.52)$$

$$B_{ij}^{(0)} = 0.1445 - \frac{0.33}{T_r} - \frac{0.1385}{T_r^2} - \frac{0.0121}{T_r^3} - \frac{0.000607}{T_r^8} \quad (10.53)$$

$$B_{ij}^{(1)} = 0.0637 + \frac{0.331}{T_r^2} - \frac{0.423}{T_r^3} - \frac{0.0008}{T_r^8} \quad (10.54)$$

$$\omega_{ij} = \frac{\omega_i + \omega_j}{2} \quad (10.55)$$

$$Z_{c,ij} = \frac{Z_{ci} + Z_{cj}}{2} \quad (10.56)$$

$$V_{c,ij}^{1/3} = \frac{V_{ci}^{1/3} V_{cj}^{1/3}}{2} \quad (10.57)$$

$$T_{c,ij} = (1 - k_{ij}) \sqrt{T_{ci} T_{cj}} \quad (10.58)$$

$$P_{c,ij} = \frac{Z_{c,ij} R T_{c,ij}}{V_{c,ij}} \quad (10.59)$$

Binary interaction parameters k_{ij} must be supplied by the user. For paraffins k_{ij} can be estimated from:

$$k_{ij} = 1 - \frac{8\sqrt{V_{ci} V_{cj}}}{(V_{ci}^{1/3} + V_{cj}^{1/3})^3} \quad (10.60)$$

DIPPR The Design Institute for Physical Property Research (DIPPR) has published a correlation for the second virial coefficient.

Chemical theory This is an extension of the Hayden O'Connell virial model that takes into account the association between molecules (see Prausnitz *et al.*, 1980).

Since the mole fractions are a function of the association, an iterative method (here Newton's method) must be used to obtain them in order to compute the virial coefficients.

10.1.6 Cubic EOS

Van der Waals (A43, B15) The Van der Waals (VdW) Equation was the first cubic equation of state.

$$P = \frac{RT}{V - b} - \frac{a}{V^2} \quad (10.61)$$

with

$$a_i = \frac{27R^2T_{ci}^2}{64P_{ci}} \quad (10.62)$$

$$b_i = \frac{RT_{ci}}{8P_{ci}} \quad (10.63)$$

and the mixing rules:

$$a = \sum_{i=1}^c \sum_{j=1}^c y_i y_j a_{ij} \quad (10.64)$$

$$a_{ij} = \sqrt{a_i a_j} \quad (10.65)$$

$$b = \sum_{i=1}^c y_i b_i \quad (10.66)$$

The VdW equation should not be used to determine properties of liquid phases, thus it may not be selected for the EOS K-value model. The basic equation has served as a starting point for many other EOS.

Redlich Kwong (A43, B43) The Redlich Kwong (RK) equation is used in the Chao-Seader method of computing thermodynamic properties. The RK equation should not be used to determine properties of liquid phases (*ChemSep* does not allow its selection as the EOS K-value model).

$$P = \frac{RT}{V - b} - \frac{a}{\sqrt{T}V(V + b)} \quad (10.67)$$

with

$$a_i = \frac{\Omega_a R^2 T_{ci}^{2.5}}{P_{ci}} \quad (10.68)$$

$$\Omega_a = 0.42748 \quad (10.69)$$

$$b_i = \frac{\Omega_b RT_{ci}}{P_{ci}} \quad (10.70)$$

$$\Omega_b = 0.08664 \quad (10.71)$$

and the mixing rules:

$$a = \sum_{i=1}^c \sum_{j=1}^c y_i y_j a_{ij} \quad (10.72)$$

$$a_{ij} = (1 - k_{ij}) \sqrt{a_i a_j} \quad (10.73)$$

$$b = \sum_{i=1}^c y_i b_i \quad (10.74)$$

where k_{ij} is a binary interaction parameter (original RK: $k_{ij} = 0$).

Soave Redlich Kwong (A43, B52) Soave's modification of the Redlich Kwong (SRK) EOS is a very widely used method of computing thermodynamic properties. The SRK EOS is most suitable for computing properties of hydrocarbon mixtures.

$$P = \frac{RT}{V - b} - \frac{a}{V(V + b)} \quad (10.75)$$

with

$$a_i = a_i(T_{ci}) \alpha_i \quad (10.76)$$

$$a_i(T_{ci}) = \frac{\Omega_a R^2 T_{ci}^2}{P_{ci}} \quad (10.77)$$

$$\Omega_a = 0.42747 \quad (10.78)$$

$$\alpha_i = \left(1 + (0.480 + 1.574\omega_i - 0.176\omega_i^2)(1 - \sqrt{T_{ri}}) \right)^2 \quad (10.79)$$

$$b_i = \frac{\Omega_b R T_{ci}}{P_{ci}} \quad (10.80)$$

$$\Omega_b = 0.08664 \quad (10.81)$$

and the mixing rules:

$$a = \sum_{i=1}^c \sum_{j=1}^c y_i y_j a_{ij} \quad (10.82)$$

$$a_{ij} = (1 - k_{ij}) \sqrt{a_i a_j} \quad (10.83)$$

$$b = \sum_{i=1}^c y_i b_i \quad (10.84)$$

API SRK EOS (B53) Graboski and Daubert modified the coefficients in the SRK EOS and provided a special relation for hydrogen. This modification of the SRK EOS has been recommended by the American Petroleum Institute (API), hence the name of this option. It uses the same equations as the SRK except that:

$$\alpha_i = \left(1 + (0.48508 + 1.55171\omega_i - 0.15613\omega_i^2)(1 - \sqrt{T_{ri}}) \right)^2 \quad (10.85)$$

with a special relation for hydrogen:

$$\alpha_i = 1.202e^{-0.30288T_{r,i}} \quad (10.86)$$

Peng Robinson EOS (A43, B54) The Peng-Robinson equation is another cubic EOS that owes its origins to the RK and SRK EOS. The PR EOS, however, gives improved predictions of liquid phase densities.

$$P = \frac{RT}{V-b} - \frac{a}{V(V+b) + b(V-b)} \quad (10.87)$$

The parameters are given by the same equations as for the SRK EOS with the following differences.

$$\Omega_a = 0.45724 \quad (10.88)$$

$$\alpha_i = \left(1 + (0.37464 + 1.5422\omega_i - 0.26992\omega_i^2)(1 - \sqrt{T_r})\right)^2 \quad (10.89)$$

$$\Omega_b = 0.07880 \quad (10.90)$$

10.1.7 Enthalpy

The mixture enthalpy can be calculated from the component enthalpies with:

$$H = \sum_{i=1}^c z_i H_i \quad (10.91)$$

where z_i is the mole fraction of species i in the phase in question. The component enthalpy may be expressed as the sum of the ideal contribution and an excess enthalpy:

$$H_i = H_i^{id} + H_i^{ex} \quad (10.92)$$

where H_i^{ex} is the excess enthalpy and H_i^{id} is the ideal contribution. The ideal gas enthalpy of component i is given by:

$$H_i^{id} = \int_{T_0}^T C_{p,i}^{g,id} dT + H_{form,i}^{T_0} + H_{ref,i} \quad (10.93)$$

where $C_{p,i}^{id}$ is the ideal gas heat capacity of component i , T_0 is the enthalpy reference temperature, $H_{form,i}^{T_0}$ the enthalpy of formation at T_0 , and T is the actual temperature at which the enthalpy is to be calculated. Note that there is no composition dependence in the ideal gas enthalpy! The normal reference temperature (T_0) in /cs/ is set to 298 K (see the "References" menu under the properties selection).

The reference enthalpy H_{ref} depends on the model selected. If an no EOS model is used then we need to correct for the state. If we select the vapor as reference state (the default), the reference enthalpies are:

$$H_{ref,i}^v = 0 \quad (10.94)$$

$$H_{ref,i}^l = -\Delta H_i^{vap}(T) \quad (10.95)$$

or, when we choose the liquid as reference state:

$$H_{ref,i}^v = \Delta H_i^{vap}(T_0) \quad (10.96)$$

$$H_{ref,i}^l = \Delta H_i^{vap}(T_0) - \Delta H_i^{vap}(T) \quad (10.97)$$

where the latent heat of vaporization (ΔH_i^{vap}) is computed with a temperature correlation (see the section on physical properties below) or, if no correlation parameters are known, using the method of Pitzer *et al.*:

$$\Delta H_i^{vap}(T) = RT_{c,i}(7.08(1 - T_{r,i})^{0.354} + 10.95(1 - T_{r,i})^{0.456}\omega_i) \quad (10.98)$$

If an EOS model is used for computing the enthalpies, we do not need to correct for the state and consequently $H_{ref} = 0$.

The enthalpies of formation are normally not needed, as they normally cancel out. These terms are large compared to the other contributions and make the enthalpy balance less accurate, so they can be ignored. Of course, when reactions are present they need to be included:

$$H_{form,i}^{T_0} = H_{form,i}^{298K} + \int_{298K}^{T_0} C_{p,i}^{g,id} dT \quad (10.99)$$

If an equation of state model is used for the K-values in *ChemSep*, the excess enthalpy of both phases is calculated from an equation of state as follows:

$$H_i^{ex} = -RT^2 \left(\frac{\partial \ln \phi_i}{\partial T} \right) \quad (10.100)$$

If an activity coefficient model is used for the liquid phase, then the component excess enthalpy is computed from:

$$H_i^{ex} = -RT^2 \left(\frac{\partial \ln \gamma_i}{\partial T} \right) \quad (10.101)$$

ChemSep incorporates the following methods for estimating the enthalpy:

None No enthalpy balance is used in the calculations. **WARNING:** the use of this model with subcooled and superheated feeds or for columns with heat addition or removal on some of the stages will give incorrect results. The heat duties of the condenser and reboiler will be reported as zero since there is no basis for calculating them.

Ideal (B152) In this model the enthalpy is computed from the ideal gas contribution and the excess enthalpy is assumed to be zero.

Excess (B518) This model includes the ideal enthalpy as above. The excess enthalpy is calculated from the activity coefficient model or the temperature derivative of the fugacity coefficients dependent on the choice of the model for the K-values, and is added to the ideal part.

Polynomial vapor as well as liquid enthalpy are calculated as functions of the absolute temperature (K). Both the enthalpies use the following function:

$$H_i = A_i + B_i T + C_i T^2 + D_i T^3 \quad (10.102)$$

You must enter the coefficients A through D in the ‘Load Data’ option of the Properties menu for vapor and liquid enthalpy for each component.

EOS the enthalpies are always computed using a EOS model, even when the K-values are computed with another model.

10.1.8 Entropy

The mixture entropy can be calculated from the component entropies in a similar manner as the enthalpy:

$$S = \sum_{i=1}^c z_i S_i \quad (10.103)$$

where z_i is the mole fraction of species i in the phase in question. The component entropy may be expressed as the sum of the component entropy plus the contributions of ideal and excess mixing entropy:

$$S_i = S_i^{id} + S_i^{ex} \quad (10.104)$$

where S_i^{ex} is the excess entropy and S_i^{id} is the ideal contribution for component i . The ideal (gas) entropy of component i is given by:

$$S_i^{id} = S_{0,i} + \int_{T_0}^T \frac{C_{p,i}^{g,id}}{T} dT - R \ln \left(\frac{p}{p_0} \right) - R \ln z_i + S_{ref,i} \quad (10.105)$$

where $C_{p,i}^{id}$ is the ideal gas heat capacity of component i , T_0 is the reference temperature, p_0 the reference pressure, $S_{0,i}$ the absolute entropy of component i at T_0 and p_0 , z_i the mole fraction of i , and T and p are the actual temperature and pressure at which the entropy is to be calculated. The entropy reference point is fixed as vapor at 298 K and 1 atm because the absolute component entropies are defined at this temperature

and pressure. These component absolute entropies are stored in the pure component database. The reference entropy is only added for liquids when we do not calculate the entropy using an EOS; it corrects for the entropy of vaporization:

$$S_{ref,i}^l = -\frac{\Delta H_{vap,i}(T_{boil})}{T_{boil}} \quad (10.106)$$

where $\Delta H_{vap,i}(T_{boil})$ is the vaporizations enthalpy of component i at T_{boil} , the component normal boiling temperature. Excess mixture entropies are computed with

$$S_i^{ex} = -R \left(T \frac{\partial \ln \gamma_i}{\partial T} + \ln \gamma_i \right) \quad (10.107)$$

when an activity coefficient model is used. If an equation of state is used γ_i is replaced by ϕ_i , the fugacity coefficient of i .

10.1.9 Exergy

The exergy (Ex) of a mixture or stream can be calculated from the enthalpy and entropy with

$$Ex = H - T_{surr} S \quad (10.108)$$

where T_{surr} is the surroundings temperature of the process (this temperature is normally set to 298 K but can be changed under the "References" menu in the properties selection). Note that with the inclusion of the enthalpy, the exergy in *ChemSep* is also a relative quantity with respect to the enthalpy reference point.

Exergy is sometimes also referred to as "availability", it indicates the work that can be extracted from a stream that is brought to equilibrium with its surrounding state. Therefore, exergies provide useful insights in the thermodynamic efficiency of processes. The second law of thermodynamics states that for any real, irreversal process there is a nett entropy production (ΔS_{irr}), for which work (either direct or indirect in the form of exergy) is lost. The lost work equals $T_o \Delta S_{irr}$ and the higher this amount relative to the work/exergy input, the lower is the thermodynamic efficiency.

10.2 Physical Properties

These properties are not needed for equilibrium stage calculations but are required for apparatus design and nonequilibrium simulations. They include: density, viscosity, thermal conductivity, diffusivity, and surface tension.

Table 10.1: Temperature correlations

Equation number	Parameter(s)	Formula
2	A,B	$A + BT$
3	A-C	$A + BT + CT^2$
4	A-D	$A + BT + CT^2 + DT^3$
5	A-E	see 100
6	A-E	$A + BT + CT^2 + DT^3 + E/T^2$
10	A-C	$\exp\left(A - \frac{B}{C+T}\right)$
100	A-E	$A + BT + CT^2 + DT^3 + ET^4$
101	A-E	$\exp\left(A + \frac{B}{T} + C \ln T + DT^E\right)$
102	A-D	$AT^B/(1 + C/T + D/T^2)$
103	A-D	$A + B \exp\left(-\frac{C}{TB}\right)$
104	A-E	$A + \frac{B}{T} + \frac{C10^6}{T^3} + \frac{D10^{16}}{T^8} + \frac{E10^{18}}{T^9}$
105	A-D	$A/B^{(1+(1-T/C)^D)}$
106	A-E	$A(1 - T_r)^{(B+CT_r+DT_r^2+ET_r^3)}$
107	A-E	$A + B\left(\frac{C}{T} / \sinh\left(\frac{C}{T}\right)\right)^2 + E\left(\frac{D}{T} / \cosh\left(\frac{D}{T}\right)\right)^2$
114	A-D	$AT + BT^2/2 + CT^3/3 + DT^4/4$
117	A-E	$AT + B\left(\frac{C}{T}\right) / \tanh\left(\frac{C}{T}\right) - D\left(\frac{E}{T}\right) / \tanh\left(\frac{E}{T}\right)$

Many different equations are available in *ChemSep* to evaluate physical properties over a certain temperature range. These temperature correlations are assigned a unique number in the range of 0-255 (see Table 10.1 and Table 10.7). For each up to 5 parameters (A-E) are available. Table 10.2 shows which pure component properties can be modeled with temperature correlations and their typical correlation number.

Any of the equations may be used for any of the physical properties but, of course, some formulas were specifically developed for prediction of particular properties. Besides the parameters A-E the temperature limits of the correlation must also be present. If the temperature specified falls out of the temperature range of a correlation (or the temperature limits are missing/incomplete) normally an alternative (default) method will be used automatically.

Physical properties models can be selected manually or the automatic selection can be used (this is the default). Below we discuss the models for calculating physical properties for pure components and mixtures, for vapor and liquid phases. *ChemSep* will make it's own choices of model when none has been selected by the user (by leaving the selection as a *). Depending on the range, phase, conditions, data avail-

Table 10.2: Component properties with the typical correlation number

Liquid density	105
vapor pressure	10, 101
Heat of vaporisation	106
Liquid heat capacity	100
Ideal gas heat capacity	4, 107
Second virial coefficient	104
Liquid viscosity	2, 101
vapor viscosity	102
Liquid thermal conductivity	100
vapor thermal conductivity	102
Surface tension	106

Table 10.3: Default physical property correlations

Mixture liquid density	Rackett
Component liquid density	Polynomial
vapor density	Cubic EOS
Mixture liquid viscosity	Molar averaging
Component vapor viscosity	Polynomial/Letsou-Stiel
Mixture vapor viscosity	Brokaw
Component vapor viscosity	Polynomial
Mixture liquid thermal conductivity	Molar average
Component liquid thermal conductivity	Polynomial
Mixture vapor thermal conductivity	Molar average
Component vapor thermal conductivity	Polynomial/9B-3
Liquid diffusivity	Kooijman-Taylor/Wilke-Chang
vapor diffusivity	Fuller <i>et al.</i>
Mixture surface tension	Molar average
Component surface tension	Polynomial
Liquid-liquid interfacial tension	Jufu <i>et al.</i>

ability, and required property *ChemSep* will guess the best model to use. *ChemSep* does allow you to pick default models, and will use them if the model's range is valid. In case a property cannot be computed with a specific model it will use an estimation method or a fixed estimate (it is a good habit to check predicted physical properties when possible).

Certain methods require mixture (critical) properties, commonly used mixing rules are:

$$T_{c,m} = \sum_{i=1}^c x_i T_{c,i} \quad (10.109)$$

$$V_{c,m} = \sum_{i=1}^c x_i V_{c,i} \quad (10.110)$$

$$Z_{c,m} = \sum_{i=1}^c x_i Z_{c,i} \quad (10.111)$$

$$P_{c,m} = Z_{c,m} R T_{c,m} / V_{c,m} \quad (10.112)$$

$$M_m = \sum_{i=1}^c x_i M_i \quad (10.113)$$

These relations will be referred to as the ‘normal’ or ‘conventional’ mixing rules. Reduced properties may be calculated from:

$$T_r = T / T_c \quad (10.114)$$

$$P_r = P / P_c \quad (10.115)$$

$$V_r = V / V_c \quad (10.116)$$

unless specified otherwise.

10.2.1 Liquid density

Mixture liquid densities (in $kmol/m^3$) may be estimated from:

Equation of State The Peng-Robinson equation of state can be used to calculate the mixture compressibility directly from pure component critical properties and mixture parameters, from which the density can be calculated easily. Use this method if some components in the mixture are supercritical.

Amagat’s law

$$\frac{1}{\rho_m^L} = \sum_{i=1}^c \frac{x_i}{\rho_i^L} \quad (10.117)$$

where the pure component liquid densities, ρ_i^L , are computed as discussed below.

Rackett (A67,89) This method requires pure component critical temperatures, pressures, molecular weights and Rackett parameters (the critical compressibilities are used if $Z_{R,i}$ is not known):

$$T_{c,m} = \sum_{i=1}^c x_i T_{c,i} \quad (10.118)$$

$$Z_{R,m} = \sum_{i=1}^c x_i Z_{R,i} \quad (10.119)$$

$$T_r = \frac{T}{T_{c,m}} \quad (10.120)$$

$$F_z = Z_{R,m}^{(1+(1-T_r)^{2/7})} \quad (10.121)$$

$$A = \sum_{i=1}^c \frac{x_i T_{c,i}}{M_i P_{c,i}} \quad (10.122)$$

$$\rho_m^L = 1/ARF_z \sum_{i=1}^c x_i M_i \quad (10.123)$$

If the reduced temperature, T_r , is greater than unity a default value of 50 kmol/m^3 is used.

Yen-Woods Mixture critical temperature, volume, and compressibility are calculated with the ‘normal’ mixing rules. If the reduced temperature, $T_r = T/T_{c,m}$, is greater than unity a default value of 50 kmol/m^3 is used, otherwise the density is calculated from:

$$T_* = (1 - T_r)^{1/3} \quad (10.124)$$

$$A = 17.4425 - 214.578Z_c + 989.625 * Z_c^2 - 1522.06Z_c^3 \quad (10.125)$$

$$Z_c \leq 0.26 : B = -3.28257 + 13.6377Z_c + 107.4844Z_c^2 - 384.211Z_c^3 \quad (10.126)$$

$$Z_c > 0.26 : B = 60.20901 - 402.063Z_c + 501Z_c^2 + 641Z_c^3 \quad (10.127)$$

$$\rho_m^L = \frac{1 + AT_* + BT_*^2 + (0.93 - B)T_*^4}{V_c} \quad (10.128)$$

Hankinson-Thompson (A55-66,89,90) The methods of Hankinson and Thomson (AIChE J, 25, 653, 1979) and Thomson *et al.* (AIChE J, 28, 671, 1982) are summarised below:

$$V_m^* = \frac{1}{4} \left(\sum_{i=1}^c x_i V_i^* + 3 \left(\sum_{i=1}^c x_i V_i^{*2/3} \right) \left(\sum_{i=1}^c x_i V_i^{*1/3} \right) \right) \quad (10.129)$$

$$T_{c,m} = \sum_{i=1}^c \sum_{j=i}^c x_i x_j V_{ij}^* T_{c,ij} / V_m^* \quad (10.130)$$

$$\omega_{SRK,m} = \sum_{i=1}^c x_i \omega_{SRK,i} \quad (10.131)$$

$$Z_{c,m} = 0.291 - 0.08 \omega_{SRK,m} \quad (10.132)$$

$$P_{c,m} = Z_{c,m} R T_{c,m} / V_m^* \quad (10.133)$$

If the reduced temperature is larger than unity a default value of 50 $kmol/m^3$ is used, otherwise the saturated liquid volume (V_s) is calculated from:

$$\frac{V_s}{V_m^*} = V_R^{(0)} (1 - \omega_{SRK,m} V_R^{(\delta)}) \quad (10.134)$$

$$V_R^{(0)} = 1 + a(1 - T_r)^{1/3} + b(1 - T_r)^{2/3} + c(1 - T_r) + d(1 - T_r)^{4/3} \quad (10.135)$$

$$V_R^{(\delta)} = \frac{e + fT_r + gT_r^2 + hT_r^3}{(T_r - 1.00001)} \quad (10.136)$$

where

a=-1.52816	e=-0.296123
b= 1.43907	f= 0.386914
c=-0.81446	g=-0.0427258
d= 0.190454	h=-0.0480645

The molar density is the inverse of the liquid molar volume.

For the density of compressed liquids the saturated liquid volume is corrected as follows (Thomson *et al.*, AIChE J, **28**, 671, 1982):

$$V = V_s \left(1 - c \ln \frac{\beta + P}{\beta + P_{vpm}} \right) \quad (10.137)$$

$$\beta/P_c = -1 + a(1 - T_r)^{1/3} + b(1 - T_r)^{2/3} + d(1 - T_r) + e(1 - T_r)^{4/3} \quad (10.138)$$

$$e = \exp(f + g\omega_{SRK,m} + h\omega_{SRK,m}^2) \quad (10.139)$$

$$c = j + k\omega_{SRK} \quad (10.140)$$

where

a=-9.070217	g= 0.250047
b= 62.45326	h= 1.14188
d=-135.1102	j= 0.0861488
f= 4.79594	k= 0.0344483

with the vapor pressure from the generalized Riedel equations:

$$P_{vpm} = P_{c,m} P_{rm} \quad (10.141)$$

$$\log P_{rm} = P_{rm}^{(0)} + \omega_{SRK,m} P_{rm}^{(1)} \quad (10.142)$$

$$P_{rm}^{(0)} = 5.8031817 \log T_{rm} + 0.07608141\alpha \quad (10.143)$$

$$P_{rm}^{(1)} = 4.86601 \log T_{rm} + 0.03721754\alpha \quad (10.144)$$

$$\alpha = 35 - 36/T_{rm} - 96.736 \log T_{rm} + T_{rm}^6 \quad (10.145)$$

$$T_{rm} = T/T_{c,m} \quad (10.146)$$

This method should be used for reduced temperatures from 0.25 up to the critical point.

Pure component liquid densities are computed from the Peng-Robinson EOS for temperatures above a components critical temperature, otherwise by one of the following methods:

Polynomial When within the temperature range, a polynomial is the default way for calculating component liquid densities.

Rackett

$$F_z = Z_R^{(1+(1-T_r)^{2/7})} \quad (10.147)$$

$$\rho_m^L = P_c/RT_c F_z \quad (10.148)$$

COSTALD The Hankinson and Thompson method described as above but with pure component parameters.

The pure component liquid densities are corrected for pressure effects with the correction of Thomson *et al.* (1982) as described for the Hankinson and Thompson method for mixtures.

10.2.2 Vapor density

Vapor densities are computed with the equation of state selected for the thermodynamic properties (possible selections are *Ideal gas EOS*, *Virial EOS*, and *Cubic EOS*).

10.2.3 Liquid Heat Capacity

The mixture liquid heat capacity is the molar average of the component liquid heat capacities, that often are computed from a temperature correlation. Alternatively the liquid heat capacity could be computed from a corresponding states method and the

ideal gas capacity. *ChemSep* uses a temperature correlation for all temperatures to prevent problems arising from using different liquid heat capacity methods in the same column (that especially trouble nonequilibrium models). Liquid heat capacities could also be computed from selected thermodynamic models to circumvent this problem.

10.2.4 Vapor Heat Capacity

The mixture vapor heat capacity is the molar average of the component vapor heat capacities, that are computed from the ideal gas heat capacity 4 parameter polynomial (A657). If no parameters for this correlation are present, the vapor heat capacity temperature correlation is used (if within the temperature range).

10.2.5 Liquid Viscosity

Mixture liquid viscosity are computed from the pure component liquid viscosities:

$$\ln \eta_m^L = \sum_{i=1}^c z_i \ln \eta_i^L \quad (10.149)$$

where z_i are either the mole fractions (for molar averaging, the default) or, alternatively, the weight fractions for mass averaging. A better method is due to Teja and Rice (1981, A479). However, this method requires interaction parameters. Here, a different mixing rule (for $T_{cij}V_{cij}$) is used which improves the model predictions with unity interaction coefficients:

$$\omega_m = \sum_{i=1}^c x_i \omega_i \quad (10.150)$$

$$M_m = \sum_{i=1}^c x_i M_i \quad (10.151)$$

$$V_{cm} = \sum_{i=1}^c \sum_{j=1}^c x_i x_j V_{cij} \quad (10.152)$$

$$V_{cij} = \frac{(V_{ci}^{1/3} + V_{cj}^{1/3})^3}{8} \quad (10.153)$$

$$T_{cm} = \frac{\sum_{i=1}^c \sum_{j=1}^c x_i x_j T_{cij} V_{cij}}{V_{cm}} \quad (10.154)$$

$$T_{cij} V_{cij} = \psi_{ij} \frac{T_{ci} V_{ci} + T_{cj} V_{cj}}{2} \quad (10.155)$$

where ψ_{ij} is set to unity for all components. The liquid viscosity of the mixture is computed from two reference components

$$\ln(\epsilon_m \eta_m) = \ln(\epsilon_1 \eta_1) + (\ln(\epsilon_2 \eta_2) - \ln(\epsilon_1 \eta_1)) \left(\frac{\omega_m - \omega_1}{\omega_2 - \omega_1} \right) \quad (10.156)$$

with ϵ defined as

$$\epsilon_i = \frac{V_{ci}^{2/3}}{\sqrt{T_{ci} M_i}} \quad (10.157)$$

and the reference component viscosities are evaluated at $T.T_{ci}/T_{cm}$. Component liquid viscosities are calculated from the liquid viscosity temperature correlation if the temperature is within the valid range. Otherwise the component viscosity is computed from the Letsou-Stiel method (1973, see A471):

$$\xi = \frac{2173.424 T_{c,i}^{1/6}}{\sqrt{M_i} P_{c,i}^{2/3}} \quad (10.158)$$

$$\xi^{(0)} = (1.5174 - 2.135 T_r + 0.75 T_r^2) 10^{-5} \quad (10.159)$$

$$\xi^{(1)} = (4.2552 - 7.674 T_r + 3.4 T_r^2) 10^{-5} \quad (10.160)$$

$$\eta_i^L = (\xi^{(0)} + \omega \xi^{(1)}) / \xi \quad (10.161)$$

Alternatively, the simple temperature correlation given in Reid *et al.* (A439) can be used:

$$\log \eta = A + B/T \quad (10.162)$$

A high pressure correction by Lucas (A436) is used to correct for the influence of the pressure on the liquid viscosity:

$$\eta = \frac{1 + D(\Delta P_r / 2.118)^A}{1 + C \omega_i \Delta P_r} \eta_{SL} \quad (10.163)$$

where η_{SL} is the viscosity of the saturated liquid at P_{vp} , and

$$\Delta P_r = (P - P_{vp}) / P_{ci} \quad (10.164)$$

$$A = 0.9991 - (4.674 \times 10^{-4} / (1.0523 T_r^{-0.03877} - 1.0513)) \quad (10.165)$$

$$C = -0.07921 + 2.1616 T_r - 13.4040 T_r^2 + 44.1706 T_r^3 - 84.8291 T_r^4 + 96.1209 T_r^5 - 59.8127 T_r^6 + 15.6719 T_r^7 \quad (10.166)$$

$$D = (0.3257 / (1.0039 - T_r^{2.573})^{0.2906}) - 0.2086 \quad (10.167)$$

10.2.6 Vapor Viscosity

Mixture vapor viscosities are computed from component viscosities as follows:

$$\eta_m^L = \sum_{i=1}^c \frac{x_i \eta_i^L}{\sum x_i \phi_{ij}} \quad (10.168)$$

where the interaction parameters ϕ_{ij} can be calculated by Wilke's method (1950):

$$\phi_{ij} = \left(\frac{1 + \sqrt{\eta_i/\eta_j} (M_i/M_j)^{1/4}}{\sqrt{8(1 + M_i/M_j)}} \right)^2 \quad (10.169)$$

or by Brokaw's method:

$$\phi_{ij} = SA \sqrt{\eta_i/\eta_j} \quad (10.170)$$

$$sm = \left(\frac{4}{(1 + M_j/M_i)(1 + M_i/M_j)} \right)^{1/4} \quad (10.171)$$

$$A = \frac{sm}{\sqrt{M_i/M_j}} \left(1 + \frac{(M_i/M_j - (M_i/M_j)^{0.45})}{2(1 + M_i/M_j)} + \frac{(1 + (M_i/M_j)^{0.45})}{\sqrt{sm}(1 + M_i/M_j)} \right) \quad (10.172)$$

If the Lennard-Jones energy parameter, ϵ (in Kelvin), and the Stockmayer polar parameter, δ , are known, S is calculated from:

$$S = \frac{1 + \sqrt{(T/\epsilon_i)(T/\epsilon_j)} + \delta_i \delta_j / 4}{\sqrt{1 + T/\epsilon_i + \delta_i^2/4} \sqrt{1 + T/\epsilon_j + \delta_j^2/4}} \quad (10.173)$$

otherwise it is approximated by $S = 1$. ϵ and δ can be estimated from:

$$\epsilon = 65.3 T_{c,i} Z_{c,i}^{3.6} \quad (10.174)$$

$$\delta = 1.744 \times 10^{59} \frac{\mu^2}{V_{b,i} T_{b,i}} \quad (10.175)$$

where μ is the dipole moment in $C.m$ and the normal boiling volume is in $m^3/kmol$. Vapor viscosities are a function of pressure and a correction normally is applied. Mixture properties are computed using the 'normal' mixing rules. The high pressure viscosity can be obtained from:

$$\rho_c = 1/V_{c,m} \quad (10.176)$$

$$\rho_r = \rho/\rho_c \quad (10.177)$$

$$\xi = 2173.4241 T_{c,m}^{1/6} / \sqrt{M_m} P_{c,m}^{2/3} \quad (10.178)$$

$$A = \exp(1.4439 \rho_r) - \exp(-1.111 \rho_r^{1.85}) \quad (10.179)$$

$$B = 1.08 \times 10^{-7} A / \xi \quad (10.180)$$

$$\eta_{hp} = \eta + B \quad (10.181)$$

Table 10.4: Constants for the Yoon-Thodos method

Hydrogen	Helium	Others
a=47.65	a=52.57	a=46.1
b=0.657	b=0.656	b=0.618
c=20.0	c=18.9	c=20.4
d=-0.858	d=-1.144	d=-0.449
e=19.0	e=17.9	e=19.4
f=-3.995	f=-5.182	f=-4.058

where ρ is the vapor mixture molar density.

Both the Wilke and Brokaw methods require pure component viscosities. These are normally obtained from the vapor viscosity temperature correlations, as long as the temperature is within the valid temperature range. If not, then the viscosity can be computed with the Chapman-Enskog kinetic theory (see Hirschfelder *et al.* 1954 and A391-393):

$$T^* = T/\epsilon \quad (10.182)$$

$$\Omega_v = a(T^*)^{-b} + c/\exp(dT^*) + e/\exp(fT^*) \quad (10.183)$$

$$\eta^V = 26.69 \times 10^{-7} MT/\sigma^2 (\Omega_v + 0.2\delta^2/T^*) \quad (10.184)$$

where the collision integral constants are $a = 1.16145$, $b = 0.14874$, $c = 0.52487$, $d = 0.77320$, $e = 2.16178$, and $f = 2.43787$. The viscosity may also be computed with the Yoon and Thodos method:

$$\xi_i = 2173.4241 T_{c,i}^{1/6} / \sqrt{M_i} P_{c,i}^{2/3} \quad (10.185)$$

$$\eta_i^V = \frac{1 + aT_r^b - c \exp(dT - r) + e \exp(fT_r)}{10^8 \xi} \quad (10.186)$$

where the constants $a - f$ are given in Table 10.4.

Another method for calculating the vapor viscosity is the Lucas (A397) method:

$$\eta = 10^{-7} (0.807 T_r^{0.618} - 0.357 \exp(-0.449 T_r) + \quad (10.187)$$

$$0.340 \exp(-4.058 T_r) + 0.018) F_p^o F_q^o / \xi \quad (10.188)$$

$$\xi = 0.176 \left(\frac{T_c}{M^3 (10^{-5} P_c)^4} \right)^{1/6} \quad (10.189)$$

where F_p^o and F_q^o are polarity and quantum correction factors. The polarity correction

depends on the reduced dipole moment:

$$\mu_r = 52.46 \frac{(\mu/3.336 \times 10^{-30})^2 (10^{-5} P_c)}{T_c^2} \quad (10.190)$$

If μ_r is smaller than 0.022 then the correction factor is unity; if it is smaller than 0.075 it is given by

$$F_p^o = 1 + 30.55(0.292 - Z_c)^{1.72} \quad (10.191)$$

else by

$$F_p^o = 1 + 30.55(0.292 - Z_c)^{1.72} (0.96 + 0.1(T_r - 0.7)) \quad (10.192)$$

The quantum correction is only used for quantum gases He, H₂, and D₂,

$$F_q^o = 1.22Q^{0.15} \left(1 + 0.00385 (T_r - 12)^2 \right)^{1/M} \text{sign}(T_r - 12) \quad (10.193)$$

where $Q = 1.38$ (He), $Q = 0.76$ (H₂), $Q = 0.52$ (D₂). There is also a specific correction for high pressures (A421) by Lucas.

$$\eta = Y F_p F_q \eta^o \quad (10.194)$$

$$Y = 1 + \frac{a P_r^e}{b P_r^f + (1 + c P_r^d)^{-1}} \quad (10.195)$$

$$F_p = \frac{1 + (F_p^o - 1) Y^{-3}}{F_p^o} \quad (10.196)$$

$$F_q = \frac{1 + (F_q^o - 1) (Y^{-1} - 0.007(\ln Y)^4)}{F_q^o} \quad (10.197)$$

where η^o refers to the low-pressure viscosity (note that the original Lucas method has a different rule for Y if T_r is below unity, however, this introduces a discontinuity that is avoided here). The parameters a through f are evaluated with:

$$a = \frac{1.245 \times 10^{-3}}{T_r} \exp 5.1726 T_r^{-0.3286} \quad (10.198)$$

$$b = a(1.6553 T_r - 1.2723) \quad (10.199)$$

$$c = \frac{0.4489}{T_r} \exp 3.0578 T_r^{-37.7332} \quad (10.200)$$

$$d = \frac{1.7368}{T_r} \exp 2.2310 T_r^{-7.6351} \quad (10.201)$$

$$e = 1.3088 \quad (10.202)$$

$$f = 0.9425 \exp -0.1853 T_r^{0.4489} \quad (10.203)$$

where, in case T_r is below unity, T_r is taken to be unity. For mixtures the Lucas model uses the following mixing rules:

$$T_{cm} = \sum_{i=1}^c y_i T_{ci} \quad (10.204)$$

$$V_{cm} = \sum_{i=1}^c y_i V_{ci} \quad (10.205)$$

$$Z_{cm} = \sum_{i=1}^c y_i Z_{ci} \quad (10.206)$$

$$P_{cm} = RT_{cm} Z_{cm} / V_{cm} \quad (10.207)$$

$$M_m = \sum_{i=1}^c y_i M_i \quad (10.208)$$

$$F_{pm}^o = \sum_{i=1}^c y_i F_{pi}^o \quad (10.209)$$

$$F_{qm}^o = A \sum_{i=1}^c y_i F_{qi}^o \quad (10.210)$$

where A is a correction factor depending on the molecular weights of the components in the mixture. Let H denote the component of highest molecular weight and L of lowest, then if $M_H/M_L > 9$:

$$A = 1 - 0.01(M_H/M_L)^{0.57} \quad (10.211)$$

else $A = 1$. The mixture vapor viscosity is computed with the Lucas method as for a component that has the mixture properties T_{cm} , P_{cm} , M_m , F_{pm}^o , and F_{qm}^o . Therefore, the method is not interpolative in the same way as are the techniques of Wilke and Brokaw (that is, the method does not necessarily lead to pure component viscosity η_i when all $y_j = 0$ except $y_i = 1$).

10.2.7 Liquid Thermal Conductivity

The mixture liquid thermal conductivity, λ_m^L ($W/m K$), can be computed using the following methods from the component liquid thermal conductivities:

Molar average This is the default method (and the simplest):

$$\lambda_m^L = \sum_{i=1}^c x_i \lambda_i^L \quad (10.212)$$

DIPPR procedure 9I

$$F_{v,i} = x_i / \sum_{i=1}^c x_i / \rho_i^L \quad (10.213)$$

$$\lambda_{ij} = 2 / (1/\lambda_i + 1/\lambda_j) \quad (10.214)$$

$$\lambda_m^L = \sum_{i=1}^c \sum_{j=1}^c F_{v,i} F_{v,j} \lambda_{ij} \quad (10.215)$$

DIPPR procedure 9H

$$\frac{1}{\sqrt{\lambda_m^L}} = \sum_{i=1}^c \frac{w_i}{(\lambda_i^L)^2} \quad (10.216)$$

where w_i is the weight fraction of component i .

A correction is applied for high pressures:

$$\lambda_{hp} = (0.63T_r^{1.2}P_r / (30 + P_r) + 0.98 + 0.0079P_rT_r^{1.4}) \lambda \quad (10.217)$$

The mixture parameters are computed by the ‘normal’ mixing rules. Pure component liquid thermal conductivities are calculated from one of the following methods:

Polynomial The temperature correlation normally is used as long as the temperature is in the valid range and no other method is explicitly selected.

Pachaiyappan et al.

$$f = 3 + 20(1 - T_r)^{2/3} \quad (10.218)$$

$$b = 3 + 20(1 - 273.15/T_{c,i})^{2/3} \quad (10.219)$$

$$\lambda_i = c10^{-4}M_i^x\rho_i^L(f/b) \quad (10.220)$$

for straight chain hydrocarbons $c = 1.811$ and $x = 1.001$ else $c = 4.407$ and $x = 0.7717$.

Latini et al. (see A549,550):

$$\lambda_i^L = \frac{A(1 - T_r)^{0.38}}{T_r^{1/6}} \quad (10.221)$$

$$A = \frac{A^*T_b^\alpha}{M_i^\beta T_c^\gamma} \quad (10.222)$$

where parameters A^* , α , β , and γ depend on the class of the component as shown in Table 10.5.

Table 10.5: Parameters for the Latini equation for liquid thermal conductivity

Family	A^*	α	β	γ
Saturated hydrocarbons	0.0035	1.2	0.5	0.167
Olefins	0.0361	1.2	1.0	0.167
Cycloparaffins	0.0310	1.2	1.0	0.167
Aromatics	0.0346	1.2	1.0	0.167
Alcohols, phenols	0.00339	1.2	0.5	0.167
Acids (organic)	0.00319	1.2	0.5	0.167
Ketones	0.00383	1.2	0.5	0.167
Esters	0.0415	1.2	1.0	0.167
Ethers	0.0385	1.2	1.0	0.167
Refrigerants:				
R20, R21, R22, R23	0.562	0.0	0.5	-0.167
Others	0.494	0.0	0.5	-0.167

10.2.8 Vapor Thermal Conductivity

Molar average The mixture vapor thermal conductivity is computed from the pure component thermal conductivities as follows:

$$\lambda_m^V = \sum_{i=1}^c x_i \lambda_i^V \quad (10.223)$$

Kinetic theory This is DIPPR procedure 9D:

$$\lambda_m^V = \sum_{i=1}^c \frac{x_i \lambda_i^V}{\sum_{j=1}^c x_j \phi_{ij}} \quad (10.224)$$

where interaction parameters ϕ_{ij} are computed from:

$$\phi_{ij} = 0.25 \left(1 + \sqrt{\frac{\eta_i M_j^{3/4} T + 1.5 T_{b,i}}{\eta_j M_i} \frac{T + 1.5 T_{b,i}}{T + 1.5 T_{b,j}}} \right) \frac{T + \sqrt{1.5^2 T_{b,i} T_{b,j}}}{T + 1.5 T_{b,i}} \quad (10.225)$$

Note that the component viscosities are required for this evaluation.

If the system pressure is larger than 1 atmosphere a corection is applied. Mixture parameters are computed using the ‘normal’ mixing rules. Critical and reduced densities are computed from:

$$\rho_c = \frac{1}{V_{c,m}} \quad (10.226)$$

$$\rho_r = \frac{\rho}{\rho_c} \quad (10.227)$$

If the reduced density is below 0.5 then $a = 2.702$, $b = 0.535$, and $c = -1$; if the reduced density is within $[0.5, 2]$ then $a = 2.528$, $b = 0.67$, and $c = -1.069$; otherwise $a = 0.574$, $b = 1.155$, and $c = 2.016$. The high pressure thermal conductivity correction then is calculated from:

$$\Delta\lambda = \frac{a10^{-8}(\exp(b\rho_r) + c)}{\left(\frac{\sqrt{M_m}T_{c,m}^{1/6}}{P_{c,m}^{2/3}}\right)Z_{c,m}^5} \quad (10.228)$$

and added to the thermal conductivity calculated at low pressure.

Pure component vapor thermal conductivities are estimated from the following methods:

Polynomial The temperature correlation is normally used as long as the temperature is in the valid range and no other method is explicitly selected.

DIPPR procedure 9B-3 This method is the default in case the temperature is out of the range of the temperature correlation:

$$\lambda_i^V = (1.15(C_p - R) + 16903.36)\eta_i^V/M_i \quad (10.229)$$

DIPPR procedure 9B-2 This method is recommended for linear molecules:

$$\lambda_i^V = (1.3(C_p - R) + 14644 - 2928.8/T_r)\eta_i^V/M_i \quad (10.230)$$

DIPPR procedure 9B-1 This method is suitable for monatomic gases only:

$$\lambda_i^V = 2.5(C_p - R)\eta_i^V/M_i \quad (10.231)$$

Misic-Thodos 2 This method is used for methane and cyclic compounds below $T_r = 1$:

$$\xi = \frac{2173.424T_{c,i}^{1/6}}{\sqrt{M_i}P_{c,i}^{2/3}} \quad (10.232)$$

$$\lambda_i = 4.91 \times 10^{-7}T_r C_p / \xi \quad (10.233)$$

Misic-Thodos 1 This is the Misic-Thodos method for all other compounds:

$$\xi = \frac{2173.424T_{c,i}^{1/6}}{\sqrt{M_i}P_{c,i}^{2/3}} \quad (10.234)$$

$$\lambda_i = 11.05 \times 10^{-8}(14.52T_r - 5.14)^{1/6}C_p / \xi \quad (10.235)$$

10.2.9 Liquid Diffusivity

Generalized Maxwell-Stefan binary diffusion coefficients \mathcal{D}_{ij} are computed from the Kooijman-Taylor (1991) correlation

$$\mathcal{D}_{ij} = \prod_{k=1}^c (\mathcal{D}_{ij}^k)^{x_k} \quad (10.236)$$

where the D_{ij}^o are the diffusion coefficients at infinite dilution and where

$$\mathcal{D}_{ij}^k = D_{ij}^o, \quad k = j \quad (10.237)$$

$$\mathcal{D}_{ij}^k = D_{ji}^o, \quad k = i \quad (10.238)$$

$$\mathcal{D}_{ij}^k = \sqrt{D_{ik}^o D_{jk}^o}, \quad k \neq i, k \neq j \quad (10.239)$$

An alternative is the method of Wesselingh and Krishna (1990) where D_{ij}^k is calculated from:

$$\mathcal{D}_{ij}^k = \sqrt{D_{ij}^o D_{ji}^o}, \quad k \neq i, k \neq j \quad (10.240)$$

Liquid binary diffusion coefficients at infinite dilution (D_{ij}^o) normally are computed by the Wilke-Chang method unless selected otherwise. The following models are available:

Wilke-Chang The Wilke and Chang (1955, see A598) method is

$$D_{ab}^o = 1.1728 \times 10^{-16} \frac{\sqrt{\phi_b M_b T}}{\eta_b V_{b,a}^{0.6}} \quad (10.241)$$

where ϕ_b is the association factor for the solvent (2.26 for water, 1.9 for methanol, 1.5 for ethanol and 1.0 for unassociated solvents).

Hayduk-Laudie The Hayduk and Laudie (1974) method is

$$D_{aw}^o = 8.62 \times 10^{-14} \eta_w^{-1.14} V_{b,a}^{-0.589} \quad (10.242)$$

Hayduk-Minhas aqueous Estimates diffusivity of solute a in water, proposed by Hayduk and Minhas (1982, see also A602):

$$D_{aw}^o = (3.36 V_{b,a}^{-0.19} - 3.65) 10^{-13} (1000 \eta_w)^{(0.00958/V_{b,a} - 1.12)} T^{1.52} \quad (10.243)$$

Hayduk-Minhas non-aqueous Estimates diffusivity of solute a in polar and non-polar solvent b (not water), proposed by Hayduk and Minhas (1982, see also A603):

$$D_{ab}^o = 4.3637 \times 10^{-18} \eta_b^{-0.19} r_a^{0.2} r_b^{-0.4} T^{1.7} \quad (10.244)$$

Hayduk-Minhas paraffins Estimates diffusion coefficients for mixtures of normal paraffins from Hayduk-Minhas correlation (equation 7 of their paper) as corrected by Siddiqi and Lucas (1986, see also A602):

$$D_{ab}^o = 9.859 \times 10^{-14} V_{b,a}^{-0.71} (1000\eta_b)^{(0.0102/V_{b,a}-0.791)} \quad (10.245)$$

Siddiqi-Lucas aqueous Estimates diffusivity of solute a in water, proposed by Siddiqi and Lucas (1986):

$$D_{aw}^o = 5.6795 \times 10^{-16} V_{b,a}^{-0.5473} \eta_w^{-1.026} T \quad (10.246)$$

Siddiqi-Lucas Estimates diffusivity of solute a in solvent b (not water), proposed by Siddiqi and Lucas (1986):

$$D_{ab}^o = 5.2383 \times 10^{-15} V_{b,a}^{-0.45} V_{b,B}^{0.265} \eta_b^{-0.907} T \quad (10.247)$$

Umesi-Danner Estimates diffusivity of solute a in solvent b :

$$D_{ab}^o = 5.927 \times 10^{-12} \frac{Tr_b}{\eta_b r_a^{2/3}} \quad (10.248)$$

Tyn-Calus correlation Estimates diffusivity of solute a in solvent b (see A600):

$$D_{ab}^o = 8.93 \times 10^{-12} \left(\frac{V_{b,a}}{V_{b,B}^2} \right)^{1/6} \left(\frac{P_b}{P_a} \right)^{0.6} \frac{T}{\eta_b} \quad (10.249)$$

where P_i is the Parachor of the pure component (at T_{nB}).

Modified Tyn-Calus correlation This method uses the above correlation where the Parachor is calculated from:

$$P_i = V_{b,i} \sigma_i^{1/4} \quad (10.250)$$

The surface tension is computed with the Brock and Bird corresponding states method. This makes the Parachor temperature dependent, in contrast to the regular Tyn-Calus method.

10.2.10 Vapor Diffusivity

Generalized Maxwell-Stefan binary diffusion coefficients D_{ij} are equal to the normal binary diffusion coefficients (since the gas is considered an ideal system for which the thermodynamic matrix is the identity matrix). Normally these are computed using the Fuller-Schettler-Giddings method (see A587) but, if the Fuller volume parameters are missing, the Wilke-Lee modification of the Chapman-Enskog kinetic theory is used.

Table 10.6: Fuller diffusion volumes

Atomic and structural diffusion volume increments			
C	15.9	F	14.7
H	2.31	Cl	21.0
O	6.11	Br	21.9
N	4.54	I	29.8
Ring	-18.3	S	22.9
Diffusion volumes of simple molecules			
He	2.67	CO	18.0
Ne	5.98	CO2	26.9
Ar	16.2	N2O	35.9
Kr	24.5	NH3	20.7
Xe	32.7	H2O	13.1
H2	6.12	SF6	71.3
D2	6.84	Cl2	38.2
N2	18.5	Br2	69.0
O2	16.3	SO2	41.8
Air	19.7		

Fuller *et al.* This method was developed by Fuller *et al.* (1966,1969):

$$D_{ab}^V = 1.013 \times 10^{-2} \frac{T^{1.75} \sqrt{(1/M_a + 1/M_b)}}{P(\sqrt[3]{V_a} + \sqrt[3]{V_b})^2} \quad (10.251)$$

where V_a and V_b are the Fuller molecular diffusion volumes, calculated by summing the atomic contributions from Table 10.6. This table also includes the diffusion volumes for some simple molecules.

Chapman-Enskog kinetic theory The binary diffusion coefficient in gas/vapor mixtures may be estimated from a simplified kinetic theory.

$$D_{ab}^V = C \frac{T^{3/2} \sqrt{1/M_a + 1/M_b}}{P \sigma_{ab} \Omega_D} \quad (10.252)$$

where constant $C = 1.883 \times 10^{-2}$.

The average collision diameter and energy parameter are:

$$\sigma_{ab} = (\sigma_a + \sigma_b)/2 \quad (10.253)$$

$$\epsilon_{ab} = \sqrt{\epsilon_a + \epsilon_b} \quad (10.254)$$

The diffusion collision integral is

$$\Omega_D = a(T^*)^{-b} + c/\exp(dT^*) + e/\exp(fT^*) + g/\exp(hT^*) \quad (10.255)$$

where

$$T^* = T/\epsilon_{ab} \quad (10.256)$$

The collision integral constants are $a = 1.06036$, $b = 0.1561$, $c = 0.193$, $d = 0.47635$, $e = 1.03587$, $f = 1.52996$, $g = 1.76474$, and $h = 3.89411$. If the Stockmayer polar parameters are known the integral must be corrected as follows:

$$\Omega_{D,c} = \Omega_D + \frac{0.19\bar{\delta}_a\bar{\delta}_b}{T^*} \quad (10.257)$$

Wilke-Lee Wilke and Lee (1955, see A587) proposed a modified version of the kinetic theory method described above with

$$C = 2.1987 \times 10^{-2} - 5.07 \times 10^{-3} \sqrt{1/M_a + 1/M_b} \quad (10.258)$$

Saksena-Saxena Saksena and Saxena, Ind.Eng.Chem.Fundam., Vol. 11 (1972) p. 420

$$T_{c,ab} = \sqrt{T_{c,a}T_{c,b}} \quad (10.259)$$

$$D_{ij} = 1.2246D_{Fuller}/(1 + 0.4364\frac{T_{c,ab}}{T})/10.1325/T^{0.25} \quad (10.260)$$

Nain-Ferron Nain and Ferron, Ind.Eng.Chem.Fundam., Vol. 11 (1972) p. 420.

$$T_{c,ab} = \sqrt{T_{c,a}T_{c,b}} \quad (10.261)$$

$$D_{ij} = D_{Fuller}/(1 + 0.1759\frac{T_{c,ab}}{T}) \quad (10.262)$$

Slattery-Bird Slattery and Bird, AIChE J. 4 (1958) p. 137-142.

$$p_t = \frac{((1 \times 10^{-10}p_{c,a}p_{c,b})^{\frac{1}{3}})}{1 \times 10^{-5}p} \quad (10.263)$$

$$t_t = (T_{c,a}T_{c,b})^{(5/12 - 1.823/2)} \quad (10.264)$$

$$f_m = \frac{1}{M_a} + \frac{1}{M_b} \quad (10.265)$$

$$D_{ij} = 2.745 \times 10^{-8}T^{1.823}p_t t_t \sqrt{f_m} \quad (10.266)$$

10.2.11 Surface Tension

Mixture surface tension for VLE are computed with:

Molar average This is the default method:

$$\sigma_m = \sum_{i=1}^c x_i \sigma_i \quad (10.267)$$

Winterfeld et al. This method, by *Winterfeld et al.* (1978), is DIPPR Procedure 7C:

$$\sigma_m = \frac{\sum_{i=1}^c \left((x_i/\rho_i^L)^2 + \sum_{j=1}^c (x_i x_j \sqrt{\sigma_i \sigma_j} / \rho_i^L \rho_j^L) \right)}{\left(\sum_{i=1}^c (x_i/\rho_i^L) \right)^2} \quad (10.268)$$

Digulio-Teja This method evaluates the component surface tensions at the components normal boiling points ($\sigma_{b,i}$) and computes the mixture critical temperature, normal boiling temperature and the mixture surface tension at normal boiling temperature with the following mixing rules:

$$T_{c,m} = \sum_{i=1}^c x_i T_{c,i} \quad (10.269)$$

$$T_{b,m} = \sum_{i=1}^c x_i T_{b,i} \quad (10.270)$$

$$\sigma_{b,m} = \sum_{i=1}^c x_i \sigma_{b,i} \quad (10.271)$$

Then it corrects the $\sigma_{b,m}$ with:

$$T^* = \frac{(1/T_r - 1)}{(1/T_{rb} - 1)} \quad (10.272)$$

$$\sigma = 1.002855(T^*)^{1.118091} \frac{T}{T_b} \sigma_r \quad (10.273)$$

Pure component surface tensions are determined only for temperatures below the component's critical temperature, otherwise it is assumed that the component does not contribute to the mixture surface tension (i.e. $\sigma_i = 0$). The following methods are available:

Polynomial The temperature correlation is normally used as long as the temperature is in the valid range and no other method is explicitly selected.

Brock-Bird

$$T_{br} = T_{b,i}/T_{c,i} \quad (10.274)$$

$$Q = 0.1207 \left(1 + \frac{T_{br}(\ln(P_{c,i}) - 11.526)}{(1 - T_r)} \right) - 0.281 \quad (10.275)$$

$$\sigma_i = 4.6 \times 10^{-7} P_{c,i}^{2/3} T_{c,i}^{1/3} Q (1 - T_r)^{11/9} \quad (10.276)$$

Lielmezs-Herrick This method by Lielmezs and Herrick (1986) uses a polynomial but evaluates it at the reduced normal boiling temperature and corrects the resulting σ_r with:

$$T^* = \frac{(1/T_r - 1)}{(1/T_{rb} - 1)} \quad (10.277)$$

$$\sigma = 1.002855(T^*)^{1.118091} \frac{T}{T_b} \sigma_r \quad (10.278)$$

surface tension Sastri-Rao Sastri and Rao (Chem. Eng. J., 59 (1996) pp. 181-186

$$T_{b,r} = \frac{T_{b,oil}}{T_c} \quad (10.279)$$

$$T_m = \left(\frac{(T_c - T)}{(T_c - T_b)} \right)^{\frac{11}{9}} \quad (10.280)$$

$$\sigma = 0.158 \times 10^{-3} \sqrt{10^{-5} p_c T_{b,r}^{-1.85} T_b^0 .35 T_m} \quad (10.281)$$

10.2.12 Liquid-Liquid Interfacial Tension

This property is required only for simulating liquid-liquid extraction operations with the nonequilibrium model. The calculated surface tensions for both liquid phases and the interfacial tension, σ' , is computed from

$$\sigma' = \sigma_1 + \sigma_2 - 1.1\sqrt{\sigma_1\sigma_2} \quad (10.282)$$

This method generally over-predicts the interfacial tension for aqueous systems. We use a general method from Jufu *et al.* (1986):

$$X = -\ln(x_1'' + x_2' + x_{3r}) \quad (10.283)$$

$$\sigma' = \frac{KRTX}{A_{w0} \exp(X)(x_1''q_1 + x_2'q_2 + x_{3r}q_3)} \quad (10.284)$$

with $A_{w0} = 2.5 \times 10^5$ (m^2/mol), $R = 8.3144$ ($J/mol/K$), $K = 0.9414$, and q_i is the UNIQUAC surface area parameters of the components i . The components are ordered in such a manner that component 1 and 2 are the dominating components in the two liquid phases. Then the rest of the components are lumped into one mole fraction, x_3 . This lumped mole fraction is taken for the phase that has the largest x_3 (the richest). q_3 is the molar averaged q for that phase for all components except 1 and 2.

Symbol List

Latin Symbols

a, b	cubic EOS parameters
B	second virial coefficient ($m^3/kmol$)
c	number of components
C_p	mass heat capacity ($J/kg/K$)
D_{ij}	binary diffusion coefficient (m^2/s)
\bar{D}_{ij}	binary Maxwell-Stafan diffusion coefficient (m^2/s)
D_{ij}^0	infinite dilution binary diffusion coefficient (m^2/s)
K_i	K-value of component i , equilibrium ratio ($K_i = y_i/x_i$)
k_{ij}	binary interaction coefficient (for EOS)
M	molecular mass ($kg/kmol$)
R	gas constant = 8134 ($J/kmol/K$)
r	radius of gyration (<i>Angstrom</i>)
P	pressure (Pa)
P^*, P_{vap}	vapor pressure (Pa)
P_i	Parachor ($m^3kg^{1/4}/s^{1/2}$) of component i
PF	Poynting correction
q	UNIQUAC surface area parameter
T	temperature (K)
T_r	reduced temperature ($T_r = T/T_c$)
T_b	normal boiling temperature (K)
V	molar volume ($m^3/kmol$)
V_b	molar volume at normal boiling point ($m^3/kmol$)
V_s	saturated molar volume ($m^3/kmol$)
w	weight fraction (of component)
x	liquid mole fraction (of component)
y	vapor mole fraction (of component)
Z	compressibility
Z_R	Racket parameter

Greek Symbols

α	attractive parameter in EOS
ω	acentric factor
Ω_a, Ω_b	EOS parameters

Ω_v	vollision integral for viscosity
Ω_D	collision integral for diffusion
γ	activity coefficient
δ	Stockmayer parameter
ϵ	molecular energy parameter (K)
λ	thermal conductivity ($W/m/K$)
ρ	molar density ($kmol/m^3$)
η	viscosity ($Pa.s$)
ϕ_i	fugacity coefficient of component i
ϕ_s	association factor for solvent s (Hayduk-Laudie)
$\phi_{i,j}$	interaction parameter for viscosities
ϕ_i	fugacity coefficient of component i
ϕ^*	pure fugacity coefficient at saturation
σ	surface tension (N/m)
	collision diameter (<i>Angstrom</i>)
σ_b	surface tension at T_b (N/m)
σ_l	liquid-liquid interfacial tension (N/m)
μ	dipole moment (<i>Debye</i>)
ξ	inverse viscosity (defined in text)

Superscripts

L	liquid
V, G	vapor, gas
$*$	saturated liquid, T/ϵ

Subscripts

0	reference
b	at normal boiling point
c	critical
i	of component i
j	of component j
m	mixture
r	reduced
s	saturated liquid

Abbreviations

EOS	equation of state
RK	Redlich-Kwong
SRK	Soave Redlich-Kwong
PR	Peng-Robinson

References

Wikipedia contains several useful pages:

http://en.wikipedia.org/wiki/Debye-H%C3%BCckel_equation
http://en.wikipedia.org/wiki/Activity_%28chemistry%29
<http://en.wikipedia.org/wiki/Fugacity>
http://en.wikipedia.org/wiki/Chemical_potential

R.P. Danner and T.E. Daubert, *Manual for Predicting Chemical Process Design Data*, AIChE, New York (1983).

R. Digulio, A.S. Teja, "Correlation and Prediction of the Surface Tensions of Mixtures", *Chem. Eng. J.*, Vol. **38** (1988) pp. 205–208.

E.N. Fuller, K. Ensley, J.C. Giddings, "A New Method for Prediction of Binary Gas-Phase Diffusion Coefficients", *Ind. Eng. Chem.*, Vol. **58** (1966) pp. 19–27.

E.N. Fuller, P.D. Schettler, J.C. Giddings, "Diffusion of Halogenated Hydrocarbons in Helium. The Effect of Structure on Collision Cross sections", *J. Phys. Chem.*, Vol. **73** (1969) pp. 3679–3685.

W. Hayduk, H. Laudie, "Prediction of Diffusion Coefficients for Nonelectrolytes in Dilute Aqueous Solutions", *AIChE J.*, Vol. **20** (1974) pp. 611–615.

W. Hayduk, B.S. Minhas, "Correlations for Prediction of Molecular Diffusivities in Liquids", *Can. J. Chem. Eng.*, **60**, 295-299 (1982); Correction, Vol. **61** (1983) p. 132.

J.O. Hirschfelder, C.F. Curtis, R.B. Bird, *Molecular Theory of Gases and Liquids*, Wiley, New York (1954).

F. Jufu, L. Buqiang, W. Zihao, "Estimation of Fluid-Fluid Interfacial Tensions of Multicomponent Mixtures", *Chem. Eng. Sci.*, Vol. **41**, No. 10 (1986) pp. 2673–2679.

H.A. Kooijman, R. Taylor, "On the estimation of Diffusion Coefficients in Multicomponent Liquid Systems", *Ind. Eng. Chem. Res.*, Vol. **30**, No. 6 (1991) pp. 1217–1222.

A. Letsou, L.I. Stiel, "Viscosity of Saturated Nonpolar Liquids at Elevated Pressure", *AIChE J.*, Vol. **19** (1973) pp. 409–414.

J. Lielmezs, T.A. Herrick, "New Surface Tension Correlation for Liquids", *Chem. Eng. J.*, Vol. **32** (1986) pp. 165–169.

J.M. Prausnitz, T. Anderson, E. Grens, C. Eckert, R. Hsieh, J. O'Connell, *Computer Calculations for Multicomponent Vapor-Liquid and Liquid-Liquid Equilibria*, Prentice-Hall (1980).

J.S. Rowlinson, *Liquids and Liquid Mixtures*, 2nd Ed., Butterworth, London (1969).

R.C. Reid, J.M. Prausnitz, T.K. Sherwood, *Properties of Gases and Liquids*, 3rd Ed., McGraw-Hill, New York (1977).

R.C. Reid, J.M. Prausnitz and B.E. Poling, *The Properties of Gases and Liquids*, 4th Ed., McGraw-Hill, New York (1988).

M.A. Siddiqi, K. Lucas, "Correlations for Prediction of Diffusion in Liquids", *Can. J. Chem. Eng.*, Vol. **64** (1986) pp. 839–843.

R. Taylor, H.A. Kooijman, "Composition Derivatives of Activity Coefficient Models (For the estimation of Thermodynamic Factors in Diffusion)", *Chem. Eng. Comm.*, Vol. **102** (1991) pp. 87–106.

A.S. Teja, P. Rice, "Generalized Corresponding States Method for the Viscosities of Liquid Mixtures", *Ind. Eng. Chem. Fundam.*, Vol. **20** (1981) pp. 77–81.

S.M. Walas, *Phase Equilibria in Chemical Engineering*, Butterworth Publishers, London (1985).

C.R. Wilke, "A Viscosity Equation for gas Mixtures", *J. Chem. Phys.*, Vol. **18** (1950) pp. 517–519.

C.R. Wilke, P. Chang, "Correlation of Diffusion Coefficients in Dilute Solutions", *AIChE J.*, Vol. **1** (1955) pp. 264-270.

C.R. Wilke, C.Y. Lee, "Estimation of Diffusion Coefficients for Gases and Vapors", *Ind. Eng. Chem.*, Vol. **47** (1955) pp. 1253-1257.

P.H. Winterfeld, L.E. Scriven, H.T. Davis, "An Approximate Theory of Interfacial Tensions of Multicomponent Systems: Applications to Binary Liquid-Vapor Tensions", *AIChE J.*, Vol. **24** (1978) pp. 1010-1014.

Table 10.7: Temperature correlations (added)

Equation number	Parameter(s)	Formula
11	A	$\exp(A)$
12	A, B	$\exp(A + BT)$
13	A, C	$\exp(A + BT + CT^2)$
14	A, D	$\exp(A + BT + CT^2 + DT^3)$
15	A, E	$\exp(A + BT + CT^2 + DT^3 + ET^4)$
16	A, E	$A + \exp(B/T + C + DT + ET^2)$
17	A, E	$A + \exp(B + CT + DT^2 + ET^3)$
18	A, C	$A + B [1 + \ln T(1 + C/T)] \exp -C/T$
19	A, C	$A + BT \ln T \exp -C/T$
45	A, E	$AT + BT^2/2 + CT^3/3 + DT^4/4 + ET^5/5$
75	B, E	$B^2CT + 3DT^2 + 4ET^3$
118	A, D	$\exp A + B/T^E + C \ln T + DT^2$
120	A, C	$A - B/(T + C)$
121	A, E	$A + B/T + C \ln T + DT^E$
122	A, E	$A + B/T + C \ln T + DT^2 + E/T^2$
123	A, E	$A + B/T + CT + DT^2 + ET^3$
124	A, E	$A + B/T + C/T^2 + DT + ET^2$
125	A, E	$\exp A + B/T + C/t^2 + DT + ET^2$
130	A, E	$\exp A + B/T + C \ln T + DT + E/T^2$
131	A, E	$a + BT + \frac{C}{(T-D)}$
150	A, E	$a + BT + CT^2 + DT^3 + E/T^2$
200	A, E	$E \exp \left(\frac{[At + Bt^{1.5} + Ct^{2.5} + Dt^5]}{[1-t]} \right)$ with $t = 1 - T_r$
201	A, E	$E + \left(\frac{[At + Bt^{1.5} + Ct^{2.5} + Dt^5]}{[1-t]} \right)$ with $t = 1 - T_r$
206	A, E	$\exp \left[a + \frac{b}{(c+t+d.t^2+e.t^3)} \right]$ with $t = T_r$
207	A, C	see 10
208	A, C	$10^{(A-B/(T+C))}$
209	A, B	$10^{(A(1/T-1/B))}$
210	A, D	$10^{(A+B/T+Ct+dt^2)}$
211	A, E	$A \left(\frac{B-T}{B-C} \right)^D$
212	A, E	$\exp \left((E/T)(A(1 - T/E) + B(1 - T/E)^{1.5} + C(1 - T/E)^3 + D(1 - T/E)^6) \right)$
213	A, E	$\frac{A-B}{1+Ct+Dt^2+Et^3}$ with $t = T_r - 1$
221	A, E	$-B/T^2 + C/T + DET^{E-1}$
230	A, E	$-B/T^2 + C/T + D - 2E/T^3$
231	B, D	$B - C/(T - D)^2$

Chapter 11

Numerical Solution of Systems of Equations

In this chapter we discuss the numerical methods used in *ChemSep* to solve the nonlinear equations that model separation processes. For more in-depth treatment of numerical methods see, for example, Ortega and Rheinbolt (1970), Pozrikidis (1998), and Garcia and Zangwill (1981).

11.1 Repeated Substitution

To use repeated substitution the equations to be solved must be expressed in the form

$$\mathbf{G}(\mathbf{x}) = \mathbf{x} \quad (11.1)$$

where \mathbf{G} is a vector consisting of all the equations to be solved and \mathbf{x} is the vector of variables. The procedure for solving the equations is summarized in Table 11.1.

11.2 Newton's Method

Newton's method is one of the most versatile and robust methods for solving systems of nonlinear equations. It is, in fact, the method of choice in *ChemSep*. To use

Table 11.1: Repeated Substitution

1. Set iteration counter, k , to zero. Estimate \mathbf{x}_k
2. Compute $\mathbf{G}(\mathbf{x}_k)$
3. Set $\mathbf{x}_{k+1} = \mathbf{G}(\mathbf{x}_k)$
4. If not converged, increment k and return to step 2

Newton's method, the equations to be solved are written in the form

$$\mathbf{F}(\mathbf{x}) = \mathbf{0} \quad (11.2)$$

where \mathbf{F} is a vector consisting of all the equations to be solved and \mathbf{x} is, again, the vector of variables. A Taylor series expansion of the function vector around the point \mathbf{x}_0 at which the functions are evaluated gives

$$\mathbf{F}(\mathbf{x}) = \mathbf{F}(\mathbf{x}_0) + \mathbf{J}(\mathbf{x} - \mathbf{x}_0) + O(|\mathbf{x} - \mathbf{x}_0|^2) \quad (11.3)$$

where \mathbf{J} is the *Jacobian* matrix of partial derivatives of \mathbf{F} with respect to the independent variables \mathbf{x} :

$$J_{ij} = \frac{\partial F_i}{\partial x_j} \quad (11.4)$$

If \mathbf{x} is the actual solution to the system of equations, then $\mathbf{F}(\mathbf{x}) = \mathbf{0}$. Additionally, if the $O(|\mathbf{x} - \mathbf{x}_0|^2)$ term is neglected then equation (11.3) can be rearranged to give

$$\mathbf{J}(\mathbf{x} - \mathbf{x}_0) = -\mathbf{F}(\mathbf{x}_0) \quad (11.5)$$

Here the only unknown is the value of the vector \mathbf{x} . Thus, by solving the linear system of equations (11.5) it is possible to obtain a value for this vector. If the new vector, \mathbf{x} , obtained in this way does not actually satisfy the set of equations, \mathbf{F} , then the procedure can be repeated using the just calculated \mathbf{x} as a new \mathbf{x}_0 . The entire procedure is summarized in Table 11.2.

Newton's method works extremely well if the initial estimate of the solution vector is "near" the actual solution. Outside this "near" region, the method may converge slowly, oscillate, or diverge. The question to which there is no simple answer is, "What exactly is it near?". Having said this it must also be stated that Newton's method is a much more reliable procedure than repeated substitution.

An important issue not addressed here is how to tell when the problem has been solved; that is, when convergence has been obtained. A proper discussion of this topic is beyond the scope of this work. What actually is done in *ChemSep* is discussed briefly in later chapters.

Table 11.2: Newton's Method

1. Set iteration counter, k , to zero. Estimate \mathbf{x}_k
2. Compute $\mathbf{F}(\mathbf{x}_k)$ and $\mathbf{J}(\mathbf{x}_k)$
3. Solve Equation (11.5) for $\mathbf{x}_k + 1$
4. Check for convergence. If not converged, increment k and return to step 2

11.2.1 Quasi-Newton Methods

To evaluate the Jacobian, one must obtain the partial derivative of each function with respect to every variable. This can be a time consuming step even if many of these derivatives are known to be zero. In order to reduce the time required to calculate the necessary derivative information for the Jacobian, a class of methods, termed quasi-Newton methods, has been developed. In these methods, approximations to the Jacobian are made and updated through the use of formulae derived to satisfy certain constraints which have been chosen to force the approximate Jacobian to mimic the behavior of the actual Jacobian. One such update formula is given by Broyden (1965):

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \frac{(\mathbf{y}_k - \mathbf{B}_k \mathbf{s}_k) \mathbf{s}_k^T}{\mathbf{s}_k^T \mathbf{s}_k} \quad (11.6)$$

where \mathbf{B}_k is the k -th approximation to the Jacobian; \mathbf{s}_k is the k -th change to the variable vector, $(\mathbf{x}_{k+1} - \mathbf{x}_k)$; \mathbf{y}_k is the k -th change to the function vector, $(\mathbf{F}(\mathbf{x}_{k+1}) - \mathbf{F}(\mathbf{x}_k))$. These methods, which make use of the step changes in the function and variable vectors, may also be designed to preserve known sparsity patterns within the matrix (Schubert, 1970).

Although these methods can considerably reduce the amount of computer time necessary to evaluate the Jacobian, they also generally increase the number of iterations necessary to obtain convergence to an extent where the quasi-Newton solution procedure may very well end up taking more time than the unaltered Newton's method. Furthermore, the methods tend to reduce the domain of convergence from that of Newton's method. That is, even though Newton's method worked from a given initial estimate of the solution vector, a quasi-Newton method may fail.

11.3 Homotopy-Continuation Methods

Homotopy continuation methods form a class of methods for solving difficult systems of nonlinear algebraic equations. In addition they can be used for tracking the solutions to nonlinear algebraic equations in terms of a parameter that occurs in the equations. The basic idea is to follow a path that is traced from a solution to a simple problem to the solution of the difficult problem.

The homotopy path is implicitly defined by the homotopy equation:

$$\mathbf{H}(\mathbf{x}, t) = \mathbf{0} \quad (11.7)$$

This represents a set of n continuously differentiable equations in terms of $n + 1$ variables.

For the systems under consideration here, the homotopy function is a continuous blending of two functions $\mathbf{F}(\mathbf{x})$ and $\mathbf{G}(\mathbf{x})$, in terms of a parameter t . In the following, $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ represents the original - presumably difficult - problem, $\mathbf{G}(\mathbf{x}) = \mathbf{0}$ is the easy problem. The class of equations we will be looking at is that of *linear convex homotopies* given by:

$$\mathbf{H}(\mathbf{x}, t) = t \cdot \mathbf{F}(\mathbf{x}) + (1 - t) \cdot \mathbf{G}(\mathbf{x}) \quad (11.8)$$

This equation reduces to $\mathbf{G}(\mathbf{x}) = \mathbf{0}$ for $t = 0$ and to $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ for $t = 1$. If $\mathbf{F}(\mathbf{x})$ and $\mathbf{G}(\mathbf{x})$ are n -dimensional problems, then, as will be shown later, $\mathbf{H}(\mathbf{x}, t)$ describes a curve in $n + 1$ dimensional space going through $\mathbf{G}(\mathbf{x}) = \mathbf{0}$ and $\mathbf{F}(\mathbf{x}) = \mathbf{0}$. The choice of $\mathbf{G}(\mathbf{x})$ can be quite arbitrary, as long as the problem has the same dimension as the original problem. Most commonly used is the Newton homotopy method, where $\mathbf{G}(\mathbf{x})$ is given by:

$$\mathbf{G}(\mathbf{x}) = \mathbf{F}(\mathbf{x}) - \mathbf{F}(\mathbf{x}_0) \quad (11.9)$$

Leading to the homotopy equation:

$$\mathbf{H}(\mathbf{x}, t) = \mathbf{F}(\mathbf{x}) - (1 - t) \cdot \mathbf{F}(\mathbf{x}_0) \quad (11.10)$$

Here \mathbf{x}_0 is an arbitrary starting guess for $\mathbf{F}(\mathbf{x})$. This method does not require us to solve an easy problem first, because the solution is already known (\mathbf{x}_0). Another important feature is that by specifying different starting guesses, we can generate different paths that can lead to different solutions. This way multiple solutions to $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ (if they exist) can be found with relative ease.

A second method that has the same convenient features is the fixed point homotopy method. In this case $\mathbf{G}(\mathbf{x})$ is given by:

$$\mathbf{G}(\mathbf{x}) = (\mathbf{x} - \mathbf{x}_0) \quad (11.11)$$

The homotopy equation is then given by:

$$\mathbf{H}(\mathbf{x}, t) = t \cdot \mathbf{F}(\mathbf{x}) + (1 - t)(\mathbf{x} - \mathbf{x}_0) \quad (11.12)$$

A third approach is the so-called ‘linear’ homotopy. Here $\mathbf{G}(\mathbf{x})$ can be a simplified version of the problem posed by $\mathbf{F}(\mathbf{x}) = \mathbf{0}$. Often, in large big systems of nonlinear equations, the number of equations that actually cause problems is quite limited. By replacing the equations causing the problems by simpler ones, a system of equations may be obtained that is solved with relative ease. This can then serve as our starting point for the homotopy curve.

11.3.1 Existence of Paths

As stated before, the homotopy equation describes a curve in $n + 1$ dimensional space going through the zero’s of $\mathbf{F}(\mathbf{x})$ and $\mathbf{G}(\mathbf{x})$. Therefore, the solution to $\mathbf{F}(\mathbf{x}) = \mathbf{0}$ can be obtained by following the curve from $t = 0$ and $\mathbf{G}(\mathbf{x}) = \mathbf{0}$ to $t = 1$. How this is done exactly will be treated later. We will follow the proof given by Garcia and Zangwill (1981) to show that such a path exists.

A homotopy equation is given by: $\mathbf{H} : R^{n+1} \rightarrow R^n$ and we are interested in the solutions to:

$$\mathbf{H}(\mathbf{x}, t) = \mathbf{0} \quad (11.13)$$

These solutions are given by:

$$\mathbf{H}^{-1} = \{(\mathbf{x}, t) | \mathbf{H}(\mathbf{x}, t) = \mathbf{0}\} \quad (11.14)$$

This is the set of all solutions $(\mathbf{x}, t) \in R^{n+1}$ for which $\mathbf{H}(\mathbf{x}, t) = \mathbf{0}$. This does not say anything about the location of (\mathbf{x}, t) and these points could be anywhere. They are, however, included in \mathbf{H}^{-1} . The key to making sure that \mathbf{H}^{-1} consists solely of paths is given by the *implicit function theorem*. For that we will examine the solutions in \mathbf{H}^{-1} more precisely. We take an arbitrary point $(\tilde{\mathbf{x}}, \tilde{t}) \in \mathbf{H}^{-1}$ so that:

$$\mathbf{H}(\tilde{\mathbf{x}}, \tilde{t}) = \mathbf{0} \quad (11.15)$$

If \mathbf{H} is continuously differentiable, we can make a linear approximation near $(\tilde{\mathbf{x}}, \tilde{t})$:

$$H_i(\mathbf{x}, t) \cong H_i(\tilde{\mathbf{x}}, \tilde{t}) + \sum_{j=1}^n \frac{\partial H_i(\tilde{\mathbf{x}}, \tilde{t})}{\partial x_j} \cdot (x_j - \tilde{x}_j) + \frac{\partial H_i(\tilde{\mathbf{x}}, \tilde{t})}{\partial t} \cdot (t - \tilde{t}) \quad (11.16)$$

or, in vector form:

$$\mathbf{H}(\mathbf{x}, t) = \mathbf{H}_x \cdot (\mathbf{x} - \tilde{\mathbf{x}}) + \mathbf{H}_t \cdot (t - \tilde{t}) \quad (11.17)$$

where \mathbf{H}_x is a Jacobian matrix defined by

$$\mathbf{H}_x = \begin{bmatrix} \frac{\partial H_1}{\partial x_1} & \cdots & \frac{\partial H_1}{\partial x_n} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \frac{\partial H_n}{\partial x_1} & \cdots & \frac{\partial H_n}{\partial x_n} \end{bmatrix} \quad (11.18)$$

and

$$\mathbf{H}_t = \begin{pmatrix} \frac{\partial H_1}{\partial t} \\ \cdot \\ \cdot \\ \frac{\partial H_n}{\partial t} \end{pmatrix} \quad (11.19)$$

Now we want to show that the points (\mathbf{x}, t) in \mathbf{H}^{-1} that are near $(\tilde{\mathbf{x}}, \tilde{t})$ are on a path through $(\tilde{\mathbf{x}}, \tilde{t})$. By definition we find that all points in \mathbf{H}^{-1} must satisfy:

$$\mathbf{H}(\mathbf{x}, t) = \mathbf{0} \quad (11.20)$$

Assuming that \mathbf{H} is actually linear near $(\tilde{\mathbf{x}}, \tilde{t})$ we get from Eq. (11.17) and Eq. (11.20):

$$\mathbf{0} = \mathbf{H}_x \cdot (\mathbf{x} - \tilde{\mathbf{x}}) + \mathbf{H}_t \cdot (t - \tilde{t}) \quad (11.21)$$

Assuming that \mathbf{H}_x is invertible at $\tilde{\mathbf{x}}, \tilde{t}$ we get:

$$(\mathbf{x} - \tilde{\mathbf{x}}) = -\mathbf{H}_x^{-1} \cdot \mathbf{H}_t \cdot (t - \tilde{t}) \quad (11.22)$$

This is a system of n linear equations and $n + 1$ variables (also referred to as the Homotopy Differential Equations, HDE's, when written as $(dx/dt) = -\mathbf{H}_x^{-1}\mathbf{H}_t$). The solutions must therefore form a straight line. In general \mathbf{H} will not be linear, but \mathbf{H} can be approximated by a linearised system as done above. And as long as \mathbf{H} is continuous differentiable and \mathbf{H}_x is invertible the homotopy equation describes a path through $(\tilde{\mathbf{x}}, \tilde{t})$. In addition, this path will be smooth and continuously differentiable. All this is summarized by the implicit function theorem:

IMPLICIT FUNCTION THEOREM

Let $\mathbf{H} : R^{n+1} \rightarrow R^n$ be continuously differentiable, $(\tilde{\mathbf{x}}, \tilde{t}) \in \mathbf{H}$ and \mathbf{H}_x are invertible. Thus, in the neighborhood of $(\tilde{\mathbf{x}}, \tilde{t})$ all points (\mathbf{x}, t) that satisfy $\mathbf{H}(\mathbf{x}, t) = \mathbf{0}$ are on a single continuously differentiable path through $(\tilde{\mathbf{x}}, \tilde{t})$

We will now use this theorem for determining a method for following the path. First we will replace the original variables (\mathbf{x}, t) by \mathbf{y} . The homotopy equation can, therefore, be written as

$$\mathbf{H}(\mathbf{y}) = \mathbf{0} \quad (11.23)$$

Next, we define the partial Jacobian \mathbf{H}_{-i} to be the $n \times n$ matrix that is formed after removing the i -th column from the original Jacobian matrix:

$$\mathbf{H}_{-i} = \begin{bmatrix} \frac{\partial H_1}{\partial y_1} & \cdots & \frac{\partial H_1}{\partial y_{i-1}} & \frac{\partial H_1}{\partial y_{i+1}} & \cdots & \frac{\partial H_1}{\partial y_{n+1}} \\ \cdot & & \cdot & \cdot & & \cdot \\ \cdot & & \cdot & \cdot & & \cdot \\ \frac{\partial H_n}{\partial y_1} & \cdots & \frac{\partial H_n}{\partial y_{i-1}} & \frac{\partial H_n}{\partial y_{i+1}} & \cdots & \frac{\partial H_n}{\partial y_{n+1}} \end{bmatrix} \quad (11.24)$$

Note here that

$$\mathbf{H}_x = \mathbf{H}_{-t} = \mathbf{H}_{-(n+1)} \quad (11.25)$$

According to the implicit function theorem, a single continuously differentiable path exists for $(\mathbf{x}, t) \in \mathbf{H}^{-1}$ in the neighborhood of $(\tilde{\mathbf{x}}, \tilde{t})$. A unique path is guaranteed by this theorem, although several \mathbf{H}_{-i} may be invertible. Proof of this is relatively simple and is left as an exercise to the reader. At least one invertible \mathbf{H}_{-i} matrix can be found if the original $(n+1) \times n$ matrix is of full rank. This leads to the following theorem:

PATH THEOREM

Let $H : R^{n+1} \rightarrow R^n$ be continuously differentiable and suppose that for every $y \in \mathbf{H}^{-1}$, the jacobian is of full rank, then \mathbf{H}^{-1} consists only of continuously differentiable paths.

This theorem forms the basis of the proof that paths exist. This is quite convenient, since if we want to follow a path, it would be nice to know that it actually exists.

11.3.2 Path Following Methods

In the next section we will focus on the Newton homotopy method. Similar derivations can be made for other methods. Recalling the original Newton homotopy equation

$$\mathbf{H}(\mathbf{x}, t) = \mathbf{F}(\mathbf{x}) - (1-t) \cdot \mathbf{F}(\mathbf{x}_0) \quad (11.26)$$

The simplest approach to following the path is given by the following method:

Algorithm 1

1. Set initial values for \mathbf{x}_0
2. Increase value of t

3. Evaluate $\mathbf{H}(\mathbf{x}, t)$
4. Solve $\mathbf{H}(\mathbf{x}, t)$, using \mathbf{x}_0 as a starting guess
5. Update solution to \mathbf{x}_0
6. If $t = 1$, return, else go to 2

This approach might work for simple problems, but is somewhat naïve. It does not allow for the homotopy path to turn back on itself. What this means will be shown later.

A better approach for curve tracking (where we are actually following a path) is given by the differential arc length continuation method. The basis of this lies in the implicit function theorem. We have shown that $\mathbf{H}(\mathbf{x}, t)$ represents a path in $n + 1$ dimensional space and, in case we want to find the solution to $\mathbf{F}(\mathbf{x})$, we only have to follow that path from $t = 0$ and $\mathbf{G}(\mathbf{x}) = \mathbf{0}$ to $t = 1$ and $\mathbf{F}(\mathbf{x}) = \mathbf{0}$.

In other words, we have to evaluate how the variables change along the curve, and correct each accordingly (including the continuation parameter t), as we walk along the curve. A natural way of doing this is by differentiating the homotopy equations with respect to the arc length of the homotopy curve. This results in the following equation (for the Newton homotopy method):

$$\frac{\partial \mathbf{H}(\mathbf{x}, t)}{\partial s} = \frac{\partial \mathbf{F}(\mathbf{x})}{\partial \mathbf{x}} \cdot \left(\frac{\partial \mathbf{x}}{\partial s} \right) + \mathbf{F}(\mathbf{x}_0) \cdot \left(\frac{\partial t}{\partial s} \right) \quad (11.27)$$

The changes of the variables are restricted by the change in the arc length: a multidimensional Pythagorean theorem:

$$(dx_1)^2 + (dx_2)^2 + \dots + (dx_n)^2 + (dt)^2 = (ds)^2 \quad (11.28)$$

Which can be rewritten to:

$$\left(\frac{\partial x_1}{\partial s} \right)^2 + \left(\frac{\partial x_2}{\partial s} \right)^2 + \dots + \left(\frac{\partial t}{\partial s} \right)^2 = 1 \quad (11.29)$$

There are a few things to be noted here. $\mathbf{F}(\mathbf{x}_0)$ is an n -dimensional vector and may be added to the Jacobian matrix leading to the following expression:

$$\frac{\partial \mathbf{H}(\mathbf{x}, t)}{\partial s} = \begin{bmatrix} \frac{\partial F_1}{\partial x_1} & \dots & \frac{\partial F_1}{\partial x_n} & F_1(\mathbf{x}_0) \\ \cdot & & \cdot & \cdot \\ \cdot & & \cdot & \cdot \\ \frac{\partial F_n}{\partial x_1} & \dots & \frac{\partial F_n}{\partial x_n} & F_n(\mathbf{x}_0) \end{bmatrix} \cdot \begin{pmatrix} \frac{\partial x_1}{\partial s} \\ \cdot \\ \frac{\partial x_n}{\partial s} \\ \frac{\partial t}{\partial s} \end{pmatrix} = \mathbf{0} \quad (11.30)$$

This is where the importance of the path theorem comes into its own. In order to be able to solve the above system of equations, the matrix should be of full rank. Then we can choose a partial Jacobian and solve the system of equations that results:

$$\begin{bmatrix} \frac{\partial F_1}{\partial x_1} & \cdots & \frac{\partial F_1}{\partial x_{i-1}} & \frac{\partial F_1}{\partial x_{i+1}} & \cdots & \frac{\partial F_1}{\partial x_n} & F_1(\mathbf{x}_0) \\ \cdot & & \cdot & \cdot & & \cdot & \cdot \\ \cdot & & \cdot & \cdot & & \cdot & \cdot \\ \frac{\partial F_n}{\partial x_1} & \cdots & \frac{\partial F_n}{\partial x_{i-1}} & \frac{\partial F_n}{\partial x_{i+1}} & \cdots & \frac{\partial F_n}{\partial x_n} & F_n(\mathbf{x}_0) \end{bmatrix} \cdot \begin{pmatrix} \frac{\partial x_1}{\partial s} \\ \cdot \\ \frac{\partial x_{i-1}}{\partial s} \\ \frac{\partial x_{i+1}}{\partial s} \\ \cdot \\ \frac{\partial x_n}{\partial s} \\ \frac{\partial t}{\partial s} \end{pmatrix} = - \begin{pmatrix} \frac{\partial F_1}{\partial x_i} \\ \cdot \\ \frac{\partial F_n}{\partial x_i} \end{pmatrix} \cdot \frac{\partial x_i}{\partial s} \quad (11.31)$$

Note here that x_i is the independent variable. This system may be solved using any sparse matrix solver to give:

$$\begin{bmatrix} 1 & & 0 \\ \cdot & & \cdot \\ \cdot & & \cdot \\ 0 & & 1 \end{bmatrix} \cdot \begin{pmatrix} \frac{\partial x_1}{\partial s} \\ \cdot \\ \frac{\partial x_{i-1}}{\partial s} \\ \frac{\partial x_{i+1}}{\partial s} \\ \cdot \\ \frac{\partial x_n}{\partial s} \\ \frac{\partial t}{\partial s} \end{pmatrix} = - \begin{bmatrix} \beta_1 \\ \cdot \\ \beta_{i-1} \\ \beta_{i+1} \\ \cdot \\ \beta_n \\ \beta_{n+1} \end{bmatrix} \cdot \frac{\partial x_i}{\partial s} \quad (11.32)$$

This is, in fact, an expression for the changes of all variables along the curve in terms of the change in x_i . The total change in x_i along the curve can now be evaluated using equation Eq. (11.29) rewritten as:

$$\left(\frac{\partial x_i}{\partial s}\right)^2 = 1 - \left(\frac{\partial x_1}{\partial s}\right)^2 - \cdots - \left(\frac{\partial x_{i-1}}{\partial s}\right)^2 - \left(\frac{\partial x_{i+1}}{\partial s}\right)^2 - \cdots - \left(\frac{\partial x_n}{\partial s}\right)^2 - \left(\frac{\partial t}{\partial s}\right)^2 \quad (11.33)$$

Now we can substitute Eq. (11.32) into this equation and solve for the derivative of x_i giving:

$$\frac{\partial x_i}{\partial s} = \frac{1}{\sqrt{1 + \sum_{k=1, k \neq i}^{n+1} \beta_k^2}} \quad (11.34)$$

Substituting Eq. (11.34) back into Eq. (11.32) then results in a system of ordinary differential equations that can be integrated using some numerical integration method.

Before we do this however we should pay some attention to selection criteria for the independent variable. Let us assume that we have not made a good choice for the independent variable so that after solving the matrix equation we end up with a matrix

that looks like this

$$\begin{bmatrix} 1 & & c_1 \\ & \cdot & \\ & & 1 & c_{n-1} \\ 0 & 0 & 0 & \end{bmatrix} \cdot \begin{pmatrix} \frac{\partial x_1}{\partial s} \\ \cdot \\ \frac{\partial x_{i-1}}{\partial s} \\ \frac{\partial x_{i+1}}{\partial s} \\ \cdot \\ \frac{\partial x_n}{\partial s} \\ \frac{\partial t}{\partial s} \end{pmatrix} = - \begin{bmatrix} \beta_1 \\ \cdot \\ \beta_{i-1} \\ \beta_{i+1} \\ \cdot \\ \beta_n \\ \beta_{n+1} \end{bmatrix} \cdot \frac{\partial x_i}{\partial s} \quad (11.35)$$

This is a singular matrix. This system cannot be solved for the independent variable. We can, however, swap the last column of the matrix with the residual vector, thereby obtaining the following equation system:

$$\begin{bmatrix} 1 & & \beta_1 \\ & \cdot & \\ & & 1 & \beta_n \\ 0 & 0 & 0 & \beta_{n+1} \end{bmatrix} \cdot \begin{pmatrix} \frac{\partial x_1}{\partial s} \\ \cdot \\ \frac{\partial x_n}{\partial s} \\ \frac{\partial x_i}{\partial s} \end{pmatrix} = - \begin{bmatrix} c_1 \\ \cdot \\ c_n \\ 0 \end{bmatrix} \cdot \frac{\partial t}{\partial s} \quad (11.36)$$

This system can be solved very easily, and here we can also see what a singular Jacobian means: The independent variable does not change along the curve at this point on the curve. We must keep this in mind when picking the independent variable. It is preferable to choose the variable that changes the most along the curve at this point. In practice we will use the variable that changed the most at the previous point. This should be a good enough approximation since it is very unlikely that the parameter that changed the most in the previous step will not change at all in the following step.

Integrating the system of differential equations over (part of) the curve will give us an estimate of the following point on the curve. Often, Euler's method is used.

$$\mathbf{x}^{n+1} = \mathbf{x}^n + h \cdot \left(\frac{\partial \mathbf{x}}{\partial s} \right) \quad (11.37)$$

Where h is the integration stepsize. It is likely that this new estimate is somewhat off the homotopy curve. Usually a Newton correction step is used to get the estimate back on the line, before the next extrapolation step is done.

A possible algorithm for a Newton homotopy method is given by:

1. Calculate the function vector
2. Calculate Jacobian.
3. Choose an independent variable (typically t for the first step)

4. Calculate the derivatives of the variables with respect to the arc length
5. Determine the variable that changes most to use as independent variable next step
6. Integrate the differential equations using Eulers method
7. Correct the newly obtained estimate with Newton's method
8. Adjust the stepsize of the integration.
9. If t is smaller than 1, go to 4.

Step 8 in this algorithm has not yet had any attention, and a few things must also be said with respect to item 6.

In Step 6 we use Euler's method for the integration of the system of differential equations. This is the simplest method for integrating a system of differential equations, but it is not the most accurate method. However, whereas accuracy is important in solving ODE's, the errors in the continuation methods are determined solely by the termination criterion of the corrector step. Since we are following a curve, we need only a global estimate of where the next point lies on the curve, and by a correction step, this estimated point may be moved arbitrarily close to the curve. Thus, we need only to make sure that the estimate is within the domain of convergence for the corrector - usually Newton's - method). Higher order integration methods are therefore not often used, as the cost of calculation increases considerably with these methods, and the advantage often is negligible, because the error caused by the integration is reduced in the correction step.

11.3.3 Step Sizes

The most simple minded step size strategy would be to specify a fixed step size. The drawback of this method is that the step size has to be small enough to get through highly curved regions of the homotopy path. However, this strategy leads to an unnecessarily large number of steps in the straighter sections. A benefit of this method is that no calculation time is lost in calculating a new stepsize.

We would like to specify the step size in such a way that the number of integration steps and correction iterations is minimized simultaneously. However there is a clear tradeoff here: A very small stepsize will result in estimates that will be relatively close to the homotopy curve and therefore we do not need many correction steps per integration step. On the other hand, specifying a large stepsize will reduce the number

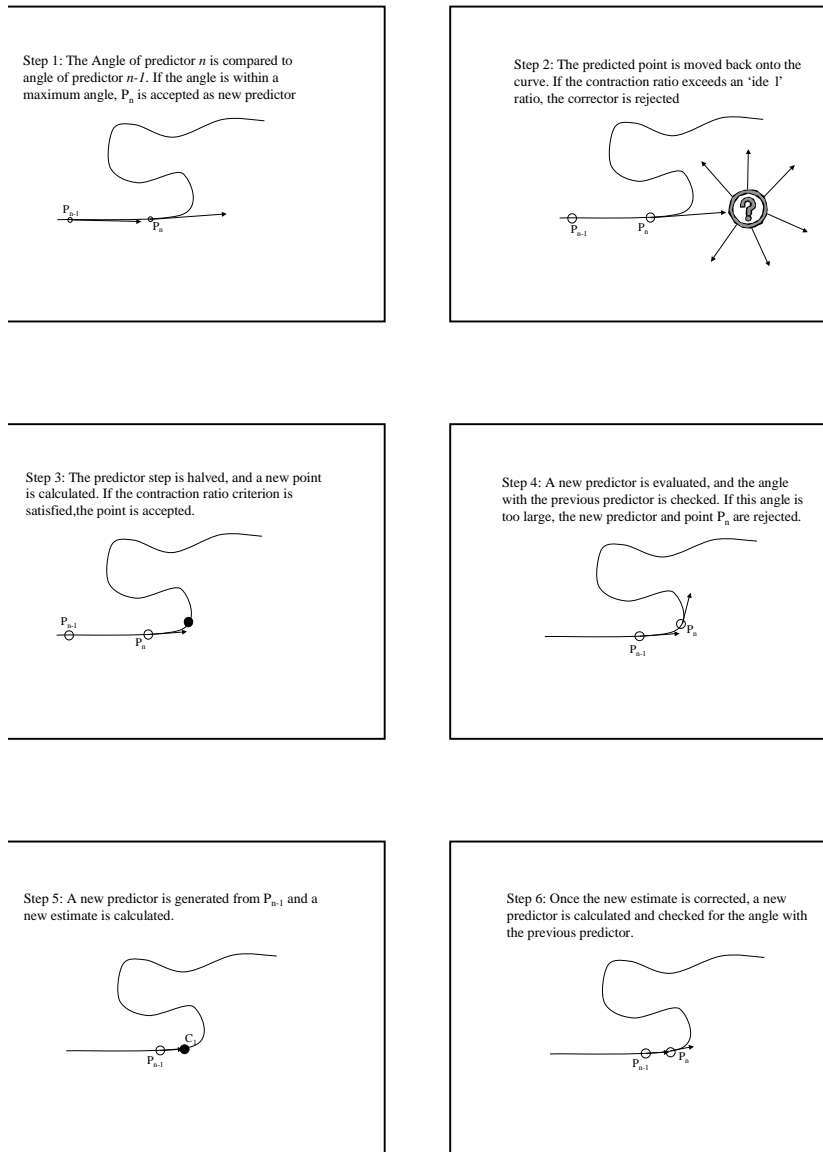


Figure 11.1: Stepsize for homotopy method.

of steps to be taken. The expected number of correction steps will, however, increase as each new estimated point will probably be further away from the curve. In some

cases they might even move outside the domain of convergence for Newton's method.

A principal aspect of a continuation method should be a technique for assessing if the corrector may be expected to converge when started from the predicted point. Den Heijer and Rheinboldt (1981), however, proved that this is theoretically impossible. At best we can find a method that maximizes the probability of convergence, while allowing for the occasional failure in the corrector. This means that the predictor will have to be adjusted.

The algorithm for choosing step sizes that is implemented in *ChemSep* is similar to the one used by Woodman (1989). The algorithm is based on two criteria: (1) the angle of two successive predictors (2), the ratio of convergence of the corrector iteration.

In the Euler integration step, the a new estimate of a point on the curve is generated by adding the predictor to the point found after the previous corrector:

$$\mathbf{x}^{n+1} = \mathbf{x}^n + h^{n+1} \cdot \left(\frac{\partial \mathbf{x}}{\partial s} \right)^{n+1} \quad (11.38)$$

where n represents the step counter. The stepsize h^{n+1} is determined from:

$$h^{n+1} = \beta \cdot h^n \quad (11.39)$$

where the β is calculated from:

$$\beta = \min \left\{ 2, \sqrt{\frac{\alpha_I^n}{\alpha_1^n}}, \frac{\theta_I}{\theta^n} \right\} \quad (11.40)$$

α_1^n (the contraction ratio) and θ^n (the predictor angle change) are defined as

$$\alpha_1^n = \frac{\epsilon_{res,1}^n}{\epsilon_{res,0}^n} \quad (11.41)$$

and

$$\theta^n = \arccos \left[\left(\frac{\partial \mathbf{x}^{n+1}}{\partial s} \right)^T \left(\frac{\partial \mathbf{x}^n}{\partial s} \right) \right] \quad (11.42)$$

In the definition of α_1^n , $\epsilon_{res,i}^n$ is the residual error of the equations at the end of the i -th Newton corrector during the n -th predictor-corrector step. α_I^n is an 'ideal' contraction ratio, given by:

$$\alpha_I^n = (\epsilon_{res,0}^n)^{p-1} \cdot \left(\frac{(\epsilon_f)^m}{(\epsilon_{res,0}^n)^{p^m}} \right)^{\frac{p-1}{p^m-1}} \quad (11.43)$$

where p is the assumed order of convergence (2 for second order convergence, and typically 1.5 for superlinear convergence), ϵ_f is the maximum required error for convergence of the Newton step, and m is the maximum number of Newton iterations to

convergence. In addition, the algorithm is accompanied by a decision structure illustrated in Figure 11.1. If a point has been accepted in the previous run, a new predictor is generated, and the angle between the two successive predictors is checked. If the angle is within a maximum allowable angle, the predictor is accepted. Subsequently the point is moved back onto the homotopy curve by means of a Newton correction step. If the iteration history suggests that convergence will not be attained within a desired number of iterations, the predicted point is rejected, the step size is halved, and a new estimate is generated. Halving the step size is done until the corrector gives an iteration history that is adequate for convergence within the desired number of iterations, or until a minimum allowable stepsize is attained. Once the estimate has been corrected a new predictor is generated and the angle between the new and the previous predictor is evaluated. If this value exceeds a maximum allowable value, the previously corrected point is rejected and a new estimate is generated based on the old predictor, while the step size is halved. If both the angle between successive predictors and the iteration history of the corrector are within the allowable limits, the step size is increased, according to Eq. (11.39) and Eq. (11.40). In addition, the step size is not allowed to become smaller than a minimum allowable stepsize and it is not allowed to become larger than a maximum allowable stepsize.

11.3.4 Problems and Pitfalls

Homotopy continuation methods sometimes are claimed to be methods that will guarantee a solution. This is not true. There are several situations, in which the continuation method will fail.

- The augmented Jacobian is not of full rank. In this case, no invertible Jacobian can be obtained, and the system cannot be solved. The points at which the Jacobian is not of full rank may be termed ‘singular’. At these points, the homotopy path may, for instance, split up into two (or more) branches that can lead to multiple solutions to the problem. This is normally only a problem if either the predictor or the corrector is located very closely to the singular point.
- Jumping over and skipping parts of the homotopy curve can take place if the homotopy curve is highly curved, and the parameters for the step size algorithm (minimum allowable step size, maximum allowable stepsize, maximum allowable angle between successive predictors) allow this. What can happen here is illustrated in Figure 11.2. An example homotopy curve is given in the top left of the figure, and a good curve tracking routine should follow the curve as depicted in the top right of the figure. If a bad step size algorithm or a badly tuned method are used, any of the four scenarios depicted in the figure may then

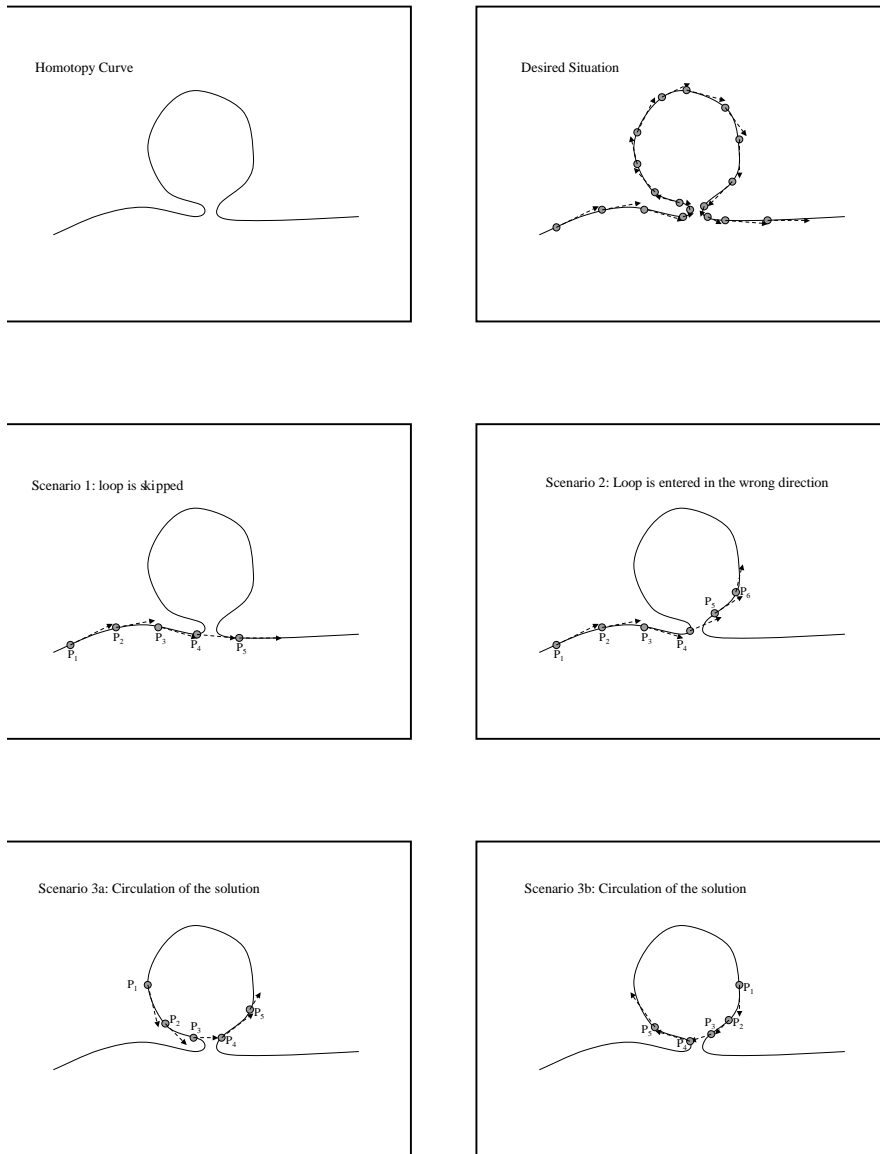


Figure 11.2: Problems and pitfalls.

occur. The loop may be skipped entirely, or entered the in the wrong direction. In addition, failures may occur in exiting the loop, also resulting in no solution.

References

Broyden, C.G., "A Class of Methods for Solving Nonlinear Simultaneous Equations", *Math. Comp.*, Vol. **19** (1965) p. 577.

Garcia, C.B. and W.I. Zangwill; *Pathways to Solutions, Fixed Points, and Equilibria*, Prentice Hall, Inc.; Englewood, NJ (1981).

Ortega, J.M., W.C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York (1970).

Pozrikidis, C. *Numerical Computation in Science and Engineering*, Oxford University Press, Oxford (1998).

Schubert, L.K., "Modification of a Quasi-Newton Method for Nonlinear Equations with a Sparse Jacobian", *Math. Comp.*, Vol. **24** (1970) p. 27.

Woodman, M.R., "Simulation of the Distillation of Three Phase Mixtures", Ph.D. Thesis, Cambridge University (1989).

Chapter 12

Flash Calculations

A flash is a one stage operation where a (multiple phase) feed is ‘flashed’ to a certain temperature and/or pressure and the resulting phases are separated. The flash in *ChemSep* deals only with two different phases leaving, a vapor and a liquid. Liquid-Liquid or multiphase Vapor-Liquid-Liquid flashes are currently not yet supported in *ChemSep*. For more information see the references given at the end of this chapter.

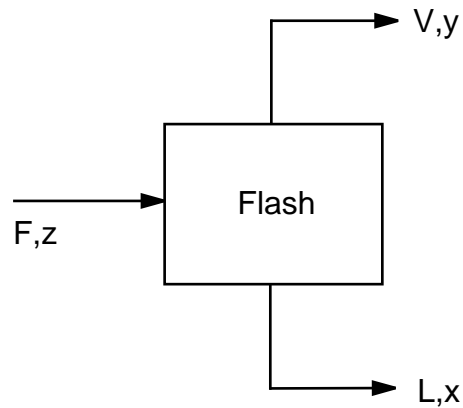


Figure 12.1: Schematic diagram of an equilibrium flash.

12.1 Equations

A schematic diagram of an equilibrium flash is drawn in Figure 12.1. The vapor and liquid streams leaving the flash are assumed to be in equilibrium with each other. The equations that model equilibrium flashes are summarized below:

- The **Total Material Balance**:

$$V + L - F = 0 \quad (12.1)$$

- The **Component Material Balances**:

$$Vy_i + Lx_i - Fz_i = 0 \quad (12.2)$$

- The **Equilibrium relations**:

$$K_i x_i - y_i = 0 \quad (12.3)$$

- The **Summation equation**:

$$\sum_{i=1}^c (y_i - x_i) = 0 \quad (12.4)$$

- The **Heat (or enthalpy) balance**:

$$VH^V + LH^L - FH^F + Q = 0 \quad (12.5)$$

where F is the molar feed rate with component mole fractions z_i . V and L are the leaving vapor and liquid flows with mole fractions y_i and x_i , respectively. Equilibrium ratios K_i and enthalpies H are computed from property models as discussed in other chapters. Q is defined as the heat added to the feed before the flash.

If we count the equations listed, we will find that there are $2c + 3$ equations, where c is the number of components. The variables appearing in these equations are:

- c vapor mole fractions, y_i
- c liquid mole fractions, x_i
- vapor flow rate, V
- liquid flow, L

- temperature, T
- pressure, p
- heat duty, Q

Since we have $2c + 3$ equations, two of the $2c + 5$ variables above must be specified. *ChemSep* allows the following nine flash specifications:

PT: pressure and temperature

PV: pressure and vapor flow

PL: pressure and liquid flow

PQ: pressure and heat duty

TV: temperature and vapor flow

TL: temperature and liquid flow

TQ: temperature and heat duty

VQ: vapor flow and heat duty

LQ: liquid flow and heat duty

12.2 Solution of the Flash Equations

ChemSep uses Newton's method for solving flash problems as well as simpler bubble and dew point calculations. The vector of variables used in the PQ-FLASH is:

$$\mathbf{X}^T = (V, y_1, y_2 \dots y_c, T, x_1, x_2 \dots x_c, L) \quad (12.6)$$

the vector of functions, (\mathbf{F}), is:

$$\mathbf{F}^T = (TMB, CMB_1, CMB_2 \dots CMB_c, H, EQM_1, EQM_2 \dots E_c, SUM) \quad (12.7)$$

The structure of the Jacobian matrix \mathbf{J} is shown below:

	V	y	T	x	L
TMB	1				1
CMB		\		\	
H	x	-	x	-	x
EQM		#		#	
SUM		-		-	

The symbols used in this diagram are as follows:

- x single matrix element
- 1 single element with a value of unity
- | vertical column of c elements
- \ diagonal with c elements
- # square submatrix of order c
- row submatrix with c elements

12.3 The Rachford-Rice Equation

Another effective method for solving the flash equations was developed by Rachford and Rice (1952). Their method combines all the material balances, equilibrium relations and summation equations into one equation. To obtain the RR equation the component material balance is used to express x_i :

$$x_i = \frac{Fz_i}{L + VK_i} \quad (12.8)$$

On eliminating L using the total molar balance $L = F - V$ we find

$$x_i = \frac{z_i}{1 + (K_i - 1)\frac{V}{F}} \quad (12.9)$$

A similar equation can be derived for the vapor mole fractions:

$$y_i = \frac{K_i z_i}{1 + (K_i - 1)\frac{V}{F}} \quad (12.10)$$

The RR equation is based on subtracting the mole fraction summation equations from each other:

$$\sum x_i - \sum y_i = 0 \quad (12.11)$$

which becomes:

$$\sum_{i=1}^c \frac{(K_i - 1)z_i}{1 + (K_i - 1)\frac{V}{F}} = 0 \quad (12.12)$$

Of course, this equation can only be solved if the K-values differ from unity. Given pressure and temperature, the K-values can be calculated and then the vapor to feed flow ratio is determined from the root of the RR equation. The RR function is monotonic in the ratio of vapor flow over the feed flow, V/F . This is convenient, for this ratio is also bounded between 0 to 1.

This method can be selected by editing the sep file and typing -1 before *method* under the Solve Options section heading (Newton's method is 1).

References

Henley, E.J., J.D. Seader, *Equilibrium-Stage Separation Operations in Chemical Engineering*, Wiley (1981).

King, C.J., *Separation Processes*, Second Edition, McGraw Hill (1980).

Rachford, H.H. Jr., J.D. Rice, *J. Petrol. Technol.*, **4**, No. 10 (1952) p. 19.

Chapter 13

Equilibrium Stage Simulation

In this chapter we give the reader a description of the equilibrium stage model for multistage, multicomponent separation processes. We also provide an overview of the methods which used to solve the system of equations.

This chapter is adapted from material written by David Vickery and Ross Taylor in 1987 at Clarkson University. Ross used this material during his sabbatical year at the University of Delft (The Netherlands) to teach an advanced separation processes course. Since then corrections have been made to bring the review more up to date, although developments continue to take place and no review of this field can ever be complete.

13.1 Introduction

Multicomponent separation processes like distillation, absorption and extraction have been modelled using the equilibrium stage concept for almost a century. The equations that model equilibrium stages are called the MESH equations. The M equations are the Material balance equations, the E equations are the Equilibrium relations, the S equations are the Summation equations and the H equations are the enthalpy balance equations.

The MESH equations for the interior stages of a column together with equations for the reboiler and condenser (if they are needed) are solved together with any specification equations to yield, for each stage, the vapor mole fractions; the liquid mole

fractions; the stage temperature and the vapor and liquid flow rates.

Since the late 1950's, hardly a year has gone by without the publication of at least one (and usually more than one) new algorithm for solving the equilibrium stage model equations. One of the incentives for the continued activity has always been (and remains) a desire to solve problems with which existing methods have trouble. The evolution of algorithms for solving the MESH equations has been influenced by, among other things: the availability (or lack) of sufficient computer storage and power, the development of mathematical techniques that can be exploited, the complexity of physical property (K-value and enthalpy) correlations and the form of the model equations being solved. Developments to about 1980 have been described in a number of textbooks (see, for example, Holland, 1963, 1975, 1981; King, 1980; Henley and Seader, 1981; Seader and Henley, 1998) and a number of reviews (see, for example, Wang and Wang, 1981; Haas, 1992). Seader (1985a) has written an interesting history of equilibrium stage simulation.

We may identify several classes of methods of solving the MESH equations:

1. Short cut methods in which a great many simplifying assumptions are made in order to obtain approximate solutions of the MESH equations for certain special cases (for example, total or minimum reflux operation). We shall not discuss such methods in this review.
2. Tearing Methods in which the equations are divided into groups and solved separately.
3. Simultaneous Correction Methods in which all of the equations are solved simultaneously using Newton's method (or a variant thereof).
4. Relaxation Methods in which the MESH equations are cast in unsteady state form and integrated numerically until the steady state solution has been found.
5. Continuation Methods which are intended for solving difficult problems and are the subject of much current research interest. We can expect to see further developments in this area in the next few years.
6. Collocation Methods that have been found useful for solving systems of partial differential equations have been adapted to solve both steady state and dynamic equilibrium stage model problems. These methods have some potential for certain types of simulation problems but have not yet become part of the mainstream of equilibrium stage simulation and are not discussed further. The reader is referred to papers by Stewart et al. (1985), Swartz and Stewart (1986), Cho and Joseph (1983, 1984), and by Seferlis and Hrymak (1994a,b).

Most algorithms for solving the MESH equations in use today fall into categories 2 and 3; tearing methods and simultaneous correction methods. Following sections on each of these two classes of method we discuss continuation methods in Section 13.5 and relaxation methods in Section 13.6. We conclude with a short discussion of three phase distillation.

13.2 The Equilibrium Stage Model

The equilibrium stage model was first used by Sorel (1893) to describe the rectification of alcohol. Since that time it has been used to model all manner of separation processes: distillation (including rectification, stripping, simple (single feed, two product columns), complex (multiple feed, multiple product columns), extractive, azeotropic and petroleum refinery distillation), absorption, stripping, liquid-liquid and supercritical extraction.

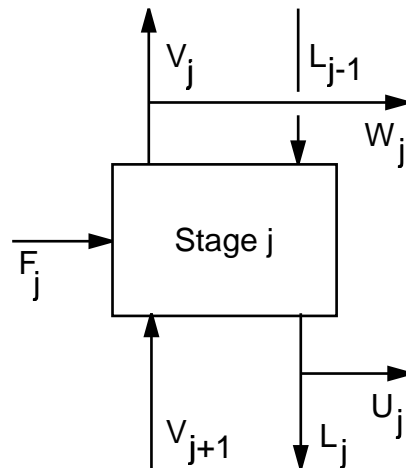


Figure 13.1: Schematic diagram of an equilibrium stage.

In order to better understand the methods that have been proposed for solving the equations modelling equilibrium stage operations, some familiarity with the basic model is necessary. A schematic diagram of an equilibrium stage is shown in Figure 13.1. Vapor from the stage below and liquid from a stage above are brought into contact on the stage together with any fresh or recycle feeds. The vapor and liquid streams *leaving* the stage are *assumed* to be in *equilibrium* with each other. A complete separation process is modeled as a sequence of s of these *Equilibrium Stages* (Figure 13.2).

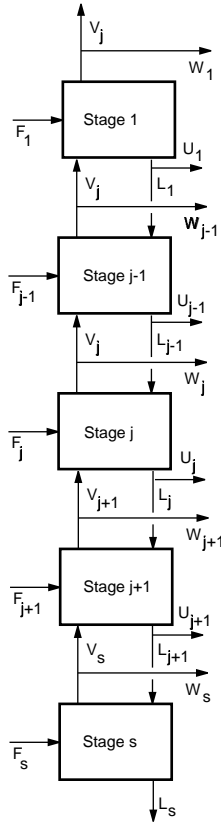


Figure 13.2: A sequence of stages modeling a separation column.

13.2.1 The MESH Equations (The 2c+3 Formulation)

The equations that model equilibrium stages are termed the MESH equations, MESH being an acronym referring to the different types of equations that form the mathematical model. The M equations are the **M**aterial balance equations, of which there are two types: the *Total Material Balance*:

$$\begin{aligned}
 M_j^T &\equiv (1 + r_j^V) V_j + (1 + r_j^L) L_j \\
 &\quad - V_{j+1} - L_{j-1} - F_j = 0
 \end{aligned}
 \tag{13.1}$$

and the *Component Material Balances*:

$$M_{ij} \equiv (1 + r_j^V) V_j y_{ij} + (1 + r_j^L) L_j x_{ij}$$

$$-V_{j+1}y_{i,j+1} - L_{j-1}x_{i,j-1} - F_j z_{ij} = 0 \quad (13.2)$$

r is the ratio of sidestream flow to interstage flow:

$$r_j^V = W_j/V_j \quad r_j^L = U_j/L_j \quad (13.3)$$

The E equations are the *Equilibrium relations*:

$$E_{ij} \equiv K_{ij}x_{ij} - y_{ij} = 0 \quad (13.4)$$

where the K_{ij} are the K-values for species i on stage j . The estimation of the K-values from fugacity and activity coefficients is discussed at length in chapters ?? and 10.

The S equations are the *Summation equations*:

$$S_j^V \equiv \sum_{i=1}^c y_{ij} - 1 = 0 \quad (13.5)$$

$$S_j^L \equiv \sum_{i=1}^c x_{ij} - 1 = 0 \quad (13.6)$$

and the H equations are the *Heat balance equations*:

$$\begin{aligned} H_{ij} \equiv & (1 + r_j^V) V_j H_j^V + (1 + r_j^L) L_j H_j^L + Q_j \\ & - V_{j+1} H_{j+1}^V - L_{j-1} H_{j-1}^L - F_j H_j^F = 0 \end{aligned} \quad (13.7)$$

where the superscripted "H's" are the enthalpies of the appropriate phase. If we count the equations listed, we will find that there are $2c + 4$ equations per stage. However, only $2c + 3$ of these equations are independent. These independent equations are generally taken to be the c component mass balance equations, the c equilibrium relations, the enthalpy balance and two more equations. These two equations can be the two summation equations or the total mass balance and one of the summation equations (or an equivalent form). The $2c + 3$ unknown variables determined by the equations are the c vapor mole fractions, y_{ij} ; the c liquid mole fractions, x_{ij} ; the stage temperature, T_j and the vapor and liquid flow, V_j and L_j . Thus, for a column of s stages, we must solve $s(2c + 3)$ equations. Table 13.1 shows how we may easily end up having to solve hundreds or even thousands of equations.

The first entry in this table corresponds to a simple binary problem that could easily be solved graphically. The second and third are fairly typical of the size of problem encountered in azeotropic and extractive distillation processes. The last two entries are typical of problems encountered in simulating hydrocarbon and petroleum mixture separation operations.

Table 13.1: Number of equations for equilibrium stage model

Number of components c	Number of stages s	Number of equations $s(2c + 3)$
2	10	70
3	20	180
5	50	650
10	30	690
40	100	8300

13.2.2 The $2c + 1$ Formulation

An alternative form of the MESH equations is used in many algorithms. In this variation, we make use of the *component flow* defined by:

$$v_{ij} = V_j y_{ij} \quad \text{or} \quad l_{ij} = L_j x_{ij} \quad (13.8)$$

to decrease the number of equations and variables for each stage by two. In terms of the component flow, the MESH equations can be written as:

$$M_{ij} \equiv (1 + r_j^V) v_{ij} + (1 + r_j^L) l_{ij} - v_{i,j+1} - l_{i,j-1} - F_{ij} = 0 \quad (13.9)$$

$$E_{ij} \equiv K_{ij} \frac{l_{ij}}{\sum_{k=1}^c l_{kj}} - \frac{v_{ij}}{\sum_{k=1}^c v_{kj}} = 0 \quad (13.10)$$

$$H_{ij} \equiv (1 + r_j^V) \sum_{k=1}^c v_{kj} H_j^V + (1 + r_j^L) \sum_{k=1}^c v_{kj} H_j^L + Q_j - \sum_{k=1}^c v_{k,j+1} H_{j+1}^V - \sum_{k=1}^c l_{k,j-1} H_{j-1}^L - F_j H_j^F = 0 \quad (13.11)$$

Since the total vapor and liquid flow rates are, by definition, the sum of the component flow rates of the respective phases, the summation equations and total mass balance equation are satisfied automatically. Thus, the number of equations and variables per stage has been reduced from $2c + 3$ to $2c + 1$.

13.2.3 The $c + 3$ Formulation

The number of unknown variables per stage can be reduced to only $c + 3$ if we use the equilibrium relations (13.4) to eliminate the vapor phase mole fractions from the

component mass balances (13.2):

$$ME_{ij} \equiv + [(1 + r_j^V) V_j K_{ij} + (1 + r_j^L) L_j] x_{ij} + F_j z_{ij} - V_{j+1} K_{i,j+1} x_{i,j+1} - L_{j-1} x_{i,j-1} = 0 \quad (13.12)$$

and from the summation equation (13.6)

$$S_j^V \equiv \sum_{i=1}^c K_{ij} x_{ij} - 1 = 0 \quad (13.13)$$

which is familiar to us from bubble point calculations. In this formulation of the MESH equations the vapor phase mole fractions are no longer independent variables but are defined by equation (13.4). This formulation of the MESH equations has been used in quite a number of algorithms. It is less useful if vapor phase nonideality is important.

13.2.4 Special Stages

The MESH equations can be applied as written to any of the interior stages of a column. In addition to these stages, the reboiler and condenser (if they are included) for the column must be considered (see Figure 13.2). These stages differ from the other stages in the column in that they are a heat source or sink for the column. The MESH equations of the preceding subsections may be used to model these stages exactly as you would any other stage in the column. For a total condenser at the top of a distillation column, for example, the distillate is U_1 and the reflux ratio is $1/r_1^L$. For a partial condenser, the vapor product is V_1 and the "reflux ratio" is L_1/V_1 . Finally, for a partial reboiler at the base of a column, the bottoms flow rate is L_N . These special stages add degrees of freedom to the separation problem and therefore additional specification equations are required. Common specifications are:

1. the flow rate of the distillate/bottoms product stream,
2. the mole fraction of a given component in either the distillate or bottoms product stream,
3. component flow rate in either the distillate or bottoms product stream,
4. a reflux/reboil ratio or rate
5. the temperature of the condenser or reboiler
6. a heat duty to the condenser or reboiler.

In the case of a total condenser, the vapor phase compositions used in the calculation of the equilibrium relations and the summation equations are those that would be in equilibrium with the liquid stream that actually exists. That is for the total condenser, the vapor composition used in the equilibrium relations is the vapor composition determined during a bubble point calculation based on the actual pressure and liquid compositions found in the condenser. At the same time, these compositions are not used in the component mass balances since there is no vapor stream from a total condenser.

13.2.5 Nonequilibrium Stages

In actual operation the trays of a distillation column rarely, if ever, operate at equilibrium despite attempts to approach this condition by proper design and choice of operating conditions. The degree of separation is, in fact, determined as much by mass and energy transfer between the phases being contacted on a tray or within sections of a packed column as it is by thermodynamic equilibrium considerations. The usual way of dealing with departures from equilibrium in multistage towers is through the use of stage and/or overall efficiencies.

There are many different definitions of stage efficiency: Murphree (1925), Hausen (1953), vaporization (Holland, 1975) and generalized Hausen (Standart, 1965). There is by no means a consensus on which is best. Arguments for and against various possibilities are presented by, among others, Standart (1965, 1971), Holland and McMahon (1970) and by Medina et al. (1978, 1979). Possibly the most soundly based definition, the generalised Hausen efficiency of Standart (1965) are the most difficult to use (see, however, Fletcher, 1987), the least soundly based, the Murphree efficiency is the one most widely used in separation process calculations because it is easily combined with the equilibrium relations:

$$E_{ij} \equiv E_{ij}^{MV} K_{ij} x_{ij} - y_{ij} - (1 - E_{ij}^{MV}) y_{i,j+1} = 0 \quad (13.14)$$

where E_{ij}^{MV} is the Murphree vapor efficiency for component i on stage j . In the $2c+1$ formulation of the MESH equations, the equilibrium relations read as follows

$$E_{ij} \equiv \frac{E_{ij}^{MV} K_{ij} l_{ij}}{\sum_{k=1}^c l_{kj}} - \frac{v_{ij}}{\sum_{k=1}^c v_{kj}} + \frac{(1 - E_{ij}^M) v_{i,j+1}}{\sum_{k=1}^c v_{k,j+1}} = 0 \quad (13.15)$$

Efficiencies are not normally incorporated in the $c + 3$ formulation for reasons that will be discussed later.

Whichever definition of efficiency is adopted, it must either be specified in advance or calculated from an equation derived by dividing by some reference separation the

actual separation obtained from a model of the transport processes taking place on a tray. Common practice is to use a single value for all of these efficiencies, one calculated (or specified) in terms of the plate efficiency of the key components. In a c component system there are $c - 1$ independent component efficiencies and there are sound theoretical reasons as well as experimental evidence for not assuming the component efficiencies to be alike (see Taylor and Krishna, 1993 for a review of the literature on this topic).

13.2.6 Numerical Solution of the MESH Equations. Preliminary Considerations

The MESH equations form a set of nonlinear, algebraic equations (algebraic in the sense that no derivatives or integrals are involved) and must, therefore, be solved by some iterative process. Their are two steps in the development of an algorithm for solving systems of nonlinear equations:

- The selection of particular numerical methods
- The selection of the order in which the equations are to be solved

Almost every one of the many numerical methods that have been devised for solving systems of nonlinear equations has been used to solve the MESH equations. However, most of the more widely used methods are based on just two methods: repeated substitution and Newton's method (or one of its relatives). These methods are described in detail in Chapter 11.

13.3 Tearing Methods

Equation Tearing involves breaking a system of nonlinear equations into small groups and solving each group of equations in turn. When solving any subset of the complete set of equations, only a corresponding number of variables, the "tear" variables, can be determined. In order to start the calculations, therefore, it is necessary to assume values for the remaining variables. The "torn" set of equations is then solved for the "tear" variables assuming that the values assigned to all other variables are correct. Successive groups of equations and variables are *torn* or *decoupled* from the full set of equations and variables until all the variables have been updated. At this point, the process starts over and is repeated until all the equations are satisfied simultaneously.

A great many tearing methods for solving the MESH equations have been proposed. According to Friday and Smith (1964), tearing methods may be analysed in the following terms:

1. The order in which the equations are grouped.
2. The order in which each group of equations is solved.
3. The selection of which variables are to be computed from which equation.

Subsequent issues are:

4. The method(s) used to solve the M and E equations.
5. The method of calculating the new stage temperatures.
6. The method of calculating the new flow rates, V_j and L_j

The first methods for solving the MESH equations involve keeping the equations for a given stage together; the equations for one stage are solved then the equations for a second stage are solved and so on until all the equations for the column have been solved. These are called *Stage to Stage* methods and are discussed in Section 13.3.1 below. Most modern methods group all the equations of a given type together; for example, the M equations or the H equations, for all the stages at once. These methods are discussed in Section 13.3.2.

13.3.1 Stage-to-Stage Methods

The papers by Lewis and Matheson (1932) and Thiele and Geddes (1933) represent the first attempts at solving the MESH equations for multicomponent systems *numerically* (graphical methods for binary systems had already been developed by Ponchon, Savarit and by McCabe and Thiele). At that time the computer had yet to be invented and since, modelling a column could require hundreds or even thousands of equations, it was necessary to break the MESH equations in smaller subsets if hand calculations were to be feasible.

In 1932, Lewis and Matheson proposed an algorithm in which the MESH equations were solved in a "tray-by-tray" approach. The calculations were for a *design* problem, a problem in which the recoveries of key components are specified and the number of trays above and below the feed must be determined. In their approach, the MESH equations of one stage were solved to determine either the stage temperature, the flow rate and composition of the entering vapor stream and the flow rate and composition

of the exiting liquid stream (in the top or enriching section of the column) or the stage temperature, the flow rate and composition of the entering liquid stream and the flow rate and composition of the exiting vapor stream (in the bottom or stripping section of the column).

Based on the specified recoveries, estimates of the product stream compositions and flow rates were made and these estimates were used to start the calculations which then proceeded along two paths. With the conditions of the distillate known (assumed), it is possible to use the MESH equations describing the condenser to determine the composition and flow rate of the liquid reflux to the column and the vapor entering the condenser (and leaving the top stage of the column itself). Since the liquid leaving this stage is in equilibrium with this vapor stream, it follows that a simple (or not so simple) dew point calculation will give the composition of the liquid stream leaving this stage. If equimolar overflow is assumed then the flow rates throughout the column are already known and the composition of the vapor stream entering the stage can be determined quite straightforwardly from the component material balances for the stage. If equimolar overflow does not apply then it is necessary to compute the flow rate of the leaving liquid stream and the flow rate and composition of the vapor stream entering from the stage below by solving the component material balances together with the total material balance and energy balance. This is a much more involved calculation.

Now we know the composition of the vapor leaving the next stage lower down the column and we can repeat the entire procedure for this next stage and so we continue until the feed stage is reached. At this point, this set of calculations is terminated and a corresponding set of calculations moving from the bottom of the column up to the feed stage is carried out. The extent to which these progressions of calculations do not agree hopefully provides a basis for adjusting the estimates of the end conditions. Then the top-down, bottom-up calculations can be repeated until the midcolumn matching achieves the desired accuracy. When this method was first proposed there was no systematic method for making the adjustments in the end conditions.

The following year, Thiele and Geddes (1933) proposed an approach which was not only a tray-by-tray method but also one in which each type of equation (M-E-S-H) was solved in turn. In this method, the number of stages above and below the feed along with the reflux ratio and the distillate flow rate were given as column specifications (a form of the operating problem). Values for the stage temperatures and vapor flow rates were assumed and the M and E equations were solved tray-to-tray for the compositions throughout the column (K-values were assumed to be functions of temperature only). Then the stage temperatures and vapor flow rates were re-estimated using the S+E and H equations respectively.

Holland and coworkers (see Holland, 1963) developed the "theta-method" in order to improve the convergence properties of the Thiele - Geddes algorithm by forcing the component material balances for the column as a whole to close on every iteration. This allowed faster, surer convergence than the original procedure. The theta method could also be applied to the Lewis - Matheson method described above for adjusting the estimated distillate composition. In both cases, the theta method significantly improves the reliability of these methods.

Despite their essential simplicity and appeal from the pedagogical viewpoint, it must be said that stage to stage calculation procedures are not used now as often as they used to be. It was the development of the digital computer that led, in the first place, to the "routine" use of these methods and, subsequently, to the discovery of problems which these, the earliest numerical methods for solving the MESH equations, could not solve. One drawback to stage to stage methods is that they are sometimes susceptible to the propagation of truncation errors caused by the use of finite length arithmetic (eight digits on most computers unless double precision is used). This is why stage-to-stage calculations with trace components are (very) difficult. Holland (1963) gives a nice illustration of this problem for those interested in finding out more.

13.3.2 Matrix Methods

Matrix techniques were first used in separation process calculations by Amundson and Pontinen (1958) who demonstrated that the combined M+E equations (the $c + 3$ formulation) could be conveniently written in matrix form. We define a column matrix of combined material balance and equilibrium relations \mathbf{F} as

$$\mathbf{F}^T = (ME_{i,1}^c, \dots, ME_{i,j-1}^c, ME_{i,j}^c, ME_{i,j+1}^c, \dots, ME_{i,s}^c)^T \quad (13.16)$$

where ME_{ij}^c is Eq. (13.12). We also define a column matrix of liquid phase mole fractions (\mathbf{X}) by

$$\mathbf{X}^T = (x_{i,1}, \dots, x_{i,j-1}, x_{i,j}, x_{i,j+1}, \dots, x_{i,s})^T \quad (13.17)$$

Now, we may write the vector function \mathbf{F} in matrix form as follows:

$$\mathbf{ABC} \quad \mathbf{x} = \mathbf{R} \quad (13.18)$$

This system of equations has the structure shown below

$$\begin{bmatrix} B_1 & C_1 & & & & \\ A_2 & B_2 & C_2 & & & \\ \dots & \dots & \dots & \dots & & \\ & A_j & B_j & C_j & & \\ & \dots & \dots & \dots & \dots & \\ & & A_{n-1} & B_{n-1} & C_{n-1} & \\ & & & A_n & B_n & \end{bmatrix} \begin{pmatrix} X_1 \\ X_2 \\ \dots \\ X_j \\ \dots \\ X_{n-1} \\ X_n \end{pmatrix} = \begin{pmatrix} R_1 \\ R_2 \\ \dots \\ R_j \\ \dots \\ R_{n-1} \\ R_n \end{pmatrix} \quad (13.19)$$

where the coefficient matrix **ABC** has *three* adjacent diagonals with coefficients:

$$A_j = L_{j-1} \quad (13.20)$$

$$B_j = -(V_j K_{ij} + L_j) \quad (13.21)$$

$$C_j = V_j K_{ij} \quad (13.22)$$

The right hand side matrix **R** has elements

$$R_j = -F_j z_{ij} \quad (13.23)$$

Note that there is one of these tridiagonal matrix equations for each component in the mixture. A specialized form of Gaussian elimination known as the Thomas Algorithm (Lapidus, 1962) can be used to solve the tridiagonal system (13.18) very efficiently. This form of elimination procedure is used to minimize the amount of storage and calculations for solving the linear system of equations and is summarized in Table 13.2.

The first step of any method in this category is to estimate the flow rates and K-values. The three other independent equations for each stage; i.e. the total mass balances, the enthalpy balances and the summation equations; are subsequently used to determine the three remaining independent variables for each stage; the vapor flow rates, the liquid flow rates and the stage temperatures. When all these variables have been re-estimated, the coefficients in the system of combined M and E equations are recalculated and the process is repeated until all the model equations are simultaneously satisfied.

Three quite different approaches to the problem of computing the flow rates and temperatures have evolved:

1. In the *Bubble Point (BP) method*, the S+E equations are used to determine the stage temperatures from a bubble point calculation. The vapor and liquid flow rates are computed from the energy balances and total material balances.
2. In the *Sum Rates (SR) method* the vapor and liquid flow rates are computed directly from the summation equations and the stage temperatures determined from the energy balances.

Table 13.2: The Thomas algorithm

- Elimination procedure:

For the first equation ($j = 1$):

$$C_1 = \frac{C_1}{B_1}$$

$$R_1 = \frac{R_1}{B_1}$$

For $j = 2$ to $N - 1$:

$$D_j = B_j - A_j C_{j-1}$$

$$C_j = \frac{C_j}{D_j}$$

$$R_j = \frac{R_j - A_j R_{j-1}}{D_j}$$

For the last equation ($j = N$):

$$D_N = B_N - A_N C_{N-1}$$

$$R_N = \frac{R_N - A_N R_{N-1}}{D_N}$$

- Backward substitution procedure:

For the last variable:

$$X_N = R_N$$

For $j = N - 1$ down to 1:

$$X_j = R_j - C_j X_{j+1}$$

3. In the Newton-Raphson methods the bubble point equations and energy balances are solved simultaneously for the stage temperatures and vapor flow rates; the liquid flow rates follow from the total material balances.

13.3.3 The BP Method

The BP method was introduced by Amundson and Pontinen (1958) who used matrix inversion to solve equations (13.12). A significant improvement in the BP method equations was introduced by Wang and Henke (1966) who introduced the Thomas Algorithm (Lapidus, 1962) to solve the tridiagonal system (4.1). Holland and more coworkers (1974,1975) have combined some of these ideas with the "theta-method" of convergence acceleration (Holland, 1963) to develop a "modified theta-method". The "theta" and "modified theta" methods themselves have been further refined by Pierucci et al. (1983) and have even been cast in a form where the "theta" correction factor is replaced with a temperature dependent parameter, the T-method (Pierucci and Ranzi, 1982; Pierucci et al., 1983). The BP method of solving the MESH equations is summarized in Table 13.3. Sridhar (1990) has developed a modified BP method using the same principles as Sridhar and Lucia (1990) for the SR method (see below) and obtained similar improvements in reliability when compared to traditional BP methods.

13.3.4 The SR Method

The SR algorithm summarized in Table 13.4 is due to Sujata (1961) and Friday (1963) actually, Sujata (1961) used the $2c+1$ version of the MESH equations in his work and the tridiagonal system of combined M+E equations was written in terms of the component flow rates in the liquid phase. Burningham and Otto (1967) incorporated the Thomas algorithm in their implementation of the SR method. Sridhar and Lucia (1990) developed a modified SR method based on insights provided by rigorous analysis that provides an analytical expression for the Newton acceleration of the inner loop as well as partial derivatives of the total vapor flow with respect to temperature needed in the outer loop. The result is a method that is capable of solving problems involving narrow, intermediate and wide-boiling mixtures alike.

13.3.5 Newton Raphson Methods

Tierney and coworkers (Tierney and Yanosik, 1969; Tierney and Bruno, 1967; Bruno et al., 1972) and Billingsley and Boynton (1971) avoided the problem of assigning either the vapor flow rates or temperatures to either the summation equations or enthalpy balances by solving both sets of equations for both sets of variables simultaneously using a multidimensional form of Newton's method (order $2N$ to be precise). The liquid flow rates are eliminated from these equations using the total material balances. Any composition dependence of the K-values and enthalpies is ignored in the application

of Newton's method of which more in the next section.

The "multi-theta method" of Holland (1975) is similar to the Tierney and Yanosik method except that the stage temperatures and flow ratios (V_j / L_j) are taken to be the independent variables. Tomich (1970) has also worked with a 2N Newton Raphson method and used Broyden's quasi Newton method to update the Jacobian after just one iteration with the full Newton method.

Other applications of Newton's method to solving distillation problems (within the context of a tearing algorithm) are due to Newman (1963) and Boynton (1970).

13.3.6 Inside-Out Methods

One interesting attempt to alleviate the difficulties mentioned resulted in the so called "inside-out" (IO) algorithms of Boston and Sullivan (1974), Boston and Britt (1978), and Boston (1980). In these methods, complicated equilibrium and enthalpy expressions were replaced by simple models and the iteration variables T , V , x and y were replaced by variables within the models which are relatively free of interactions with each other. However, the resulting algorithms are quite problem specific requiring new software for each type of specification, flash problem, etc. This is not to say that other decoupling methods are not at all application specific. IO methods for most of the common equilibrium stage separation processes have been developed; indeed, IO methods have replaced many of the older algorithms in commercial simulation programs, the modified version of Russell (1983) being the basis for some of them. Readers are referred to Seader and Henley (1998) for complete details of the somewhat lengthy algorithm.

13.3.7 Nonequilibrium Stages

Murphree efficiencies are not normally included in the matrix methods described in the preceding section because if Equation (13.4) is used to eliminate the y_{ij} from the component material balances (13.2) the upper triangular portion of the coefficient matrix is filled out (Huber, 1977). We could use Equation (13.4) to eliminate the x_{ij} from the component mass balances and solve the resulting tridiagonal system for the vapor phase compositions instead but this is not recommended (King, 1980, p 472). Rather than combine Equations (13.4) and (13.2) into a single set of equations, we define a column matrix of discrepancy functions

$$\mathbf{F}^T = (M_{i,1}^c, E_{i,1}, \dots, M_{i,j-1}^c, E_{i,j-1}, M_{i,j}^c, E_{i,j},$$

Table 13.3: Bubble Point Method for Solving the MESH Equations

Begin Outer Loop

1. Initialize the vapor and liquid flow rates using the total mass balance and the enthalpy balance with an assumption of constant molar flows (flow rates change from one stage to the next only upon the introduction of a feed).

Begin Inner Loop

2. Choose an initial temperature profile for the column.
3. Evaluate the K-values for each stage (if this is the first iteration, either use an ideal K-value [temperature dependence only] or assume that all mole fractions are the same).
4. Set up the linearized systems of equations (one system for each component) as defined by equations (13.12) and solve for the liquid phase compositions.
5. For each stage j , normalize the liquid compositions.
6. Using the normalized liquid compositions, solve the summation equation for each stage [in the form $(x_i - y_i)$] for the stage temperatures. If there is no change since the last iteration of the inner loop proceed to Step 7, otherwise return to Step 3.

End Inner Loop

7. Use the enthalpy and total mass balances to re-estimate the vapor and liquid flow rates. If there is no change since the last outer loop iteration, end; otherwise return to Step 4 (the K-values will not have changed since the last inner loop iteration).

End Outer Loop

$$M_{i,j+1}^c, E_{i,j+1}, \dots, M_{i,s}^c, E_{i,s})^T \quad (13.24)$$

where M_{ij}^c is the component material balance Equation (13.2) and E_{ij} represents equation (13.4). Each pair of equations depends on only four mole fractions: $x_{i,j-1}$, y_{ij} , x_{ij} and $y_{i,j+1}$. Thus, if we define a column matrix of mole fractions \mathbf{X} by

$$\mathbf{X}^T = (y_{i,1}, x_{i,1}, \dots, y_{i,j-1}, x_{i,j-1}, y_{i,j}, x_{i,j},$$

Table 13.4: Sum Rates Method for Solving the MESH Equations

1. Estimate the vapor flow rates and calculate the liquid flow rates using the total mass balances. Assume constant molar flows from stage to stage (flow rates change from one stage to the next only upon the introduction of a feed).
2. Choose an initial temperature profile for the column.
3. Evaluate the K-values for each stage (if this is the first iteration, either use an ideal K-value [temperature dependence only] or assume that all mole fractions are the same as the combined feed composition).
4. Set up the tridiagonal systems of equations (one system for each component) as defined by equations (13.12) and solve for the liquid phase compositions.
5. For each stage, sum the liquid compositions. Compute new liquid flow rates by multiplying old value by the sums computed in step 5. Compute new vapor flow rates from the total mass balances.
6. Normalize the $x_{i,j}$. Compute vapor phase compositions from the equilibrium relations and then normalize the vapor mole fractions.
7. Use the enthalpy balances to re-estimate the stage temperatures. Compare new stage temperatures and flow rates with the previous values. If there is no change since the last iteration, end; otherwise return to Step 3.

$$y_{i,j+1}, x_{i,j+1}, \dots, y_{i,s}, x_{i,s})^T \quad (13.25)$$

we may write

$$\mathbf{ABCD} \mathbf{X} - \mathbf{R} = \mathbf{0} \quad (13.26)$$

With the equations and variables ordered in this way, the coefficient matrix **ABCD** has *four* adjacent diagonals with the additional diagonal in the upper triangular portion of the matrix:

$$\begin{bmatrix} B_1 & C_1 & D_1 & & & \\ A_2 & B_2 & C_2 & D_2 & & \\ \dots & \dots & \dots & \dots & & \\ & A_j & B_j & C_j & D_j & \\ \dots & \dots & \dots & \dots & \dots & \\ & & A_{n-1} & B_{n-1} & C_{n-1} & \\ & & & A_n & B_n & \end{bmatrix} \begin{pmatrix} X_1 \\ X_2 \\ \dots \\ X_j \\ \dots \\ X_{n-1} \\ X_n \end{pmatrix} = \begin{pmatrix} R_1 \\ R_2 \\ \dots \\ R_j \\ \dots \\ R_{n-1} \\ R_n \end{pmatrix} \quad (13.27)$$

where $n (=2s)$ is the order of the matrix. On the odd numbered rows of **ABCD** the elements are given by

$$\begin{aligned} A_{2j-1} &= -L_j & B_{2j-1} &= W_j + V_j \\ C_{2j-1} &= U_j + L_j & D_{2j-1} &= -V_{j+1} \end{aligned} \quad (13.28)$$

(these elements being derived from the component material balances) whereas on the even numbered rows we have .

$$\begin{aligned} A_{2j} &= -1 & B_{2j} &= E_{ij}^{MV} K_{ij} \\ C_{2j} &= 1 - E_{ij}^{MV} & D_{2j} &= 0 \end{aligned} \quad (13.29)$$

which belong to the equilibrium relations. For a total condenser set $E^{MV} = 0$ For a total reboiler, set $E^{MV} = 0$ and $B_{2j} = 1$ The right hand side matrix **R** has elements

$$R_{2j-1} = F_j z_{i,j} \quad R_{2j} = 0 \quad (13.30)$$

The index j corresponds to the stage number and stages are numbered down from the top. This linear system of equations can be solved for the x_{ij} and y_{ij} very easily using Gaussian elimination.

We have used this method of solving the component material balances and efficiency equations in place of the corresponding steps in the BP method for distillation simulations. While not as fast as the conventional tridiagonal matrix algorithm (the quad-diagonal matrix has twice the number of rows as the tridiagonal matrix in the Wang and Henke method), it permits the straightforward solution of problems in which the stage efficiency is not unity. We estimate that the quad-diagonal algorithm will be faster than Huber's (1977) method of dealing with stage efficiencies (with an upper Hessenberg matrix of order s) if the number of stages exceeds nine.

13.3.8 Summary

Equation tearing involves iteratively solving a subset of the complete set of model equations for a subset (the tear variables) of the complete set of variables. Within the loop that determines how the tear variables are to be re-estimated, all other equations are solved exactly. If the remaining set of equations requires iteration (or, possibly, further tearing) the result is a series of nested iteration loops.

There are essentially two ways of tearing the equilibrium stage model equations: by stage or by type. Stage to stage methods were the first to be developed but have largely been abandoned in favor of the by-type methods which make use of matrix

formulations of the equations. By their very nature, matrix methods are limited to solving operating problems; design problems can be tackled by changing the problem specification, solving the operating problem that results and seeing whether it meets the design requirements. This limitation of the matrix methods (when compared to the stage to stage methods) is offset by the following:

1. There is no limit on the number of components. Stage to stage methods work best for two component systems and least well for systems with a great many components for which it is hard to make good estimates of the distillate composition.
2. There is no limit on the number of feed streams. Feed may even be introduced directly into the reboiler or condenser should the need arise without changing the form of (4.1). There is no limit on the number of sidestreams that may be withdrawn from the column (multiple feeds and sidedraws cause problems for stage to stage methods).
3. Most important of all is the fact that the combined mass balance and equilibrium relations depend on the liquid phase composition on only *three adjacent stages*: $x_{i,j-1}$, x_{ij} and $x_{i,j+1}$. This means that the very efficient Thomas algorithm for computing the liquid phase composition from Equation (13.12) can be used.

There are three major sub-types of matrix method for solving equilibrium stage separation process problems: the BP, SR and Newton-Raphson methods. The BP formulation supposedly works "well" for narrow boiling systems and the SR formulation for wide boiling systems typical in absorption and stripping operations (see Friday and Smith (1964) for the first discussion of this point). Two recent papers by Sridhar and Lucia (1989, 1990a) have analyzed these classic tearing algorithms in some detail. Sridhar and Lucia (1990b) have presented a modified sum-rates method that can be used to solve problems involving narrow boiling mixtures as well as wide boiling systems.

Inside-out methods are widely used in commercial simulation programs.

13.4 Simultaneous Correction Methods

One problem with tearing methods is the number of times physical properties must be evaluated (several times per outer loop iteration) if temperature and composition dependent physical properties are used. It is the physical properties calculations that generally dominate the computational cost of chemical process simulation problems

(Westerberg et al., 1979; Baden and Michelson, 1987). A second problem can arise if any of the iteration loops are hard to converge. One might well wonder what is the point of requiring subsets of the complete set of equations to be satisfied exactly (by iteration) when the estimate of the tear variables is not the correct solution. Might it not be a better strategy to solve all of the MESH equations simultaneously using Newton's method (or one of its relatives)? By guessing enough unknown variables to permit *all* other quantities in the MESH equations to be calculated explicitly *and* by *not requiring* that any of the equations be satisfied until complete convergence has been achieved we might avoid both of these problems with tearing methods.

It is not completely clear who first implemented a simultaneous correction method for solving multicomponent distillation and absorption problems. As is so often the case, it would appear that the problem was being tackled by a number of people independently. Simultaneous solution of *all* the MESH equations was suggested as a method of last resort by Friday and Smith (1964) in a classic paper analysing the reasons why other algorithms fail. They did not, however, implement such a technique. The two best known and most frequently cited papers are those of Goldstein and Stanfield (1970) and Naphtali and Sandholm (1971), the latter providing more details of an application of Newton's method described by Naphtali at an AIChE meeting in May 1965. Naphtali (1965), Goldstein and Stanfield (1970) and Wang and Wang (1980) cite a communication by Wang and Oleson (1964), but their work is not in the public domain. As described by Naphtali (1965), the method of Wang and Oleson employs just 2 independent equations and variables per stage and is not, therefore, a full simultaneous correction method.

To the best of our knowledge, a method to solve all the MESH equations for all stages at once using Newton's method was first implemented by Whitehouse (1964) (see, also, Stainthorp and Whitehouse, 1967). Among other things, Whitehouse's code allowed for specifications of purity, T , V , L or Q on any stage. Interlinked systems of columns and nonideal solutions could be dealt with even though no examples of the latter type were solved by Whitehouse.

It is interesting to note that the early work in this field was done prior to the publication of many of the methods of estimating thermodynamic properties taken for granted today (NRTL, UNIQUAC and UNIFAC methods for liquid phase activity coefficients and SRK, PR and other equations of state) and before some algorithms widely used today were developed. Since the pioneering work of Whitehouse, Naphtali and Sandholm and Goldstein and Stanfield, many others have employed Newton's method or one of its relatives to solve the MESH equations (Roche, 1971; Ishii and Otto, 1973; Shah and Bishnoi, 1978; Christiansen et al., 1979; Magnussen et al., 1979; Ferraris and Morbidelli, 1982 to name but a few).

Simultaneous correction procedures have shown themselves to be generally fast and reliable, having a locally quadratic convergence rate in the case of Newton's method, and these methods much less sensitive to difficulties associated with nonideal solutions than are tearing methods. Extensions to the basic method to include complex column configurations, interlinked columns (see Section 8), nonstandard specifications (Ferraris, 1981), and applications to column design (Ricker and Grens, 1976) result in only minor changes in the algorithm. In addition, simultaneous correction procedures can easily incorporate stage efficiencies within the calculations (something that is not always possible with other algorithms).

13.4.1 Designing a Simultaneous Solution Procedure

Seader (1986) lists a number of things to be taken into consideration when designing a simultaneous correction method; a revised and extended list follows and is discussed in more detail below.

1. What equations should be used?
2. What variables should be used?
3. How should the equations be ordered?
4. How should the variables be ordered?
5. How should the linearized equations be solved?
6. Should J be updated on each iteration or should it be held constant for a number of iterations or should it be approximated using quasi-Newton methods. Should derivatives of physical properties be retained in the calculation of J ?
7. Should flexibility in specifications be provided and, if so, how?
8. What criterion should we use to determine convergence?
9. How should the initial guess be obtained?
10. What techniques should we use to improve reliability?

13.4.2 What Equations and Variables Should be Used?

The possible choices include: the $2c + 3$ formulation, the $2c + 1$ formulation, the $c + 3$ formulation or some other, equivalent set of equations?

The variables that go along with these alternative formulations are $(L_j, V_j, T_j, y_{ij}, x_{ij})$ in the $2c+3$ formulation, (l_j, v_j, T_j) in the $2c+1$ formulation and (L_j, V_j, T_j, x_{ij}) in the $c+3$ formulation. Goldstein and Stanfield (1970) used the $c+3$ formulation of the MESH equations while Naphtali (1965) and Naphtali and Sandholm (1971) used the $2c+1$ formulation.

The advantage of the $c+3$ formulation is that the much smaller number of equations means that much less computer time will be required to solve the linearized system of equations. The disadvantage is that the vapor phase composition dependence of the K -values cannot be properly accounted for. This limits this formulation to systems which exhibit ideal behavior in the vapor phase. Departures from equilibrium normally are handled through some form of stage efficiency factor. Both the $2c+1$ and $2c+3$ formulations are easily modified to handle stage efficiencies without losing some of the highly desirable Jacobian structures we note below. The same cannot be said of the $c+3$ formulation.

In the $2c+1$ formulation, the material balance equations are linear and will, if Newton's method is used, be satisfied on every iteration after the first. Thus, the problem reduces to solving the equilibrium relations and energy balance equations. In the $2c+3$ formulation the material balance equations are quadratic and will require a few iterations before being satisfied. The $2c+1$ formulation is probably the best choice for single columns. For interlinked systems the $2c+3$ formulation may be better depending on just how much flexibility in specifications is allowed.

13.4.3 How should the Equations and Variables be Ordered?

Goldstein and Stanfield (1970) grouped the equations by type (M+E-S-H) (as also, and somewhat earlier, did Whitehouse, 1964), whereas Naphtali and Sandholm (1971) grouped the equations and variables by stage. Grouping the equations and variables by stage is preferred for systems with more stages than components (practically all distillation and many absorption and extraction problems) while grouping by type is preferred for systems with more components than stages (some gas absorption problems).

13.4.4 Computation of the Jacobian

To evaluate the Jacobian, one must obtain the partial derivative of each function with respect to every variable. Part of the appeal of the grouping by stage formulation is that for standard specifications (and some others), the Jacobian matrix is *block tridiagonal*

codes this differentiation is done numerically. This can be an extremely time consuming step. However, neglect of these derivatives is not recommended unless one is dealing with nearly ideal solutions, since, to do so, will almost certainly lead to an increase in the required number of iterations or even to failure. Michelson (1986) and Baden and Michelson (1987) have shown that coding analytical derivatives of the complicated physical property models is well worth while as this leads to substantial reductions in the computer time required for this portion of the calculations.

Almost all of the partial derivatives needed in *ChemSep* are computed from analytical expressions. The exception is the temperature derivatives of the excess enthalpy which requires a second differentiation with respect to temperature of the activity and fugacity coefficient models. In the case of non-standard specifications (which allow "wild" specification such as the temperature on a stage in the middle of a column, instead at one of its ends) the specification equation must be calculated using finite differencing.

In order to reduce the time required to calculate the necessary derivative information for the Jacobian, a class of methods, termed quasi-Newton methods, has been used. In these methods, approximations to the Jacobian are made and updated through the use of formulae derived to satisfy certain constraints which have been chosen to force the approximate Jacobian to mimic the behavior of the actual Jacobian. Experience with quasi-Newton methods has been reported by Gallun and Holland (1980), Lucia and coworkers (Lucia and Macchietto, 1983; Lucia and Westman, 1983; Westman, Lucia and Miller, 1984, Venkat and Lucia, 1987) and others. It is probably fair to say that pure quasi-Newton methods are not especially useful for separation process problems.

In an effort both to decrease the number of iterations required by a pure quasi-Newton method and to try to maintain the region of convergence of Newton's method, Lucia and coworkers have developed what they call a hybrid approach to the solution of sets of model equations. In this hybrid approach, any derivative information which is easy (i.e., inexpensive) to calculate is included in a "computed" part of the Jacobian, **C**, while derivative information which is difficult or expensive to obtain is included in an "approximated" part of the Jacobian, **A**

$$\mathbf{J} = \mathbf{C} + \mathbf{A} \quad (13.31)$$

In equilibrium stage calculations **A** includes derivatives of activity and fugacity coefficients with respect to composition. On each iteration in the solution procedure, **C** is calculated exactly and **A** is updated using some appropriate quasi-Newton formula. The result is a method that, while taking more iterations than Newton's method to solve a given problem, generally takes about 50% less computer time to converge (when compared to numerical differentiation but *not* when compared to analytical differentiation) and has nearly the same region of convergence as Newton's method.

Another possibility is to selectively retain the derivatives of physical properties in the calculation of the Jacobian. For dynamic simulations this has shown to be counterproductive (Kooijman, 1995) and therefore *ChemSep* doesn't support such a technique.

13.4.5 How Should the Linearized Equations be Solved?

The Jacobian matrix above is of order $s(2c + 3)$. Thus, the number of elements in the matrix is $s^2(2c + 3)^2$. Since the vast majority of the elements are zero it is absolutely essential to take account of the sparsity of J when solving the linearized equations; straightforward matrix inversion is totally impractical (not to say numerically impossible). Software for solving sparse linear systems is described by Seader (1985b).

Linear systems with a block tridiagonal structure may efficiently be solved using a generalized form of the Thomas algorithm, one where the scalar arguments are replaced by matrices and divisions are replaced by premultiplication by inverse matrices. The steps of this algorithm are given by Henley and Seader (1981). Still further improvements in the block elimination algorithm for solution of separation process problems can be effected if we take advantage of the that the special structure of the submatrices **A**, **B** and **C**, in fact, **A** and **C** are nearly empty (Baden, 1984; Baden and Michelson, 1987).

If the column has pumparounds extra matrices will be present which are *not* on the diagonal and the use of block tridiagonal methods becomes less straight forward. Similar problems arise with non-standard specifications that are not on the variables of the condenser and reboiler. When we solve multiple interlinked columns (currently not supported by *ChemSep*) a special ordering is required to maintain the diagonal structure of the jacobian. Therefore, *ChemSep* now uses a sparse Gaussian elimination solver (NSPIV) to solve the linear system of equations.

13.4.6 The Convergence Criterion

Convergence of Newton's method in *ChemSep* is assessed by looking at the residuals of the equations that make up the model. These function-values are squared and summed to give one single convergence parameter: the sum of squares (SS). In *ChemSep* we compare the square root of this sum of squares with a user specified convergence criterion. Typical values for this convergence criterion vary between 10^{-6} for equilibrium column and flash calculations to 10^{-3} for nonequilibrium column calculations.

As *ChemSep* does not scale the equations it is possible that this sum of squares remains too large for such small convergence criterions. As this is not always foreseeable, we extended the convergence test in *ChemSep* with a check on the change in the variables. If the absolute relative value of the change of each variable is smaller than the convergence criterion, *ChemSep* also considers the problem solved.

Finally, in order to prevent calculations from continuing endlessly, there is a maximum number of iterations after which the program pauses. *ChemSep* will then ask the user whether to continue the calculations, and for how many additional iterations. Typically, we find that 30 Newton iterations suffices to solve most separation problems at hand. Often, when more iterations are needed, there are other problems that prevent the method from converging. For example, specifications can be impossible to attain, or the iterations oscillate between two or more sets of values, neither of which represents a solution to the model equations.

13.4.7 How Should the Initial Guess be Obtained?

In order to obtain convergence, Newton's method requires that reasonable initial estimates be provided for all $N(2c + 3)$ independent variables. It is obviously impractical to expect the user of a SC method to guess this number of quantities. Thus, the designer of a computer code implementing a SC method must provide one or more methods of generating initial estimates of all the unknown variables from, at most, one or two user supplied values of end stage temperatures and flow rates.

There is no standard method for generating initial estimates of the flow rates, temperatures and compositions. Temperatures are often easy to estimate using linear interpolation between two (or more) guessed values. Flow rates, too, can be estimated fairly easily by assuming constant molar flows from stage to stage. Estimating the compositions of the vapor and liquid phases is a little more difficult. Possible approaches include performing one (or more) iterations of a plate to plate calculation procedure (Burton and Morton, 1987) or a tridiagonal matrix method such as the Wang and Henke algorithm for distillation or the sum rates method for absorption (see, for example, Wayburn and Seader, 1983). Alternative procedures that have been used include:

1. short cut calculation procedures (Venkat and Lucia, 1987)
2. relaxation techniques (Ketchum, 1979)
3. simply setting the composition of each stream equal to the composition of the combined feeds (Furzer, 1986)

4. flashing the combined feeds at the average column pressure and assuming the resulting composition of the vapor and liquid streams hold for each stage.

ChemSep uses an *automatic* initialization procedure that does not require the user to make *any* estimates. Flow rates are estimated assuming constant molar flows from stage to stage. If the bottoms flow rate and reflux ratio are **NOT** specified and cannot be estimated from the specifications that are supplied then the bottoms flow rate is arbitrarily assigned a value of half the total feed flow and the reflux ratio is given a default value of 2. This, of course, could cause serious convergence problems. Therefore, *optional guesses* have been added to the specifications to circumvent this problem (you need to toggle an interface switch to make them visible).

Mole fractions of both phases and temperatures are estimated by carrying out a few iterations of the Bubble Point method described above. The first time through this procedure we use K-values estimated assuming ideal solution behavior at the column average pressure and an estimate of the boiling point of the combined feeds. (This eliminates the normal requirement of estimates of end stage temperatures). Two more iterations of the BP method are carried out using the (possibly nonideal) K-value model actually selected.

For columns without condenser and reboiler a different method is used for the initialization. The composition of the liquid streams are set equal to the top liquid feed composition and the composition of the vapor equal to the bottom vapor feed composition. Temperatures in the column are set equal to the temperature of the first feed specified. For columns with either a condenser or a reboiler - but not both - the compositions are initialized to the overall compositions of all feeds combined, and the temperatures to the bubble point at the column pressure and the overall feed compositions.

Currently, no special methods are employed for columns with pumparounds. Columns with large pumparound flows may require the flows to be estimated by the user, or, to repetitively solve the problem using the old results as the initialization and increasing the pumparound flow.

13.4.8 Reliability

SC methods are far more reliable and versatile than most other methods. The same method will solve distillation, gas absorption and liquid extraction problems. It must be admitted though, that although the probability that Newton's method will converge from the automatic initial estimates is quite high, there is no guarantee of convergence. The difficulty of supplying good initial estimates is most severe for problems

involving strongly nonideal mixtures, interlinked systems of columns and nonstandard specifications.

Several methods have been used to improve the reliability of Newton's method; damping the Newton step (Naphtali and Sandholm, 1971), use of steepest descent (ascent) formulations for some of the iterations (Powell, 1970), and combination with relaxation procedures (Ketchum, 1979); none of which has proven to be completely satisfactory. The methods most recently proposed for assisting convergence of Newton's method are continuation methods discussed in a later section.

In default mode, *ChemSep* does NOT use any of these techniques, other than a check to make sure that all quantities remain positive. Mole fractions, for example, are not permitted to take on negative values. The user does have the option of supplying damping factors.

13.4.9 Damping Factors

The Newton's method in *ChemSep* has some extra features that will enhance the convergence to the solution. Newton's method computes a new estimate of the solution based on the current Jacobian and function vector. However, the new solution vector might be physically meaningless, for example if a mole fraction becomes less than zero or larger than unity. Also, the new solution vector might represent too large a change in stage temperatures or flows for the method to be stable. To alleviate these problems *ChemSep* allows damping of the Newton iteration. Temperature changes are limited to a maximum number of degrees per iteration (the default to 10 K) and flow changes up to a maximum fraction of the old flows (the default fraction is 0.5). The compositions require a special type of damping. If a mole fraction is going to be negative or larger than unity, the change is limited to half the distance to the relevant extreme value. In addition, if a damping factor is specified, the maximum change in composition equals the factor (the default factor is 1 allowing a change over the whole mol fraction range). This type of damping has proven to be very effective. The damping factors can be found under the Options - Solve options menu.

If, for some reason, your column simulation does not converge, changing the damping factors might help. If you know the iteration history (by either limiting the number of iterations or by printing out the intermediate answers. Both can be done in the Options - Solve options menu) you can adapt the damping factors so the column simulation might converge. Note that convergence tends to be slower when you start to apply extra damping by making the factors smaller, Newton method loses its effectiveness when damped. Also, damping does not guarantee convergence.

13.4.10 User Initialization

For difficult problems it might be necessary for the user to provide initial temperature, pressure, or flow profiles.

The user can specify either temperature or flow profiles, or both. The only requirement is that values for the first and last stages are provided. Missing temperatures on intermediate stages are computed by linear interpolation, missing flows are computed on a constant flow from stage to stage basis. Therefore, it is better to specify the flows of the first and last two stages in case a condenser and reboiler are present. Composition profiles are computed through the method described above, however, temperatures are not computed using the bubble point calculations. If both user specified temperature and flow profiles are incomplete *ChemSep* switches to the automatic initialization method.

13.4.11 Initialization with Old Results

In some cases it is useful to use the converged results of a previous run as the initial guess for a new - similar - problem (for example, when the bottoms flow rate and reflux rate are not specified and cannot be estimated, and the automatic initialization always uses a reflux ratio of 2). This is a very straightforward way of specifying the initial guess as long as the number of components remains the same. Care must be taken when feed or product specifications or locations are changed. The results are interpolated if the number of stages is changed.

13.5 Continuation Methods for Difficult Problems

An unconstrained Newton's method should converge at least 75% of all separation process problems with no particular difficulty. Adding bells and whistles in the form of damping factors, line searches and the like will increase the probability of success to around 95%. However, the problems that Newton's method fails to solve demand much more than 5% of the engineers time as he/she attempts to make the method converge. Continuation methods introduced in Section 11.3 are intended to alleviate this problem. There are three categories of continuation method that have been used for solving the equilibrium stage MESH equations:

1. Mathematical methods which place the MESH equations into a *homotopy* equation of purely mathematical origin. In this category we find the Newton homo-

topy used by Wayburn and Seader (1983), Bhargava and Hlavacek (1984) and many others.

2. Physical continuation methods in which the nature of the equations being solved is exploited in some way. In this category we have the thermodynamic homotopies of Vickery and Taylor (1986a,b) and a related method due to Frantz and Van Brunt (1986a,b). Parametric continuation methods also belong in this category (Jelinek et al. (1973) and others).
3. Proprietary continuation methods, about which very little can be said (Byrne and Baird, 1983; 1985).

We might also note that continuation methods may be combined in the same program.

The first to use the Newton homotopy for separation process problems were Salgovic et al. (1981). They integrated the HDEs with a high order differential equation solving technique and sufficiently small step sizes to ensure that no significant buildup of error had occurred (i.e. $\mathbf{x}(1)$ was indeed the solution to $\mathbf{F}(\mathbf{x}) = 0$). They found that while this method was reliable, the number of function evaluations became quite large. Salgovic et al. also solved their test problems by switching to Newton's method to solve the final problem after a prescribed number of Euler steps was used to integrate the HDEs. No correction was used for of the intermediate problems; Newton's method was used only to solve the final problem. This method was found to be quite effective as long as the differential equations were solved to an appropriate accuracy. Unfortunately, the appropriate number of steps and the size of each step could *not* be determined a priori.

Wayburn and Seader (1983) used the Newton homotopy in their solution of the MESH equations for interlinked distillation columns. Initial estimates of the solution were obtained by assuming temperature and flow rate profiles throughout the columns and then obtaining composition estimates from one iteration of a Wang-Henke type method (see Wayburn and Seader, 1983). If Newton's method failed to solve the given problem from any of the starting guesses, the Newton homotopy was used. They parameterized the problem in terms of the arclength of the curve being traced out in the solution space and used an Euler step to proceed from one intermediate problem to the next and corrected the resulting estimate of the solution vector for the functions using Newton's method in the plane orthogonal to the unit normal vector at the point being considered. The problems they solved involve the "ideal" system benzene - toluene - o-xylene and included some problems with nonstandard specifications.

Using these same procedures, Chavez et al. (1986) were able to find multiple solutions for some systems of interlinked columns; a Petlyuk system and a fractionator with side stripper used to separate a mixture of benzene, toluene and o-xylene. Problem specifications included the purity of the three product streams as well as a bottoms flow rate for the Petlyuk configuration. While it was not possible to find all the solutions at a given reflux ratio through use of Newton's method alone, the homotopy method was successful in all cases. Lin et al. (1987) have shown how all these solutions may be computed from a single starting point.

Bhargava and Hlavacek (1984) have made use of an implementation of the Newton homotopy that they found to be both effective and efficient for the nonideal problems with which they dealt. In it, they do only one Newton correction on the problems leading to the final problem to be solved. The estimate of the solution to the next problem is taken to be the result of that iteration; there is no attempt to use a predictor step to obtain the solution to the next problem. In their paper, they found that an even step length in the range 0.2-0.5 was sufficient to solve their problems. However, this suggestion was based upon the results of some problems which could be solved by Newton's method without any particular difficulty.

Kovach and Seider (1986) used the Newton homotopy for solving azeotropic distillation problems in conjunction with another homotopy for the liquid-liquid equilibrium and three phase equilibrium problems arising in the distillation calculations. They used the Newton homotopy to find a solution to the azeotropic distillation problems in much the same way that Wayburn and Seader did. From this solution they defined another homotopy which allowed them to look for multiple solutions to the MESH equations and effectively to perform parametric sensitivity calculations. This second homotopy as well as the homotopies used for the liquid-liquid equilibrium calculations fall more appropriately in the section on physically based homotopies and will be discussed there.

Burton and Morton (1987) (see, also, Burton, 1986) modified an equation oriented flowsheeting system so that the Newton homotopy could be called upon. A number of example problems involving distillation were solved using essentially the methods described by Wayburn and Seader (1983). The method was not able to find solutions in every case.

Ellis et al. (1986) used Algorithm 6.4 with a parameterized form of the Newton homotopy. They integrated the HDEs using Euler's method and a semi-implicit Runge-Kutta method, both with Newton correction. In addition, Gear's method was applied directly to solve the homotopy equation $\mathbf{H}(\mathbf{x}, t) = \mathbf{0}$. Step size was adjusted as the integration proceeded. The results of three test problems show that Euler's method required fewest steps but Gear's method took the shortest time. The performance of

the Runge-Kutta method was disappointing.

Frantz and Van Brunt (1986a,b) and Vickery and Taylor (1986a,b) have also used the Newton homotopy (more on their work below).

13.5.1 Drawbacks to Mathematical Homotopies

The homotopy functions, $\mathbf{H}(\mathbf{x}, t)$, used in most algorithms are purely mathematical contrivances; they do not model a real separation process except when the continuation parameter takes on its final value. This, of itself, is not a serious difficulty; after all, the function is being used only as a means of obtaining the solution to a problem which *is* of physical origin. However, difficulties can arise if elements of x take on negative values at intermediate values of the continuation parameter. Again, of itself, this is not always a serious matter. However, most methods for estimating, for example, activity coefficients, demand at least nonnegative input and possibly normalized mole fractions in order to return real values. If, for example, one or more of the mole fractions were negative, a subroutine implementing such a method would be unable to deliver meaningful values. This situation was encountered by Vickery and Taylor (1986a,b) and by Burton and Morton (1987). Ellis et al. (1986) report negative total flow rates (a much less serious problem) at intermediate values of the continuation parameter.

13.5.2 Physical Continuation Methods

In order to be certain that a continuation method does not encounter difficulties with physically meaningless values of variables while solving intermediate problems, the method should be designed so that the equations $\mathbf{H}(\mathbf{x}, t)$ are physically meaningful for any value of the continuation parameter. Parametric continuation methods automatically meet this requirement, mathematical homotopies do not. A logical first step in *designing* a physically based homotopy function is to look for the reasons why algorithms used to solve the MESH equations fail. It is the nonlinearity of the equilibrium ratios or "K-values" and the enthalpies that are responsible for the majority of convergence difficulties with Newton's method. Enthalpies are normally expressed as the sum of an ideal enthalpy and an "excess" term that accounts for the nonidealities in the phase under consideration. The K_i may be estimated from.

$$K_i = \frac{y_i}{x_i} = \frac{\phi_i^L}{\phi_i^V} = \frac{\gamma_i P_i^*}{\phi_i^V p} \quad (13.32)$$

The vapor pressure p_i^* is a relatively simple function of temperature. Fugacity coefficients normally are calculated from equations of state (EOS), e.g. the Redlich-Kwong and Peng-Robinson EOS. The nonlinearities in these terms arise from the temperature dependence of the EOS and a composition dependence that manifests itself primarily through the rules used for the evaluation of mixture parameters. The liquid phase activity coefficient is obtained from a complicated function of composition and temperature, e.g. the van Laar, Wilson, NRTL, UNIQUAC or UNIFAC equations (see, for example, Walas, 1985).

13.5.3 Thermodynamic Homotopies

Vickery and Taylor (1986a,b) have designed a homotopy in which the thermodynamic quantities are modified, simplifying them for the initial problem so that the K-values and enthalpies are much easier to deal with. Then, as the homotopy parameter is increased, the thermodynamic quantities are brought back to their original forms, until, finally, the original problem has been solved. Here, we generalize their formulation by writing

$$K_i(T, x, y, t) = K_{i,simple}^{(1-g(t))} K_{i,actual}^{g(t)} \quad (13.33)$$

$$H_i(T, x, y, t) = H_{i,simple}^{(1-g(t))} H_{i,actual}^{g(t)} \quad (13.34)$$

where $K_{i,simple}$ is a simple model for the K-value and $K_{i,actual}$ is the rigorous expression above. H_{simple} is a simple model for the enthalpy and H_{actual} is the rigorous model. The functions $g(t)$ and $w(t)$ are zero at $t = 0$ and one at $t = 1$ but are otherwise arbitrary. Possible choices for $K_{i,simple}$ include:

1. unity
2. a real, positive number, different for each component
3. an ideal solution K-value

Vickery and Taylor took $g(t) = w(t) = t$ and $K_{i,simple} = p_i^*/p$ with p_i^* given by the Antoine equation. Powers et al. (1986) have used $K_{i,simple} = 1$ in another, related, application of these ideas. Choices for H_{simple} are discussed by Vickery and Taylor (1986a).

The evaluation of \mathbf{H}_x and \mathbf{H}_t is not quite as straightforward as it is in the linear homotopies. None the less, these quantities are quite easily calculated and do not require much extra work. The square matrix \mathbf{H}_x is still identically equal to the Jacobian

used in the Newton-Raphson iterations. The evaluation of \mathbf{H}_t is discussed by Vickery (1988).

Vickery and Taylor (1986a,b) found the thermodynamic homotopy to be rather more effective than the simple implementation of the Newton homotopy (Algorithm 6.4) for solving distillation problems involving strongly nonideal systems. Fewer steps and fewer iterations were needed by the thermodynamic homotopy and the Newton homotopy occasionally failed (for the reasons discussed above). Fidkowski et al. (1993) used a linear combination of ideal and nonideal K-values as part of a homotopy method for finding all the azeotropic points in multicomponent mixtures.

Frantz and Van Brunt (1986) used analogous arguments in developing physically based homotopies for application to hydrometallurgical solvent extraction models. In their work, they describe the development of custom imbedding techniques which allow a physically realistic model for the chemical equilibrium in the process flow-sheets they consider. The method they develop is shown to be superior to any of the mathematical homotopies they employed.

Since they were dealing with heterogeneous azeotropic systems, Kovach and Seider (1986) had to consider liquid-liquid phase splitting. To solve these problems they would start by assuming the two phases formed as almost pure binaries with the remaining components added as the homotopy parameter is changed. At the same time, the isoactivity of only the component being added is enforced at all parameter values. The isoactivity equations are solved in a manner reminiscent of the Newton homotopy. Kovach and Seider also showed that the two phase envelope for ternary systems could be traced quite easily by parameterizing the feed composition and following the phase split as the homotopy parameter is changed.

13.5.4 Parametric Continuation Methods

The reflux ratio or bottoms flow rate are parameters that have been used in parametric solutions of the MESH equations (Jelinek et al., 1973). Parametric continuation has also been used to detect multiple solutions of the MESH equations (Ellis et al., 1986; Kovach and Seider, 1986; Burton, 1986).

13.5.5 An Efficiency Based Continuation Method

A parameter occurring naturally in the MESH equations that makes a good continuation parameter is the stage efficiency, E^{MV} . The maximum separation possible on a

Table 13.5: Efficiency continuation

1. Choose initial (nonzero) value of E^{MV} and compute initial vector \mathbf{x}
2. Solve $\mathbf{F}(\mathbf{x}, E^{MV}) = 0$ at current value of E^{MV} using Newton's method
3. If E^{MV} is not unity, solve the homotopy differential equations

$$d\mathbf{x}/dE^{MV} = -H_x^{-1}H$$
4. Adjust E^{MV} , use Euler's method to approximate \mathbf{x} at $E^{MV} + \Delta E^{MV}$ and initialize \mathbf{x} for next problem in the sequence. Return to step 2.

given stage is obtained with an efficiency of unity. On the other hand, for a vanishingly small stage efficiency the stage performs no separation worth mentioning and the streams leaving the stage have essentially the same flow rates, composition and temperature as the combined feeds. This fact can be exploited in a simple yet "efficient" continuation method for solving difficult equilibrium stage separation process problems

Muller (1979) appears to have been the first to use the Murphree efficiency as a continuation parameter. Each problem in the sequence was solved using a stage to stage procedure. Sereno (1985) used the efficiency as a continuation parameter for solving liquid liquid extraction problems. Vickery et al. (1988) have used the efficiency as a continuation parameter in conjunction with Newton's method and the "HDEs" (with $t \equiv E^{MV}$) to solve multicomponent distillation problems in single and interlinked systems of columns. The algorithm of Vickery et al. (1988) is summarized in Table 13.5. *The initialization procedure*, Step 1, described in detail by Vickery et al. (1988), is completely automatic and *does not require any guesswork on the part of the user*. Although this method properly belongs to the class of parametric continuation methods, it was developed for solving difficult problems, not for parametric sensitivity studies. In this sense, it is closer in spirit to homotopy continuation methods.

The efficient continuation method is very effective at solving difficult problems involving "standard specifications"; it cannot, without modification, handle problems in which a product stream purity is specified. Part of the reason that this method is so effective is that the nonlinearities of the equilibrium relations are again reduced. The K-values are now multiplied (for the first problem) by a small number - the efficiency. Also, since the compositions vary little from tray to tray, the temperatures, excess enthalpies and activity coefficients do not change significantly from tray to tray. These two factors combine to make the MESH equations much easier to solve at very low values of E^{MV} .

13.5.6 Combined Physical Homotopies

An efficiency-based continuation method is very effective at solving problems involving standard specifications but cannot handle problems in which a product stream purity is specified. Woodman (1989, 1990) used efficiency-based continuation to solve a thermodynamically ideal problem involving standard specifications. Thermodynamic continuation was then used to bring in the solution nonidealities in a controlled way. Finally, parametric continuation was invoked to solve any problems that involved non-standard specifications.

13.5.7 Summary

If a well designed code for solving the MESH equations using Newton's method is available, then a simple implementation of the Newton homotopy, the thermodynamic homotopy or the efficiency based continuation method can be implemented very easily. A little extra effort is required to implement the HDEs and still more work is needed if a sophisticated step size algorithm and/or arc length parameterization is desired. However, the end result of all of this work will be a code that rarely will fail, as long as the problem being solved actually has a solution.

We would like to stress that Newton's method should usually be tried before resorting to a continuation method (at least until a continuation method is designed that can outperform Newton's method on problems where both work).

13.6 Relaxation Methods

Relaxation techniques differ from both tearing algorithms and simultaneous convergence methods in that the relaxation techniques do *not* solve *steady state* problems. Rather, at least one set of the MESH equations is cast in an unsteady state form. The equations are then integrated from some initial state (guess) until successive values of the variables do not change.

The first published results of application of the relaxation technique were given by Rose, Sweeney, and Schrodt (1958). In their method, the M equations alone were cast in an unsteady state form. Compositions were then calculated using Euler's method and the temperatures and flow rates calculated from the S and H equations. An improvement to the method was proposed by Ball (1961) in which Euler's method was replaced by an implicit integration scheme which led to the formulation of a tridiago-

nal set of algebraic equations. The H equations were written in unsteady state form by Vermeuil and Oleson (1971). Other authors have extended the idea to more complex columns and higher order integration schemes.

The appeal of the relaxation technique lies in the extreme stability of the method. As long as the time step stays within the region of stability for the differential equation solver, the method should converge to a solution. However, this convergence is generally a very slow process, slowing even more as the solution is approached. For this reason, the method is used (only) for problems which are very difficult to converge or for situations where a knowledge of how the steady state is achieved is important. Ketchum (1979) has proposed a compromise in which a relaxation technique is used in conjunction with a simultaneous correction procedure which gradually takes over as the rate of convergence due to relaxation becomes small.

13.7 Three Phase Distillation

There are many commercially important operations involving the distillation of mixtures which may form two distinct liquid phases. Steam distillation (the injection of live or "open" steam into the base of a distillation column) is widely used in the petrochemical industries for the recovery and purification of light aromatics such as benzene, toluene and xylene, for the recovery of gasoline by steam distillation from absorber oils and for the separation of many complex petroleum mixtures into fractions of narrow boiling range. Many of these systems will form two liquid phases on some or all of the trays in a distillation column; a water rich phase and a hydrocarbon rich phase. Azeotropic and extractive distillation are used in the chemical industry for separating mixtures that are hard to separate by fractional distillation. Many azeotropic systems form two-liquid phases either in a decanter or, additionally, on some of the stages in a column.

Examples of such processes include the distillation of many alcohol/water mixtures such as butanol-water, propanol-butanol-water (Block and Hegner, 1976), dehydration of ethanol using, as the entrainer, benzene (Ross and Seider, 1979; Prokopakis and Seider, 1983), trichloroethylene, diethyl ether, pentane, gasoline fractions or trimethylpentane (Furzer, 1985); dehydration of secondary butyl alcohol (Kovach and Seider, 1986b); dehydration of acetic acid and water using butyl acetate (Fuchs et al., 1983); dehydration of isopropanol with cyclohexane and the distillation of vinyl acetate - acrylonitrile - water and acetonitrile - acrylonitrile - water mixtures (Ferraris and Morbidelli, 1982). Furfural and water is used as a separating agent in the extractive distillation of butane/butadiene mixtures, a process that assumed enormous commercial importance in the 1940's for the production of butadiene for use in the manufacture of

synthetic rubber (Grohse 1948 and others). Grohse reported the existence of two (and sometimes even three) liquid phases on the trays of such columns.

The separation of fermentation products (Pucci et al., 1986); the production of fluorocarbons by reaction with carbon tetrachloride and hydrogen fluoride (Ross and Seider, 1980), the distillation of light hydrocarbons with ammonia (see van Winkle, 1967), the cryogenic distillation of nitrogen/natural gas mixtures (Yu et al., 1969) are other distillation processes that involves two liquid phases.

13.7.1 Equilibrium Stage Model of Three Phase Distillation

Design and simulation of three-phase distillation processes has also been carried out using equilibrium stage approaches and there has been a good deal of interest shown in this topic in the last few years. This resurgence of interest in the computer aided design and simulation of these operations has followed the development of thermodynamic models that are able to accurately account for the high degree of nonideality exhibited by these systems. The extension of the mathematical model to three-phase systems is quite straightforward, it is obtaining the numerical solution that is "difficult" (more on this below). The equations conventionally used to model the behavior of a two-phase stage are the same as those presented in Section 2. For those stages with three phases, the MESH equations are written as described in section 2 with some minor modifications:

1. Extra terms are added to the mass balance equations to allow for one more liquid stream into the stage and one more liquid stream out of the stage. the *Total Material Balance*:

$$M_j \equiv V_j + L'_j + L''_j - V_{j+1} - L'_{j-1} - L''_{j-1} - F_j = 0 \quad (13.35)$$

and the *Component Material Balances*:

$$M_{ij} \equiv V_j y_{ij} + L'_j x'_{ij} + L''_j x''_{ij} - V_{j+1} y_{i,j+1} - L'_{j-1} x'_{i,j-1} - L''_{j-1} x''_{i,j-1} - F_j z_{ij} = 0 \quad (13.36)$$

2. Two new sets of c equilibrium relations are added. The second liquid phase is in equilibrium with both the vapor phase and the other liquid phase.

$$E'_{ij} = K'_{ij} x'_{ij} - y_{ij} = 0 \quad (13.37)$$

$$E''_{ij} = K''_{ij} x''_{ij} - y_{ij} = 0 \quad (13.38)$$

$$E^*_{ij} = K^*_{ij} x''_{ij} - y_{ij} = 0 \quad (13.39)$$

However, a total of only $2c$ equilibrium relations are independent since

$$K_{ij}^* = K_{ij}''/K_{ij}' \quad (13.40)$$

3. One more summation equation is added to deal with the second liquid phase.

$$S_j^V = \sum_{i=1}^c y_{ij} - 1 = 0 \quad (13.41)$$

$$S_j^{L'} = \sum_{i=1}^c x_{ij}' - 1 = 0 \quad (13.42)$$

$$S_j^{L''} = \sum_{i=1}^c x_{ij}'' - 1 = 0 \quad (13.43)$$

4. Extra terms are added to the enthalpy balance to allow for one more liquid stream entering the stage and one more liquid stream exiting the stage.

$$\begin{aligned} H_{ij} \equiv & V_j H_{ij}^V + L_j' H_{ij}^{L'} + L_j'' H_{ij}^{L''} - V_{j+1} H_{i,j+1}^V \\ & - L_{j-1}' H_{i,j-1}^{L'} - L_{j-1}'' H_{i,j-1}^{L''} - F_j H_{ij}^F + Q_j = 0 \end{aligned} \quad (13.44)$$

This brings the total number of *independent* equations describing a stage with three phases on it to $3c + 4$ equations as compared to the $2c + 3$ required to describe a two phase separation stage.

These equations must be solved for all stages in the column in order to predict the product compositions and flow rates. However, the numerical solution of these equations is a decidedly nontrivial task. There are a number of reasons for this:

1. The very high nonlinearity of the thermodynamic quantities (the K-values and enthalpies) means that the numerical computations are unusually sensitive to estimates of the temperatures and compositions and convergence often is very difficult to obtain.
2. Unlike two phase separation process problems, the number of equations and variables necessary to determine the *operating* condition of a column is not known until the problem has been solved. The reason for this is that even though the process is called "three phase" distillation, there will almost certainly be a number (probably a majority) of stages with only one liquid phase present.
3. The set of model equations listed above always admits the ("trivial") two-phase solution; that is, one vapor and one liquid phase.

Many different algorithms for solving this set of equations have been proposed (Block and Hegner, 1976; Boston and Shah, 1979; Ross and Seider, 1980; Prokopakis et al, 1981; Ferraris and Morbidelli, 1981, 1982; Kinoshita et al, 1983; Schuil and Bool, 1985; Pucci et al., 1986; Kovach and Seider, 1986; Baden, 1984; Baden and Michelson, 1987). All of these methods are, in one sense or another, extensions of methods that have been found useful for solving two-phase distillation problems; for example, simultaneous solution of all of the model equations using Newton's method (Ferraris and Morbidelli, 1981,1982; Baden, 1984; Baden and Michelson, 1987; Cairns and Furzer, 1990), a multistage flash algorithm (Ferraris and Morbidelli, 1981 again), "bubble-point" type methods (Ferraris and Morbidelli, 1981, Block and Hegner, 1976; Kinoshita et al, 1983), "inside-out" methods in which the property evaluations are put in an outer iteration loop rather than in an inner loop as in the "bubble point" type methods (Boston and Shah, 1979; Ross and Seider, 1980; Prokopakis et al, 1981; Schuil and Bool, 1985) and homotopy-continuation methods (Kovach and Seider, 1986a,b), Woodman et al. (1989).

Since the number of liquid phases present on each tray in the separation unit is not generally known, the number of equations necessary for modelling the unit cannot be determined until the actual operating conditions of the column have been determined. In some algorithms, for example, the SC algorithm of Ferraris and Morbidelli, it is necessary to specify in advance which stages have two phases and which stages have three phases. This is an unsatisfactory feature of their method that makes convergence a rather uncertain adventure.

13.7.2 The Mixed K-value Model

Schuil and Bool (1985) extended the work of Niedzwieki et al. (1980) so that a three phase separation process unit could be modelled using existing *two* phase separation process computer programs. A similar approach is used in the SC method of Baden and Michelson (1987).

The basic idea is to treat the two liquid phases as a single phase with thermodynamic properties defined by an appropriate averaging of the two liquid phases present. If the stage being considered actually has only one liquid phase then the thermodynamic properties are those of the actual liquid. In either situation, the number of equations used to model a single stage is always known.

The two quantities for which a suitable averaging must be determined are the equilibrium ratios and the liquid phase enthalpy. Considering first the equilibrium ratios,

recall that the K-value for species i is defined by

$$K_i(T, x, y) = \frac{y_i}{x_i} \quad (13.45)$$

In this case, we will take x_i to refer to the overall mole fraction of component i in the two liquid phases.

To determine x_i in terms of the two existing liquid phase compositions, x'_i and x''_i , the following expression is used.

$$x_i = \frac{x'_i L' + x''_i L''}{L' + L''} \quad (13.46)$$

By defining K'_i to be y_i/x'_i and K''_i to be y_i/x''_i , equations (13.45) and (13.46) can be combined to yield

$$K_i = \frac{(L' + L'')K'_i K''_i}{K'_i L' + K''_i L''} \quad (13.47)$$

or if α is defined to be the fraction of the liquid that is found in the first (') phase (L'/L'')

$$K_i = \frac{K'_i K''_i}{\alpha K'_i + (1 - \alpha) K''_i} \quad (13.48)$$

Recalling that the K-value definition

$$K_i(T, x, y) \equiv \frac{y_i}{x_i} = \frac{\gamma_i P_i^*}{\phi_i^V p} \quad (13.49)$$

it can be seen that the only term forming the K-value which depends upon the conditions in the liquid phase is the liquid phase activity coefficient. The pure phase information as well as the vapor phase fugacity coefficient are the same for both K'_i and K''_i . With this information, it is possible to show that equation (13.48) effectively reduces to

$$\gamma_i = \frac{\gamma'_i \gamma''_i}{\alpha \gamma'_i + (1 - \alpha) \gamma''_i} \quad (13.50)$$

and the mixed K-value can be given by (13.48) using this mixed activity coefficient.

In order to calculate a mixed liquid phase enthalpy, start with

$$(L' + L'')H = L'H' + L''H'' \quad (13.51)$$

Then, dividing through by $(L' + L'')$ gives

$$H = H' + (1 - \alpha)H'' \quad (13.52)$$

These two phase combined properties replace the single liquid phase properties in a two phase separation process algorithm.

13.7.3 Efficiency of Three Phase Distillation

The equilibrium stage model, so widely used in distillation simulation and design, does not, of course, represent reality in that few stages actually operate at equilibrium. The usual way out of this difficulty is either to use overall efficiencies or to combine an equation for "stage efficiency" with the equilibrium relations. Stage efficiencies have a more fundamental basis than overall efficiencies (which are really nothing more than a disguised form of rules of thumb and operating experience) and, for two phase systems, can be estimated from appropriate correlations (e.g. the AIChE method; King, 1980). However, no methods have been developed that can be used to estimate the stage efficiency of a tray with two liquid phases on it.

Very few experimental studies on this interesting and, from this point of view, badly overlooked area of distillation technology have been published. Schoenborn et al. (1941) report overall column efficiencies for the distillation of trichloroethylene, toluene and water in laboratory scale columns. An investigation into the efficiency of the extractive distillation of butane/butadiene mixtures using furfural and furfural/water as the extractive agent was carried out by Grohse (1948). He found that a second liquid phase was present whenever the water weight fraction was in the range 6 to 9%. Both groups found that the overall efficiency was not affected by the number of liquid phases present (two or three). However, in the discussion that accompanies Schoenborn's paper, D.B. Keyes, G.G. Brown and D.F. Othmer separately warn against generalizing the finding of Schoenborn et al. stating that in some cases the presence of a second liquid phase adversely affects efficiency, sometimes it has no effect at all and on some occasions it actually helps the separating ability of the column. Brown writes that "the efficiency goes all to pot if you don't keep the water off the plates" and "in some cases the tray efficiency closely approaches zero".

Furzer (1985) and Cairns and Furzer (1987, 1990a) give tray to tray composition data for the dehydration of ethanol using 224-trimethylpentane. These experiments were carried out at total reflux. Data for only one experiment are given in the first paper but Furzer (1985) does present a graphical summary of the efficiency calculated for ten similar experiments. From these graphs it can be seen that the overall efficiency for the two phase and three phase regions are not always the same; efficiencies of these two regimes show no consistent trends and fluctuates with the column F-factor. Moreover, these efficiencies are, as implied by Furzer, dependent on the model used to estimate the activity coefficients (UNIFAC in this case). This finding is not so very surprising; efficiencies of strongly nonideal multicomponent fluids can be extremely sensitive to the methods used to estimate the thermodynamic properties (K-values).

This seems to cast further doubt on the suitability of the equilibrium stage - stage efficiency approach for such systems. Kovach and Seider (1986b) have reported some

plant data for the dehydration of secondary butyl alcohol. They simulated the column with an equilibrium stage model assuming an overall column efficiency of around 80%.

Schoenborn et al (1941) carried out some of their experiments in a glass column so that the action on the trays could be observed. According to Schoenborn et al., good mixing of the two liquids was observed and although the two phases could readily be distinguished, it was impossible to tell which phase (if any) was dispersed in the other. When samples of the liquid were withdrawn it appeared that the solvent phase was usually dispersed in the water phase. Grohse (1948) does not provide much detail on the nature of the dispersion except to imply that they too had good mixing. Furzer (1985) states that an emulsion existed on the trays in his column. Davies and Porter (1987) have recently published photographs that show that three phase systems tend to foam in the region around the binodal curve. It would also appear that, in some instances at least, the two liquid phases lie one above the other.

Only Seider and coworkers (Ross and Seider, 1980; Prokopakis et al, 1981; Kovach and Seider, 1986a) and Pucci et al (1986) have even attempted to handle departures from equilibrium on three phase trays. Ross and Seider have done this by introducing two Murphree-type stage efficiencies. In their calculations Ross and Seider provide arbitrary values for these stage efficiencies and, also for lack of any better information, assume that all efficiencies are equal, i.e. the two liquid phases are in equilibrium with each other. Two particularly interesting conclusions to arise from the study by Ross and Seider are

1. "The Murphree tray efficiency has an important effect on the location of 2 liquid phases and, in some cases whether two liquid phases appear".
2. The extension to account for two liquid phases not in equilibrium is nontrivial.

Since a second liquid phase may seriously affect the performance of the column, it is important to be able to accurately locate those trays on which a second liquid phase forms (so that it can be withdrawn by a sidestream if desired). Ross and Seider's calculations suggest that this cannot be done reliably, given that there is no sufficiently extensive base of stage efficiency data for this kind of distillation process.

13.8 Reactive Distillation

Much of the literature on RD modelling by equilibrium stage models is concerned primarily with the development of methods for solving the steady state EQ stage model.

For the most part such methods are more or less straightforward extensions of methods that had been developed for solving conventional distillation problems. The number of examples that illustrate most papers usually is rather limited, both in number as well as in the type of RD process considered (most often it is an esterification reaction). Only rarely is there any attempt to compare the results of simulations to experimental data (some exceptions are noted below). More and more of the more recent modelling studies are carried out using one or other commercial simulation package: Aspen Plus, Pro/II, HYSYS, and SpeedUp are the packages mentioned most often in the papers discussed in what follows. A comprehensive review of RD modelling has been given by Taylor and Krishna (2000).

Symbol List

Latin Symbols

c	number of components
E^{MV}	Murphree vapor efficiency
E	equilibrium or efficiency equation
f	feed component flow rate ($kmol/s$)
F	feed flow rate ($kmol/s$)
(F)	discrepancy function vector or column matrix of mass balance and efficiency equations
$G(x)$	"Easy" function
H	enthalpy ($J/kmol$)
H	heat balance equation (J/s)
H	homotopy function
H_t	derivative of homotopy function with respect to t
$[J]$	Jacobian matrix
K	K-value
l	liquid component flow rate ($kmol/s$)
L	liquid flow rate ($kmol/s$)
M	component material balance
p	pressure (Pa)
Q	heat load on stage (J/s)
r	ratio of side stream flow to interstage flow
R	rate equation
s	number of stages
S	summation equation
t	homotopy parameter

U	liquid sidestream flow ($kmol/s$)
v	vapor component flow rate ($kmol/s$)
V	vapor flow rate ($kmol/s$)
W	vapor sidestream flow ($kmol/s$)
(X)	column matrix of independent variables, or vector of variables
x	liquid phase mole fractions
y	vapor phase mole fractions
z	feed mole fractions

Greek Symbols

γ	activity coefficient
$[\Gamma]$	matrix of thermodynamic factors
ϕ	fugacity coefficient

Subscripts

i	component number
j	stage number
k	iteration number

Superscripts

*	saturated
F	pertains to feed
L	pertains to liquid phase
V	pertains to vapor phase

Matrix Notation

\square	square matrix
$()$	vector

References

- Amundson, N.R., and A.J. Pontinen, "Multicomponent Distillation Calculations on a Large Digital Computer", *Ind. Eng. Chem.*, Vol **50** (1958) p. 730.
- Amundson, N.R., and A.J. Pontinen and J.W. Tierney, *AIChE J.*, Vol. **5** (1959) p. 730.
- Ball, W.E., "Computer Programs for Distillation", paper presented at the AIChE 44-th National Mtg., New Orleans, Feb. (1961).
- Bhargava, R. and V. Hlavacek; "Experience with Adopting One-Parameter Imbedding Methods Toward Calculation of Countercurrent Separation Processes", *Chem. Eng. Commun.*, Vol. **28** (1984) p. 165.
- Billingsley, D.S., and G.W. Boynton, "Iterative Methods for Solving Problems in Multicomponent Distillation at the Steady State", *AIChE J.*, Vol. **17**, No. 1 (1971) p. 65.
- Block, U. and B. Hegner, "Development and Application of a Simulation Model for Three Phase Distillation", *AIChE J.*, Vol. **22** (1976) p. 582.
- Boston, J.F., and S.L. Sullivan, Jr., "A New Class of Solution Methods for Multicomponent, Multistage Separation Processes", *Can. J. Chem. Eng.*, Vol. **52**, 52 (1974) p. 52.
- Boston, J.F., and H.I. Britt, "A Radically Different Formulation and Solution of the Single-Stage Flash Problem", *Comp. Chem. Eng.*, Vol. **2** (1978) p. 109.
- Boston, J.F., "Inside-Out Algorithms for Multicomponent Separation Process Calculations" in "Computer Applications in Chemical Engineering Process Design and Simulation", *ACS Symposium Series No.*, 124 (1981) p. 135.
- Boston, J.F. and V.B. Shah, "An Algorithm for Rigorous Distillation Calculations with Two Liquid Phases", Paper presented at AIChE National Meeting, Houston, April (1979).
- Broyden, C.G., "A Class of Methods for Solving Nonlinear Simultaneous Equations", *Math. Comp.*, Vol. **19** (1965) p. 577.
- Bruno, J.W., J.L. Yanosik and J.W. Tierney, "Distillation Calculations with Nonideal Mixtures", Adv. Chem. Number 115, 122, Am. Chem. Soc. (1972).
- Burningham, D.W., and F.D. Otto, "Which Computer Design for Absorbers?", *Hydroc. Process.*, Vol. **40**, 10 (1967) p. 163.

- Burton, P.J., PhD Thesis in Chemical Engineering, Cambridge University (1986).
- Burton, P.J. and W. Morton, "Differential Arclength Homotopy Continuation in Equation Oriented Simulation", Proc. Conf. on The Use of Computers in Chemical Engineering (CEF 87), p59 (1987).
- Byrne, G.D. and L.A. Baird, "Research on the Convergence of a Distillation Code", paper presented at Spring National AIChE Meeting, March 1983, Houston, TX.
- Byrne, G.D. and L.A. Baird, "Distillation Calculations Using a Locally Parameterized Continuation Method", *Comp. Chem. Eng.*, Vol. **9**, 6 (1985) p. 593.
- Cairns, B.P. and I.A. Furzer, "Three Phase Azeotropic Distillation. Experimental Results", *I. Chem. E. Symp. Ser.*, 104 (1987) p. B505.
- Cairns, B.P. and I.A. Furzer, "Multicomponent Three Phase Azeotropic Distillation. 1. Extensive Experimental Data and Simulation Results", *Ind. Eng. Chem. Res.*, Vol. **29** (1990a) pp. 1349–1363.
- Cairns, B.P. and I.A. Furzer, "Multicomponent Three Phase Azeotropic Distillation. 2. Phase Stability and Phase Splitting Algorithms", *Ind. Eng. Chem. Res.*, Vol. **29** (1990b) pp. 1364–1382.
- Cairns, B.P. and I.A. Furzer, "Multicomponent Three Phase Azeotropic Distillation. 3. Modern Thermodynamic Models and Multiple Solutions", *Ind. Eng. Chem. Res.*, Vol. **29** (1990b) pp. 1364–1382.
- Chavez C, R., J.D. Seader and T.L. Wayburn; "Multiple Steady State Solutions for Interlinked Separation Systems", *Ind. Eng. Chem. Fundam.*, Vol. **25**, (1986) p. 566.
- Cho, Y.S. and B. Joseph, "Reduced Order Steady State and Dynamic Models for Separation Processes", *AIChE J.*, Vol. **29** (1983) pp. 261–269, 270–276.
- Cho, Y.S. and B. Joseph, "Reduced Order Models for Separation Columns - III: Application to Columns with Multiple Feeds and Sidestreams", *Comput. Chem. Eng.*, Vol. **8**, (1984) pp. 81–90.
- Christiansen, L.J., M.L. Michelson and A. Fredenslund, *Comput. Chem. Eng.*, Vol. **3** (1979) p. 535.
- Ellis, M.F., R. Koshy, G. Mijares, A. Gomez-Munoz and C.D. Holland, "Use of Multi-point Algorithms and Continuation Methods in the Solution of Distillation Problems", *Comput. Chem. Eng.*, Vol. **10** (1986) p. 433.

- Ferraris, G.B., "Interlinked, Multistaged Separators with Nonstandard Specifications Solved by the Newton-Raphson Method", *AIChE J.*, Vol. **27**, 1 (1981) p. 163.
- Ferraris, G.B. and M. Morbidelli, "Distillation Models for Two Partially Immiscible Liquids", *AIChE J.*, Vol. **27** (1981) p. 881.
- Ferraris, G.B. and M. Morbidelli, "An Approximate Mathematical Model for Three Phase Multistage Separators", *AIChE J.*, Vol. **27** (1982) p. 881.
- Ferraris, G.B. and M. Morbidelli, "Mathematical Modelling of Multistaged Separators with Mixtures whose Components Have Largely Different Volatilities", *Comput. Chem. Enmg.*, Vol. **6** (1982) p. 303.
- Fletcher, J.P. "A Method for the Rigorous Calculation of Distillation Columns Using a Generalized Efficiency Model", Proc 4-th International Symposium on Distillation A437, *I. Chem. E.*, Brighton (1987)
- Frantz, R.W., L.N. O'Quinn and V. Van Brunt, "Rate Process Based Continuation Applied to Hydrometallurgical Solvent Extraction", Paper presented at AIChE National Meeting, Miami Beach, November (1986a).
- Frantz, R.W., L.N. O'Quinn and V. Van Brunt, "Stability of a Steady State Hydrometallurgical Solvent Extraction Model", Paper presented at AIChE National Meeting, Miami Beach, November (1986b).
- Friday, Ph.D. Thesis, Purdue University (1963).
- Friday, J.R. and B.D. Smith, "An Analysis of the Equilibrium Stage Separations Problem - Formulation and Convergence", *AIChE J.*, Vol. **10** (1964) p. 698.
- Fuchs, R., M. Gipsper and J. Gaube, "Calculation of Ternary Vapor Liquid Liquid Equilibria for Design of Three Phase Distillation", *Fluid Phase Equilibria*, Vol. **14** (1983) p. 325.
- Furzer, I.A., "Ethanol Dehydration Efficiencies Using UNIFAC", *AIChE J.*, Vol. **31** (1985) p. 1389.
- Gallun, S.E. and C.D. Holland, "Solve More Distillation Problems — Part V: For Highly Nonideal Solutions", *Hydroc. Process.*, Vol. **55**, 1 (1976) p. 137.
- Gallun, S.E. and C.D. Holland, "A Modification of Broyden's Method for the Solution of Sparse Systems with Application to Distillation Problems Described by Nonideal Thermodynamic Functions", *Comput. Chem. Eng.*, Vol. **4** (1980) p. 93.

Grohse E.W., "Plate Efficiencies in the Separation of C4 Hydrocarbons by Extractive Distillation", PhD Dissertation, University of Delaware (1948).

Goldstein, R.P. and R.B. Stanfield, "Flexible Method for the Solution of Distillation Design Problems using the Newton-Raphson Technique", *Ind. Eng. Chem. Process Des. Dev.*, **9**, (1970) p. 78.

Haas, J.R. "Rigorous Distillation Calculations", Chapter 4 in Kister, H. *Distillation Design*, McGraw-Hill, New York (1992).

Hausen, H. "The Definition of the Degree of Exchange on Rectifying Plates for Binary and Ternary Mixtures", *Chemie Ingr. Tech.*, Vol. **25** (1953) pp. 595–597.

Henley, E.J., and J.D. Seader, *Equilibrium-Stage Separation Operations in Chemical Engineering*, Wiley (1981).

Hidalgo S., R., A. Correa V., A. Gomez M. and J.D. Seader, "An Optimal Arrangement of Simultaneous Linearized Equations for General Systems of Interlinked, Multistaged Separators", *AIChE J.*, Vol. **26**, 4 (1980) p. 585.

Hlavacek, V. and P. van Rompay, "Calculation of Parametric Dependence and Finite-Difference Methods", *AIChE J.*, Vol. **28**, 6 (1982) p. 1033.

Hlavacek, V. and P. van Rompay, "Simulation of Countercurrent Separation Processes via a Global Approach", *Comp. Chem. Eng.*, Vol. **9**, 4 (1985) p. 343.

Hofeling, B.S. and J.D. Seader, "A Modified Naphtali-Sandholm Method for General Systems of Interlinked, Multistage Separators", *AIChE J.*, Vol. **24**, 6 (1978) p. 1131.

Holland, C.D., *Multicomponent Distillation*, Prentice Hall, Inc., NJ (1963).

Holland, C.D. and McMahan, K.S. "Comparison of Vaporization Efficiencies with Murphree — type Efficiencies in Distillation — I.", *Chem. Eng. Sci.*, Vol. **25** (1972) pp. 431–436.

Holland, C.D., *Fundamentals and Modelling of Separation Processes*, Prentice Hall, Inc., NJ (1975).

Holland, C.D., *Fundamentals of Multicomponent Distillation*, McGraw-Hill, Inc.; New York (1981).

Holland, C.D., and G.P. Pendon, "Solve More Distillation Problems — Part I: Improvements Give Exact Answers", *Hydroc. Process.*, Vol. **53**, 7 (1974) p. 148.

Holland, C.D., G.P. Pendon, and S.E. Gallun, "Solve More Distillation Problems — Part III: Application to Absorbers", *Hydroc. Process.*, Vol. **54**, 1 (1975) p. 101.

Huber, W.F., *Hydroc. Process.*, August (1977) p. 121.

Ishii, Y. and F.D. Otto, "A General Algorithm for Multistage Multicomponent Separation Calculations", *Can J. Chem. Eng.*, Vol. **51** (1973) p. 601.

Jelinek, J., V. Hlavacek and M. Kubicek, "Calculation of Multistage Countercurrent Separation Processes — I. Multicomponent Multistage Rectification by Differentiation with respect to an Actual Parameter", *Chem. Eng. Sci.*, Vol. **28** (1973) p. 1555.

Krishna, R., R.Taylor, "Multicomponent Mass Transfer — Theory and Applications", Chapter 7 of Handbook of Heat and Mass Transfer Operations, Volume II, Gulf Publishing Company, Houston (1986).

Ketchum, R.G., "A Combined Relaxation-Newton Method as a Global Approach to the Computation of Thermal Separation Processes", *Chem. Eng. Sci.*, Vol. **34** (1979) p. 387.

King, C.J., *Separation Processes*, Second Edition, McGraw Hill (1980).

Kinoshita, M., I. Hashimoto and T. Takamatsu, "A New Simulation Procedure for Multicomponent Distillation Column Processing of Nonideal Solutions or Reactive Solutions", *J. Chem. Eng. Japan*, Vol. **16**, 5 (1983) p. 370.

Kovach, J.W. and W.D. Seider, "Heterogeneous Azeotropic Distillation — Homotopy-Continuation Methods", Paper presented at AIChE National Meeting, Miami Beach, November (1986a).

Kovach, J.W. and W.D. Seider, "Heterogeneous Azeotropic Distillation — Experimental and Simulation Results", Paper presented at AIChE National Meeting, Miami Beach, November (1986b).

Kubicek, M., V. Hlavacek and F. Prochaska, "Global Modular Newton - Raphson Technique for Simulation of an Interconnected Plant Applied to Complex Rectification Columns", *Chem. Eng. Sci.*, Vol. **31** (1976) p. 277.

Lapidus, L., *Digital Computation for Chemical Engineers*, McGraw Hill (1962).

Lewis, W.K. and G.L. Matheson, "Studies in Distillation: Design of Rectifying Columns for Natural And Refinery Gasoline", *Ind. Eng. Chem.*, Vol. **24**, 5 (1932) p. 496.

Lin, W-J., J.D. Seader and T.L. Wayburn, "Computing Multiple Solutions to Systems

of Interlinked Separation Columns”, *AIChE J.*, Vol. **33** (1987) p. 886.

Lucia, A., and S. Macchietto, “A New Approach to the Approximation of Quantities Involving Physical Properties Derivatives in Equation-Oriented Process Design”, *AIChE J.*, Vol. **29** (1983) p. 705.

Lucia, A. and K.R. Westman , “Low-Cost Solutions to Multistage, Multicomponent Separation Problems by a Hybrid Fixed Point Algorithm”, *Proceedings of the Second International Conference on Foundations of Computer-Aided Process Design*, A.W. Westerberg and H.H. Chen, eds., CACHE (1984) p. 741.

Magnussen, T., M.L. Michelson and A. Fredenslund, “Azeotropic Distillation Using UNIFAC”, *I. Chem. E. Symp. Ser.*, 56, Proc. International Symposium on Distillation, London, (1979).

Medina, A.G., Ashton, N. and McDermott, C. “Murphree and Vaporization Efficiencies in Multicomponent Distillation”, *Chem. Eng. Sci.*, Vol. **33** (1978) pp. 331–339.

Medina, A.G., Ashton, N. and McDermott, C. “Hausen and Murphree Efficiencies in Binary and Multicomponent Distillation”, *Chem. Eng. Sci.*, Vol. **34** (1979) pp. 1105–1112.

Muller, F.R., PhD Thesis in Chemical Engineering, ETH Zurich (1979).

Murphree, E.V. “Rectifying Column Calculations”, *Ind. Eng. Chem.*, Vol. **17** (1925) pp. 747–750; 960–964.

Naphtali, L.M., “The Distillation Column as a Large System”, presented at AIChE 56-th National Mtg., San Francisco, May 16, (1965).

Naphtali, L.M. and D.P. Sandholm, “Multicomponent Separation Calculations by Linearization”, *AIChE J.*, Vol. **17**, 1 (1971) p. 148.

Newman, J.S., Hydrocarbon Processing, *Petrol. Refiner.*, Vol. **42**, 4 (1963) p. 141.

Niedzwiecki, J.L., R.D. Springer and R.G. Wolfe, “Multicomponent Distillation in the Presence of Free Water”, *Chem. Eng. Prog.*, Vol. **76** (1980) p. 57.

Pierucci, S.J., E.M. Ranzi, G.E. Bardi and M. Dente; “T-Method Computes Distillation”, *Hydroc. Process.*, Sept. (1981) p. 179.

Pierucci, S. and E. Ranzi, “T Method for Distillation Columns”, *Chem. Eng. Commun.*, Vol. **14** (1982) p. 1.

- Pierucci, S.J., E.M. Ranzi and G.E. Biardi, "Corrected Flowrates Estimation by Using (Convergence Promoter for Distillation Columns)", *AIChE J.*, Vol. **29**, 1 (1983) p. 113.
- Powell, M.J.D., "A Hybrid Method for Nonlinear Equations", in *Numerical Methods for Nonlinear Algebraic Equations*, P. Rabinowitz, Ed., Gordon and Breach (1970).
- Powers, M.F., D.J. Vickery and R. Taylor, "Simulation of Azeotropic and Extractive Distillation Operations Using a Nonequilibrium Stage Model", Paper presented at AIChE National Meeting, Miami Beach, November (1986).
- Pucci, A., P. Mikitenko and L. Asselineau, "Three Phase Distillation, Simulation and Application to the Separation of Fermentation Products", *Chem. Eng. Sci.*, Vol. **41** (1986) p. 485.
- Roche, E.C., "General Design Algorithm for Multistage Counter Current Equilibrium Processes", *Brit. Chem. Eng.*, Vol. **16** (1971) p. 821.
- Rose, A., R.F. Sweeney, and V.N. Schrodtt, "Continuous Distillation Calculations by Relaxation Method", *Ind. Eng. Chem.*, Vol. **50**, 5 (1958) p. 737.
- Ross, B.A. and W.D. Seider, "Simulation of Three Phase Distillation Towers", *Comput. Chem. Eng.*, Vol. **5** (1981) p. 7.
- Russell, R.A., "A Flexible and Reliable Method solves single tower and crude distillation problems", *Chem. Eng* October 17 (1983) pp. 53–59.
- Saeger, R. and P.R. Bishnoi, "A Modified 'Inside-Out' Algorithm for Simulation of Multistage Multicomponent Separation Processes Using the UNIFAC Group Contribution Method", *Can. J. Chem. Eng.*, Vol. **64** (1986) pp. 759–767.
- Salgovic, A., V. Hlavacek and J. Ilavsky, "Global Simulation of Countercurrent Separation Processes via One-Parameter Imbedding Techniques", *Chem. Eng. Sci.*, Vol. **36**, 10 (1981) p. 1599.
- Schoenborn, E.M., J.H. Koffolt and J.R. Withrow, "Rectification in the Presence of an Insoluble Component", *Trans AIChE* (1941).
- Schuil J.A. and K.K. Bool, "Three-Phase Flash and Distillation", *Comp. Chem. Eng.*, Vol. **9**, 3 (1985) p. 295.
- Seader, J.D., "The BC (Before Computers) and AD of Equilibrium Stage Operations", *Chem. Eng. Ed.*, Spring, 88, (1985a).

Seader, J.D., "Computer Modeling of Chemical Processes", *AIChE Monograph Series 15*, Vol. **81** (1985b).

Seader, J.D. and E.J. Henley *Separation Process Principles*, Wiley, New York (1998).

Seader, J.D., C.C. Rafael and T.L. Wayburn; "Multiple Solutions to Systems of Interlinked Distillation Columns by Differential Homotopy Continuation", paper 63f, AIChE National Meeting, San Francisco, CA; Nov. (1984).

Sereno, A.M., Doctoral Thesis, University of Porto, Portugal (1985).

Shah, M.K. and P.R. Bishnoi, *Can. J. Chem. Eng.*, Vol. **56** (1978) p. 478.

Smith, B.D., *Design of Equilibrium Stage Processes*, McGraw-Hill, New York (1964).

Sridhar, L.N. and A. Lucia, "Analysis of Multicomponent Separation Processes: Fixed Temperature and Fixed Pressure Profiles" *Ind. Eng. Chem. Res.*, Vol. **29** (1989) pp. 1668–1675.

Sridhar, L.N. and A. Lucia, "Analysis and Algorithms of Multicomponent Separation Processes", *Ind. Eng. Chem. Res.*, Vol. **29** (1990a) pp. 793-803.

Sridhar, L.N. and A. Lucia, "Tearing Algorithms for Separation Process Simulation", *Comp. Chem. Eng.*, Vol. **14** (1990b) pp. 901–905.

Stadtherr, M.A. and M.A. Malachowski, "On Efficient Solution of Complex Systems of Interlinked Multistage Separators", *Comp. Chem. Eng.*, Vol. **6**, 2 (1982) p. 121.

Stainthorp, F.P., and P.A. Whitehouse, "General Computer Programs for Multi Stage Counter Current Separation Problems — I: Formulation of the Problem and Method of Solution", *I. Chem. E. Symp. Series*, Vol. **23** (1967) p. 181.

Standart, G.L. "Studies on Distillation — V. Generalized Definition of a Theoretical Plate or Stage of Contacting Equipment", *Chem. Eng. Sci.*, Vol. **20** (1965) pp. 611–622.

Standart, G.L. "Comparison of Murphree Efficiencies with Vaporization Efficiencies", *Chem. Eng. Sci.*, Vol. **26** (1971) pp. 985–988.

Stewart, W.E. "Simulation of Fractionation by Orthogonal Collocation", *Chem. Eng. Sci.*, Vol. **40** (1985) pp. 409–421.

Sujata, A.D., "Absorber-Stripper Calculations Made Easier", *Hydroc. Process.*, Vol. **40**, 12 (1961) p. 137.

- Swartz, C.L.E. and W.E. Stewart, "A Collocation Approach to Distillation Column Design", *AIChE J.*, Vol. **32**, (1986) pp. 1832–1838.
- Taylor, R. and R. Krishna, "Modelling Reactive Distillation", *Chem. Eng. Sci.*, Vol. **55** (2000).
- Thiele, E.W., and R.L. Geddes, "Computation of Distillation Apparatus for Hydrocarbon Mixtures", *Ind. Eng. Chem.*, Vol. **25** (1933) p. 289.
- Tierney, J.W. and J.A. Bruno, "Equilibrium Stage Calculations", *AIChE J.*, Vol. **13**, 3 (1967) p. 556.
- Tierney, J.W., and J.L. Yanosik, "Simultaneous Flow and Temperature Correction in the Equilibrium Stage Problem", *AIChE J.*, Vol. **15**, 6, (1969) p. 897.
- Tomich, T.F., "A New Simulation Method for Equilibrium Stage Processes", *AIChE J.*, Vol. **16**, 2 (1970) p. 229.
- Van Winkle, M., *Distillation*, McGraw-Hill Book Co., New York, NY (1967).
- Verneuil, V.S., and A.P. Oleson, "Steady-state Distillation via Transient-state Calculations", presented at 161-st ACS National Mtg., Los Angeles, March (1971).
- Vickery, D.J. and R. Taylor, "Path Following Algorithms to the Solution of Multicomponent, Multistage Separation Process Problems", *AIChE J.*, Vol. **32**, 4 (1986a) p. 547.
- Vickery, D.J. and R. Taylor, "A Thermodynamic Continuation Method for the Solution of Multicomponent, Multistage Separation Process Problems", paper 50e AIChE Spring Nat. Mtg., New Orleans, April (1986b)
- Vickery, D.J., J.J. Ferrari and R. Taylor, "An 'Efficient' Continuation Method for the Solution of Difficult Equilibrium Stage Separation Process Problems", *Comput. Chem. Eng.*, (1987)
- Vickery D.J., PhD Thesis in Chemical Engineering, Clarkson University (1987)
- Waggoner, R.C. and C.D. Holland, "Solution of Problems Involving Conventional and Complex Distillation Columns at Unsteady State Operation", *AIChE J.*, Vol. **11**, 1 (1965) p. 113.
- Wang, J.C. and G.E. Henke, "Tridiagonal Matrix for Distillation", *Hydroc. Process.*, Vol. **45**, 8 (1966) p. 155.

- Wang, J.C. and Y.L. Wang; "A Review on the Modeling and Simulation of Multi-Stage Separation Processes", *Foundations of Computer-Aided Chemical Process Design*, Vol. II; R.S.H. Mah and W.D. Seider, eds.; Engineering Foundation; (1981) p. 121.
- Wang, Y.L., and A.P. Oleson, cited as private communication in Wang and Wang, 1981, (1964).
- Walas, S.M. *Phase Equilibria in Chemical Engineering*, Butterworth Publishers, (1985).
- Wayburn, T.L. Ph.D. Thesis, Univ. of Utah (1983).
- Wayburn, T.L. and J.D. Seader; "Solutions of Systems of Interlinked Distillation Columns by Differential Homotopy-Continuation Methods", *Proceedings of the Second International Conference on Foundations of Computer-Aided Process Design*, A.W. Westerberg and H.H. Chen, eds., CACHE (1983) p. 765.
- Wayburn, T.L., and J.D. Seader, "Degree Theory and Homotopy: Tools for Computer-Aided Process Design", paper 27d, AIChE Nat. Meeting, Nov. 25-30, San Francisco, CA, Nov. (1984).
- Whitehouse, P.A., PhD Thesis in Chemical Engineering, University of Manchester Institute of Science and Technology, April (1964).
- Westerberg, A.W., H.P. Hutchison, R.L. Motard and P. Winter, *Process Flowsheeting*, Cambridge University Press, (1979).
- Westman, K.R., A. Lucia and D. Miller, "Flash and Distillation Calculations by a Newton-like Method", *Comp. Chem. Eng.*, Vol. **8** (3/4), (1984) p. 219.
- Woodman, M.R., W. Morton and W.R. Paterson, "Simulation of the Distillation of Three Phase Mixtures: A new Algorithm Predicts New Phenomena", Paper presented at CHISA, Prague, (1990).
- Yu, P., I.M. Elshayal and B.C-Y. Lu, "Liquid-liquid-vapor Equilibria in the Nitrogen Methane Ethane System", *Can. J. Chem. Eng.*, Vol. **47** (1969) p. 495.

Chapter 14

Nonequilibrium Column Modelling

The nonequilibrium model and the model equations are described in detail in three chapters. In this chapter we focus our attention on those aspects of the nonequilibrium model that pertain to all such operations. Subsequent chapters include details of the model that are specific to gas/vapor-liquid separations (distillation and absorption) and to liquid-liquid extraction.

14.1 The Model Equations

A second generation nonequilibrium model was developed by Taylor and coworkers and is described in detail by Taylor *et al.* (1994) and Taylor and Krishna (1993). It can be used to simulate trayed columns as well as packed columns. A schematic diagram of a nonequilibrium stage is shown in Figure 14.1. This stage may represent one (or more than one) tray in a trayed column or a section of packing in a packed column. The vertical wavy line in the middle of the diagram represents the interface between the two phases which may be vapor and liquid (distillation), gas and liquid (absorption) or two liquids (extraction).

Figure 14.1 also serves to introduce the notation used in writing down the equations that model the behavior of this nonequilibrium stage. The flow rates of the light and heavy phases leaving the j -th stage are denoted by V_j and L_j respectively. The mole

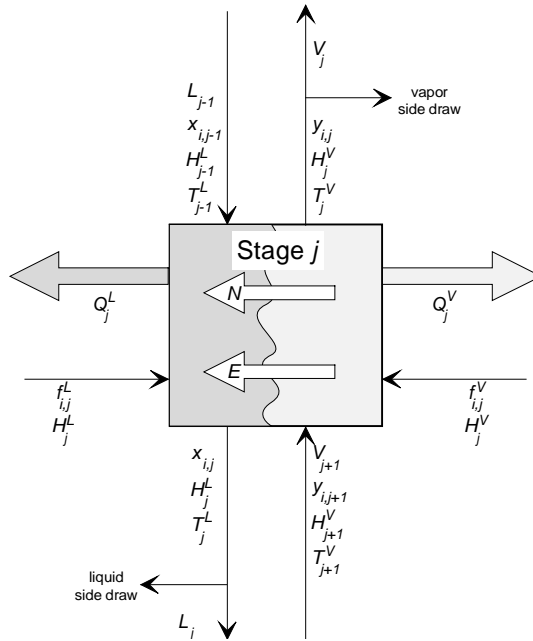


Figure 14.1: Schematic diagram of a nonequilibrium stage (after Taylor and Krishna, 1993).

fractions in these streams are y_{ij} and x_{ij} . The N_{ij} are the molar fluxes of species i on stage j . When multiplied by the area available for interphase mass transfer we obtain the rates of interphase mass transfer. The temperatures of the two phases are not assumed to be equal and we must allow for heat transfer as well as mass transfer across the interface.

If Figure 14.1 represents a single tray then the term ϕ_j^L is the fractional heavy phase entrainment defined as the ratio of the moles of heavy phase entrained in the light phase in stage j to the moles of downwards flowing heavy phase from stage j . Similarly, ϕ_j^V is the ratio of the light phase entrained in the heavy phase leaving stage j . For packed columns, this term represents axial dispersion. Weeping in tray columns may be accounted for with a similar term.

The component **material** balance equations for each phase may be written as follows:

$$M_{ij}^V \equiv (1 + r_j^V + \phi_j^V)V_j y_{ij} - V_{j+1} y_{i,j+1} - \phi_{j-1}^V V_{j-1} y_{i,j-1}$$

$$\begin{aligned}
& -f_{ij}^V - \sum_{\nu=1}^n G_{ij\nu}^V + N_{ij} \\
= & 0 \quad i = 1, 2, \dots, c
\end{aligned} \tag{14.1}$$

$$\begin{aligned}
M_{ij}^L & \equiv (1 + r_j^L + \phi_j^L)L_j x_{ij} - L_{j-1}x_{i,j-1} - \phi_{j+1}^L L_{j+1}x_{i,j+1} \\
& - f_{ij}^L - \sum_{\nu=1}^n G_{ij\nu}^L - N_{ij} \\
= & 0 \quad i = 1, 2, \dots, c
\end{aligned} \tag{14.2}$$

where $G_{ij\nu}$ is the interlinked flow rate for component i from stage ν to stage j , and n is the number of total stages (trays or sections of packing). The last terms in Equations (14.1) and (14.2) are the mass transfer rates (in $kmol/s$), where mass transfer from the “V” phase to the “L” phase is defined as positive. At the V/L interface we have continuity of mass and, thus, the mass transfer rates in both phases must be equal.

The total material balances for the two phases are obtained by summing Equations (14.1) and (14.2) over the component index i .

$$\begin{aligned}
M_{tj}^V & \equiv (1 + r_j^V + \phi_j^V)V_j - V_{j+1} - \phi_{j-1}^V V_{j-1} \\
& - F_j^V - \sum_{i=1}^c \sum_{\nu=1}^n G_{ij\nu}^V + N_{tj} = 0
\end{aligned} \tag{14.3}$$

$$\begin{aligned}
M_{tj}^L & \equiv (1 + r_j^L + \phi_j^L)L_j - L_{j-1} - \phi_{j+1}^L L_{j+1} \\
& - F_j^L - \sum_{i=1}^c \sum_{\nu=1}^n G_{ij\nu}^L - N_{tj} = 0
\end{aligned} \tag{14.4}$$

F_j denotes the total feed flow rate for stage j , $F_j = \sum_{i=1}^c f_{ij}$.

Here total flow rates and mole fractions are used as independent variables and total as well as component material balances are included in the set of independent model equations. In the nonequilibrium model of Krishnamurthy and Taylor (1985) component flow rates were treated as variables.

The nonequilibrium model uses two sets of **rate** equations for each stage:

$$R_{ij}^V \equiv N_{ij} - N_{ij}^V = 0 \quad i = 1, 2, \dots, c - 1 \tag{14.5}$$

$$R_{ij}^L \equiv N_{ij} - N_{ij}^L = 0 \quad i = 1, 2, \dots, c - 1 \tag{14.6}$$

where N_{ij} is the mass transfer rate of component i on stage j . The mass transfer rate in each phase is computed from a diffusive and a convective contribution with

$$N_{ij}^V = a_j^I J_{ij}^V + y_{ij} N_{tj} \tag{14.7}$$

$$N_{ij}^L = a_j^I J_{ij}^L + x_{ij} N_{tj} \tag{14.8}$$

where a_j^I is the total interfacial area for stage j and N_{tj} is the total mass transfer rate on stage j ($N_{tj} = \sum_{i=1}^c N_{ij}$). The diffusion fluxes J are given by (in matrix form):

$$(J^V) = c_t^V [k^V] (\overline{y^V - y^I}) \quad (14.9)$$

$$(J^L) = c_t^L [k^L] (\overline{x^I - x^L}) \quad (14.10)$$

where $(\overline{y^V - y^I})$ and $(\overline{x^I - x^L})$ are the average mole fraction difference between the bulk and the interface mole fractions (Note that the *fluxes* are multiplied by the interfacial *area* to obtain mass transfer *rates*). The average mole fraction differences are calculated depends on the chosen flow model (more on this topic later). The matrices of mass transfer coefficients, $[k]$, are calculated from

$$[k^P] = [R^P]^{-1} [\Gamma^P] \quad (14.11)$$

where $[\Gamma^P]$ is a matrix of thermodynamic factors for phase P . For systems where an activity coefficient model is used for the phase equilibrium properties the thermodynamic factor matrix Γ (order $c - 1$) is defined by

$$\Gamma_{ij} = \delta_{ij} + x_i \left(\frac{\partial \ln \gamma_i}{\partial x_j} \right)_{T,P,x_k, k \neq j=1 \dots c-1} \quad (14.12)$$

If equations of state are used γ_i is replaced by ϕ_i . Expressions for the composition derivatives of $\ln \gamma_i$ are given by Taylor and Kooijman (1991). The rate matrix $[R]$ (order $c - 1$) is a matrix of mass transfer resistances calculated from the following formulae:

$$R_{ii}^P = \frac{z_i}{k_{ic}^P} + \sum_{k=1, k \neq i}^c \frac{z_k}{k_{ik}^P} \quad (14.13)$$

$$R_{ij}^P = -z_i \left(\frac{1}{k_{ij}^P} - \frac{1}{k_{ic}^P} \right) \quad (14.14)$$

where k_{ij}^P are binary pair mass transfer coefficients for phase P . Mass transfer coefficients, k_{ij} , are computed from empirical models given in the next two chapters (see, also, Taylor and Krishna, 1993). Equations (14.13) and (14.14) are suggested by the Maxwell-Stefan equations that describe mass transfer in multicomponent systems (see Taylor and Krishna, 1993). The matrix of thermodynamic factors appears because the fundamental driving force for mass transfer is the chemical potential gradient and not the mole fraction or concentration gradient. This matrix is calculated from an appropriate thermodynamic model. Thus, the mass transfer coefficients are of central importance in the nonequilibrium model; indeed, sometimes it is possible to predict different behavior of a column by selecting different mass transfer coefficient correlations.

Note that there are $c(c-1)/2$ binary pair Maxwell-Stefan diffusion coefficients and $c-1$ times $c-1$ elements in the $[R^P]$ and $[k^P]$ matrices and, therefore, only $c-1$ equations per set of rate equations. This is the result of the fact that diffusion calculations only yield relative transfer rates. We will need an extra equation that will "bootstrap" the mass transfer rates: the energy balance for the interface.

The **energy** balance equations on stage j are written for each phase as follows:

$$\begin{aligned} E_j^V &\equiv (1 + r_j^V + \phi_j^V)V_j H_j^V - V_{j+1}H_{j+1}^V - \phi_{j-1}^V V_{j-1}H_{j-1}^V \\ &\quad - F_j^V H_j^{VF} - \sum_{\nu=1}^n G_{j\nu}^V H_{j\nu}^V + Q_j^V + e_j^V = 0 \end{aligned} \quad (14.15)$$

$$\begin{aligned} E_j^L &\equiv (1 + r_j^L + \phi_j^L)L_j H_j^L - L_{j-1}H_{j-1}^L - \phi_{j+1}^L L_{j+1}H_{j+1}^L \\ &\quad - F_j^L H_j^{LF} - \sum_{\nu=1}^n G_{j\nu}^L H_{j\nu}^L + Q_j^L - e_j^L = 0 \end{aligned} \quad (14.16)$$

where $G_{j\nu}$ is the interlink flow rate from stage ν to stage j . The last term in the left-hand-side of Equations (14.15) and (14.16), e_j , represents the energy transfer rates for the two phases and are defined by

$$e_j^V = a_j^I h^V (T^V - T^I) + \sum_{i=1}^c N_{ij}^V \bar{H}_{ij}^V \quad (14.17)$$

$$e_j^L = a_j^I h^L (T^I - T^L) + \sum_{i=1}^c N_{ij}^L \bar{H}_{ij}^L \quad (14.18)$$

where \bar{H}_{ij} are the partial molar enthalpies of component i for stage j . We also have continuity of the energy fluxes across the V/L interface which gives the interface energy balance:

$$E_j^I \equiv e_j^V - e_j^L = 0 \quad (14.19)$$

where h^V and h^L are the vapor and liquid heat transfer coefficients respectively, and T^V , T^I , and T^L the light phase, interface, and heavy phase temperatures. For the calculation of vapor phase heat transfer coefficients the Chilton-Colburn analogy between mass and heat transfer is used:

$$h^V = k\rho C_p \text{Le}^{2/3} \quad (14.20)$$

where

$$\text{Le} = \frac{\lambda}{DC_p\rho} = \frac{\text{Sc}}{\text{Pr}} \quad (14.21)$$

For the calculation of the liquid phase heat transfer coefficients a penetration model is used:

$$h^L = k\rho C_p \sqrt{Le} \quad (14.22)$$

where k is the average mass transfer coefficient and D the average diffusion coefficient.

In the nonequilibrium model of Krishnamurthy and Taylor (1985) the pressure was taken to be specified on all stages. However, column pressure drop is a function of tray or packing type and design and column operating conditions, information that is required for or available during the solution of the nonequilibrium model equations. It is, therefore, quite straightforward to add an **hydraulic** equation to the set of independent equations for each stage and to make the pressure of each stage (tray or packed section) an unknown variable. The stage is assumed to be at mechanical equilibrium so $p_j^V = p_j^L = p_j$.

In the second generation model, the pressure of the top tray (or top of the packing) is specified along with the pressure of any condenser. The pressure of trays (or packed sections) below the topmost are calculated from the pressure of the stage above and the pressure drop on that tray (or over that packed section). If the column has a condenser (which is numbered as stage 1 here) the hydraulic equations are expressed as follows:

$$P_1 \equiv p_c - p_1 = 0 \quad (14.23)$$

$$P_2 \equiv p_{spec} - p_2 = 0 \quad (14.24)$$

$$P_j \equiv p_j - p_{j-1} - (\Delta p_{j-1}) = 0 \quad j = 3, 4, \dots, n \quad (14.25)$$

where p_c is the specified condenser pressure, p_{spec} is the specified pressure of the tray or section of packing at the top of the column, and Δp_{j-1} is the pressure drop per tray or section of packing from section/stage $j-1$ to section/stage j . If the top stage is not a condenser, the hydraulic equations are expressed as

$$P_1 \equiv p_{spec} - p_1 = 0 \quad (14.26)$$

$$P_j \equiv p_j - p_{j-1} - (\Delta p_{j-1}) = 0 \quad j = 2, 3, \dots, n \quad (14.27)$$

In general we may consider the pressure drop to be a function of the internal flows, the fluid densities, and equipment design parameters.

$$\Delta p_{j-1} = f(V_{j-1}, L_{j-1}, \rho_{j-1}^V, \rho_{j-1}^L, Design) \quad (14.28)$$

The calculation of the pressure drop is discussed in the next two chapters.

Phase **equilibrium** is assumed to exist only at the interface with the mole fractions in both phases related by:

$$Q_{ij}^I \equiv K_{ij}x_{ij}^I - y_{ij}^I = 0 \quad i = 1, 2, \dots, c \quad (14.29)$$

where K_{ij} is the equilibrium ratio for component i on stage j . The K_{ij} are evaluated at the (calculated) temperature, pressure, and mole fractions at the interface.

The mole fractions must **sum** to unity in each phase:

$$S_j^V \equiv \sum_{i=1}^c y_{ij} - 1 = 0 \quad (14.30)$$

$$S_j^L \equiv \sum_{i=1}^c x_{ij} - 1 = 0 \quad (14.31)$$

as well as at the interface:

$$S_j^{VI} \equiv \sum_{i=1}^c y_{ij}^I - 1 = 0 \quad (14.32)$$

$$S_j^{LI} \equiv \sum_{i=1}^c x_{ij}^I - 1 = 0 \quad (14.33)$$

14.1.1 Degrees of Freedom and Specifications

Table 14.1 lists the type and number of equations for the nonequilibrium model. The model has $5c + 6$ equations and variables, where c is the number of components. Nonequilibrium and equilibrium models require similar specifications. Feed flows and their thermal condition must be specified for both models, as must the column configuration (number of stages, feed and sidestream locations etc.). Additional specifications that are the same for both simulation models include the specification of, for example, reflux ratios or bottom product flow rates if the column is equipped with a condenser and/or a reboiler. The specification of the pressure on each stage is necessary if the pressure drop is not computed; if it is, only the top stage pressure needs be specified (the pressure of all other stages being determined from the pressure drop equations that are part of the model described above).

14.1.2 Solving the Model Equations

The nonequilibrium model equations are solved simultaneously using Newton's method as described in Section 11.2. This requires initial estimates for all the variables. *Chem-*

Table 14.1: Nonequilibrium model equations type and number

Equation	Number
Material balances	$2c + 2$
Energy balances	3
transfer Rate equations	$2c - 2$
Summations equations	2
Hydraulic equation	1
interface eQuilibrium relations	c
Total MERSHQ	$5c + 6$

Table 14.2: Currently supported column internals

Bubble-cap trays
Sieve trays
Valve trays (including double weight valves)
Dumped packings
Structured packings
Equilibrium stage (with Murphree stage efficiency)
Rotating Disk Contactor (RDC) compartment (for extraction)

Sep uses the same automatic initialization procedure for the nonequilibrium model and the equilibrium model. The temperatures of the vapor, interface, and liquid are set equal to each other. Mass and energy transfer rates are initialized as zero and the interface mole fractions are set equal to the bulk mole fractions as estimated for the equilibrium stage model. Pressure drops initially are assumed to be zero.

14.2 Model Requirements

A nonequilibrium simulation cannot proceed without some knowledge of the column type and the internals column design. Equipment design details may be needed in order to calculate the mass transfer coefficients. Table 14.2 lists the currently supported types of column internals.

The nonequilibrium model requires the evaluation of many more physical properties and of the heat and mass transfer coefficients. For the evaluation of the heat and mass transfer coefficients, pressure drop, and the entrainment/weeping flows a nonequilibrium

rium simulation needs the following:

- Column internals type
- Column internals layout or design mode parameters (such as fraction of flooding etc.)
- Mass transfer coefficient model
- Flow model for both phases
- Entrainment and weeping models
- Pressure drop model

Flow models are discussed below. Mass transfer coefficient models are at the heart of the nonequilibrium model. All of the models incorporated in *ChemSep* are from the published literature. The choice of mass transfer coefficient can influence the results of a simulation. For this reason, we have tried – in the following two chapters that deal with distillation (and absorption) operations and with liquid-liquid extraction – to document the origin of each method in order to guide you in selecting models.

14.3 Flow Models

For the calculation of the diffusion fluxes the average mole fraction difference between the bulk and the interface mole fractions were required (see Equations 14.9 and 14.10). How these average mole fraction differences are computed depends on the flow model. Two flow models are discussed: mixed flow and plug flow.

14.3.1 Mixed flow

If we assume both phases are present in a completely mixed state, we can use

$$\overline{(y^V - y^I)} = (y^V - y^I) \quad (14.34)$$

$$\overline{(x^I - x^L)} = (x^I - x^L) \quad (14.35)$$

This keeps the rate equations (relatively) simple and only a function of the mole fractions leaving the current stage. However, on a tray where the vapor bubbles through a liquid which flows from one downcomer to the opposite downcomer this model is

not accurate. Indeed, only for very small diameter columns will the mixed flow model give reasonable results. The mixed model is the simplest flow model and is the easiest to converge. Convergence to the true column profiles for packed columns by using increasing number of stages can be quite slow using the mixed flow model.

14.3.2 Plug flow

In the plug flow model we assume that the vapor or liquid moves in plug flow (without mixing) through the froth. This complicates the rate equations so much that no exact solution is possible. The mass transfer rate equations need to be integrated over the froth. Kooijman and Taylor (1995) derived expressions for the average vapor and liquid compositions assuming constant mass transfer coefficients and that the interface compositions is constant (it isn't, but its "average" value is obtained):

$$\overline{(y^V - y^I)} = \Omega[-N^V](y^V - y^I) \quad (14.36)$$

$$\overline{(x^I - x^L)} = \Omega[-N^L](x^I - x^L) \quad (14.37)$$

where the mole fractions refer to the leaving streams and the number of mass transfer units (N) for the vapor and liquid are defined as:

$$N^V = c_t^V k^V a^V h_f A_b / V \quad (14.38)$$

$$N^L = c_t^L k^L a^L h_f A_b / L \quad (14.39)$$

$\Omega[M]$ is a matrix function defined as

$$\Omega[M] = [\exp[M] - [I]][M]^{-1}[\exp[M]]^{-1} = [\exp[-M] - [I]][-M]^{-1} \quad (14.40)$$

The efficiencies predicted for tray columns are more accurately represented with this model. The plug flow model can also be used for packed columns, providing much faster convergence to the true column profiles when compared to the mixed flow model.

Currently no correction terms is applied to the plug flow model to correct for the change in mole fractions over the integration (as is discussed by Kooijman and Taylor, 1995).

14.4 The Design Mode

The nonequilibrium column simulator has an optional *design mode* to automatically assign equipment design parameters. The user just needs to select one of the types

of internals (for each section in the column). The design-mode is activated simply by failing to specify the column diameter (leaving it as a "default" with "***") for a specific section. Other layout parameters can be specified but they may be changed by the design mode. Packed bed heights must be specified since this parameter determines the desired separation and the capacity.

The internal layout is determined after the flows have been estimated. Each stage in the column is designed separately and independently of adjacent stages. The different designs then are rationalized so that all stages within a particular section of the column have the same layout. During each iteration (that is, an update of the flows) each stage is re-designed only if the flowrates have changed by more than by a certain fraction (that can be specified). Only sections with re-designed stages are rationalized again. After convergence a complete design of any trayed or packed section in the column is obtained. Different types of column internal can be freely mixed in a column simulation/design.

The different design methods employed generate a column-design that might not be optimal from an engineers viewpoint. They must be seen as starting points for the actual design layouts. Also, the design does not include calculations to determine equipment support details, or the thicknesses of trays or the column.

Symbol List

Latin Symbols

c	Number of components, Molar concentration ($kmol/m^3$)
D	Binary diffusivity coefficient (m^2/s)
e	Energy transfer rate (J/s)
f_{ij}	Component i feed flow to stage j ($kmol/s$)
F_j	Total feed flow rate to stage j ($kmol/s$)
G	Interlinked flow rate ($kmol/s$)
h	Heat transfer coefficient ($J/m^2/K/s$)
H	Molar enthalpy ($J/kmol$)
\bar{H}_i	Partial molar enthalpy of component i ($J/kmol$)
J	Molar diffusion flux ($kmol/m^2s$)
k	Binary mass transfer coefficient (m/s)
K_i	K-value or equilibrium ratio component i : $K_i = y_i/x_i$
L	Liquid flow rate ($kmol/s$)

Le	Lewis number ($Le = Sc/Pr$)
M	Mass flow rate (kg/s)
N	Mass transfer rate ($kmol/s$)
n	Number of stages
p	Pressure (Pa)
Δp	Pressure drop (Pa)
Q	Heat input (J/s)
r	Ratio sidestream to internal flow
$[R]$	Matrix defined by (14.13) and (14.14)
T	Temperature (K)
V	Vapor flow rate ($kmol/s$)
x	Liquid mole fraction
y	Vapor mole fraction
z	Mole fraction

Greek Symbols

$[\Gamma]$	Thermodynamic matrix
λ	Heat conductivity ($W/m/K$)

Superscripts

I	Interfacial
L	Heavy phase
P	Phase P
V	Light phase

Subscripts

i	component i
j	stage j , component j
t	total
ν	from interlinking stage ν

References

H.A. Kooijman, R. Taylor, "Modelling Mass Transfer in Multicomponent Distillation", *Chem. Eng. J.*, Vol. **57**, No. 2 (1995) pp. 177–188.

R. Krishnamurthy, R. Taylor, "A Nonequilibrium Stage Model of Multicomponent Separation Processes. Part I: Model Description and Method of Solution", *AIChE J.*, Vol. **31**, No. 3 (1985) pp. 449–455.

R. Taylor, H.A. Kooijman, "Composition derivatives of Activity Models (for the estimation of Thermodynamic Factors in Diffusion)", *Chem. Eng. Comm.*, Vol. **102** (1991) pp. 87–106.

R. Taylor, H.A. Kooijman, J-S. Hung, "A second generation nonequilibrium model for computer simulation of multicomponent separation processes", *Comput. Chem. Engng.*, Vol. **18**, No. 3 (1994) pp. 205–217.

R. Taylor, R. Krishna, *Multicomponent Mass Transfer*, Wiley, New York (1993).

Chapter 15

Design Models for Trayed Columns

This chapter serves as a guide to the correlations in *ChemSep* for estimating mass transfer coefficients and the pressure drop for distillation (and absorption) trayed columns. The methods used in *Design Mode* to determine column design parameters also are discussed in detail. The chapter ends with the comparison of experimental distillation data and nonequilibrium model simulations.

15.1 Mass Transfer Coefficient Correlations

Table 16.1 provides a summary of the available correlations for trays and packings; the various correlations are discussed in more detail below. Recommended models are shown in boldface.

Binary mass transfer coefficients (MTC's) can be computed from the Number of Transfer Units (NTU's = N) by:

$$k^V = N^V / t_V a^V \quad (15.1)$$

$$k^L = N^L / t_L a^L \quad (15.2)$$

where the vapor and liquid areas are calculated with

$$a^V = a_d / \epsilon h_f \quad (15.3)$$

$$a^L = a_d / \alpha h_f \quad (15.4)$$

Table 15.1: Available mass transfer coefficient correlations per internals type

Bubble-Cap tray	Sieve tray	Valve tray
AIChE Hughmark	AIChE Chan-Fair Zuiderweg Chen-Chuang Harris Bubble-Jet	AIChE

the interfacial area density may be estimated from Zuiderweg's (1982) method (see below).

AIChE One of the oldest methods for estimating numbers of transfer units came from the AIChE tray efficiency research program of the 1950s. The correlations can be used for sieve trays, valve trays, and bubble-cap trays.

$$N^V = (0.776 + 4.57h_w - 0.238F_s + 104.8Q_L/W_l)/\sqrt{Sc_V} \quad (15.5)$$

$$N^L = 19700\sqrt{D^L}(0.4F_s + 0.17)t_L \quad (15.6)$$

where

$$F_s = u_s\sqrt{\rho_t^V} \quad (15.7)$$

$$Sc_V = \eta^V/\rho_t^V D^V \quad (15.8)$$

$$t_L = h_L ZW_l/Q_L \quad (15.9)$$

The clear liquid height h_L is computed from a correlation due to Bennett *et al.* (1983):

$$h_L = \alpha_e (h_w + C(Q_L/\alpha_e W_l)^{0.67}) \quad (15.10)$$

$$\alpha_e = \exp(-12.55(u_s(\rho^V/(\rho^L - \rho^V))^{0.5})^{0.91}) \quad (15.11)$$

$$C = 0.50 + 0.438 \exp(-137.8h_w) \quad (15.12)$$

Chan-Fair The number of transfer units for the vapor phase is:

$$N^V = (10300 - 8670FF)FF\sqrt{D^V}t_V/\sqrt{h_L} \quad (15.13)$$

$$t_V = (1 - \alpha_e)h_L/(\alpha_e u_s) \quad (15.14)$$

The AIChE correlation is used for the number of transfer units for the liquid phase. (h_L and α_e also are computed with the correlation of Bennett *et al.*).

Zuiderweg The vapor phase mass transfer coefficient is

$$k^V = 0.13/\rho_t^V - 0.065/(\rho_t^V)^2 \quad (15.15)$$

Note that k^V is independent of the diffusion coefficient. The liquid mass transfer coefficient is computed from either:

$$k^L = 2.6 \cdot 10^{-5}(\eta^L)^{-0.25} \quad (15.16)$$

or

$$k^L = 0.024(D^L)^{0.25} \quad (15.17)$$

The interfacial area in the spray regime is computed from:

$$a_d h_f = \frac{40}{\phi^{0.3}} \left(\frac{U_s^2 \rho_t^V h_L F P}{\sigma} \right)^{0.37} \quad (15.18)$$

and in the froth-emulsion regime from:

$$a_d h_f = \frac{43}{\phi^{0.3}} \left(\frac{U_s^2 \rho_t^V h_L F P}{\sigma} \right)^{0.37} \quad (15.19)$$

The transition from the spray to mixed froth-emulsion flow is described by:

$$F P > 3b h_L \quad (15.20)$$

where b is the weir length per unit bubbling area:

$$b = W_l / A_b \quad (15.21)$$

The clear liquid height is given by:

$$h_L = 0.6 h_w^{0.5} (p F P / b)^{0.25} \quad (15.22)$$

Hughmark The numbers of transfer units are given by:

$$N^V = (0.051 + 0.0105 F_s) \sqrt{\frac{\rho_L}{F_s}} \quad (15.23)$$

$$N^L = (-44 + 10.7747 \cdot 10^4 Q_L / W_l + 127.1457 F_s) \sqrt{D_L A_{bub} / Q_L} \quad (15.24)$$

Harris The numbers of transfer units are given by:

$$N^V = \frac{0.3 + 15 t_G}{\sqrt{Sc_G}} \quad (15.25)$$

$$N^L = \frac{5 + 10 t_L (1 + 0.17(0.82 F_s - 1)(39.3 h_w + 2))}{\sqrt{Sc_L}} \quad (15.26)$$

Chen-Chuang The numbers of transfer units for the vapor is:

$$t_V = \frac{h_l}{u_s} \quad (15.27)$$

$$F_s = U_s \sqrt{\rho_V} \quad (15.28)$$

$$N^V = 11 \frac{1}{\eta_L^{0.1} \beta^{0.14}} \left(\frac{\rho_L F_s^2}{\sigma^2} \right)^{1/3} \sqrt{D_V t_V} \quad (15.29)$$

and for the liquid

$$t_L = \frac{\rho_L}{\rho_V} t_V \quad (15.30)$$

$$N^L = 14 \frac{1}{\eta_L^{0.1} \beta^{0.14}} \left(\frac{\rho_L F_s^2}{\sigma^2} \right)^{1/3} \left(\frac{V}{L} \right) \sqrt{D_L t_L} \quad (15.31)$$

Bubble-Jet This is a combination of a theoretical model described by Taylor and Krishna (1993) and empirical models (Prado, 1986, Prado and Fair, 1990) to determine bubble sizes and velocities.

15.2 Pressure Drop Models

There are as many ways to compute tray pressure drops as there to estimate mass transfer coefficients. For packings there is a move away from the use of generalized pressure drop charts (GPDC) to more theoretically based correlations. We have chosen to employ the most recently published models. For packings there are at least 7 methods available (see Table 16.2). For packings operating above the loading point ($FF > 0.7$) we advise the use of models that take the correction for the liquid holdup into account, such as the SBF-89. However, disadvantage is that these models can have complex (imaginary) solutions, especially at high fractions of flood. This can cause non-convergence! The Lev-92 model can be an alternative for it includes a dependence on the liquid flow rate to simulate the increased pressure drop at loading conditions. When the pressure drop is specified as fixed, it is assumed zero.

15.2.1 Tray pressure drop Estimation

The liquid heights on the trays are evaluated from the tray pressure drop calculations. The wet tray pressure drop liquid height is calculated from:

$$h_{wt} = h_d + h_l \quad (15.32)$$

Table 15.2: Pressure drop correlations per internals type

Bubble-Cap tray	Sieve tray	Valve tray
Fixed Estimated	Fixed Estimated	Fixed Estimated

where h_d is the dry tray pressure drop liquid height and h_l the liquid height:

$$h_l = h_{cl} + h_r + \frac{h_{lg}}{2} \quad (15.33)$$

The clear liquid height, h_{cl} , is calculated with

$$h_{cl} = \alpha h_w + h_{ow} \quad (15.34)$$

where the liquid fraction of the froth, α , is computed with the Barker and Self (1962) correlation:

$$\alpha = \frac{0.37h_w + 0.012F_s + 1.78Q_L/W_l + 0.024}{1.06h_w + 0.035F_s + 4.82Q_L/W_l + 0.035} \quad (15.35)$$

The choice of correlation for the liquid fraction turns out to be important as certain correlations are dynamically unstable. The height of liquid over the weir, h_{ow} , is computed by various correlations for different types of tray weirs (see Perry's Handbook, 1984) and a weir factor (F_w) correction (see Smith, pp. 487) is employed. For example for a segmental weir:

$$h_{ow} = 0.664F_w \left(\frac{Q_L}{W_l} \right)^{2/3} \quad (15.36)$$

$$w = \frac{W_l}{D_c} \quad (15.37)$$

$$F_w^3 = \frac{w^2}{1 - (F_w w (\frac{1.68Q_L}{W_l^{2.5}})^{2/3} + \sqrt{1 - w^2})^2} \quad (15.38)$$

where Q_L is the volumetric flow over the weir per weir length. The residual height, h_r , is only taken into account for sieve trays. Bennett's method (see Lockett, 1986, pp. 81) is:

$$h_r = \left(\frac{6}{1.27\rho_L} \right) \left(\frac{\sigma}{g} \right)^{2/3} \left(\frac{\rho_L - \rho_V}{d_h} \right)^{1/3} \quad (15.39)$$

Dry tray pressure, h_d , is calculated with:

$$h_d = K \frac{\rho_G}{\rho_L} u_h^2 \quad (15.40)$$

$$K = \frac{\xi}{2g} \quad (15.41)$$

where the orifice coefficient ξ for sieve trays is computed as described by Stichlmair and Mersmann (1978). For valve trays we use the method of Klein (1982) as described in Kister (1992, pp. 309–312) where K is given for the cases with the valves closed or open. It is extended for double weight valve trays as discussed by Lockett (1986, pp. 82–86). The dry tray pressure drop is corrected for liquid fractional entrainment.

The froth density is computed from

$$h_f = \frac{h_{cl}}{\alpha} \quad (15.42)$$

The liquid gradient, h_{lg} , is estimated from a method due to Fair (Lockett, 1986, pp. 72):

$$R_h = \frac{Wh_f}{W + 2h_f} \quad (15.43)$$

$$U_f = \frac{Q_L}{Wh_{cl}} \quad (15.44)$$

$$\text{Re}_f = \frac{R_h U_f \rho_L}{\eta_L} \quad (15.45)$$

$$f = 7 \times 10^4 h_w \text{Re}_f^{-1.06} \quad (15.46)$$

$$h_{lg} = \frac{ZfU_f^2}{gR_h} \quad (15.47)$$

where W is the average flow-path width for liquid flow, and Z the flow path length. The height of liquid at the tray inlet is:

$$h_i = \sqrt{\frac{2}{g} \left(\frac{Q_L}{W_l} \right)^2 \left(\frac{1}{h_{cl}} - \frac{1}{h_c} \right) + \frac{2\alpha h_f^2}{3}} \quad (15.48)$$

where h_c is the height of the clearance under the downcomer. The pressure loss under downcomer (expressed as a liquid height) is

$$h_{udc} = \left(\frac{1}{2g} \right) \left(\frac{Q_L}{C_d W_l h_c} \right)^2 \quad (15.49)$$

where $C_d = 0.56$ according to Koch design rules. The height of liquid in the downcomer can now be calculated with the summation:

$$h_{db} = h_{wt} + h_i + h_{udc} \quad (15.50)$$

Liquid heights on bubble-cap trays are estimated from a method given in Perry's (1984) handbook and in Smith (1963). The liquid fraction of the froth is computed according to Kastanek (1970).

15.3 Entrainment and Weeping

Entrainment and weeping flows (for trays only) change the internal liquid flows and influence the performance of the column internals. *ChemSep* **currently does not support the handling of these flows**. This is due to the fact that few entrainment models behave properly. Neither is the effect of the entrainment and weeping flows on the mass transfer properly taken into account.

Entrainment can be estimated from the fractional liquid entrainment from Hunt's correlation and from Figure 5.11 of Lockett (1986) for sieve trays:

$$\phi^L = 7.75 \cdot 10^{-5} \left(\frac{0.073}{\sigma} \right) M_v \left(\frac{U_v}{T_s - 2.5h_{cl}} \right)^{3.2} \quad (15.51)$$

The weep factor is estimated from a figure from Smith (1963, p. 548), that was fitted to the following correlation

$$WF = \frac{0.135 \phi \ln(34(H_w + H_{ow}) + 1)}{(H_d + H_r)} \quad (15.52)$$

where ϕ is the open area ratio.

15.4 Packing Flooding and Minimum Wetting

The fraction of flooding for packings is computed by dividing the superficial gas velocity by the gas velocity at flood. The latter is found in an iterative process from the pressure drop correlation (keeping the liquid to vapor ratio constant!) and the specified pressure drop of flood. If no flood pressure drop is specified it is computed from the packing factor with the Kister and Gill correlation. If no pressure drop model is selected, the Leva's method is used, which also has the packing factor as only parameter.

The minimum operating conditions, similar to the weep point for tray operation, is set to the minimum wetting rate predicted by Schmidt (1979). Schmidt calculates a liquid falling number

$$C_L = 1633.6(0.062428\rho_L)(1000\sigma)^3/(1000\eta_L)^4 \quad (15.53)$$

and the minimum wetting rate is (in *USgal/min/ft²*):

$$Q_{MW} = \frac{0.3182C_L^{2/9}(1 - \cos \phi)^{2/3}}{\sqrt{(1 - T_L)a_p/0.3048}} \quad (15.54)$$

where ϕ is the liquid contact angle and T_L is the shear stress number:

$$T_L = 0.9FF^{-2.8} \quad (15.55)$$

with FF as the fraction of flood. The contact angle for metals is 10° (0.1745 rad). To obtain contact angles for other packings the contact angle is assumed inverse proportional to the critical surface tension, σ_c . For metals we have $\sigma_c = 0.075$, so the contact angle is computed with

$$\phi = 0.1745 \left(\frac{0.075}{\sigma_c} \right) \quad (15.56)$$

With this approach we obtain higher minimum wetting rates for plastic packings than for metal packings. The minimum operating or ‘weep’ fraction WF is then taken as

$$WF = \frac{Q_{MW}}{u_L} \quad (15.57)$$

where u_L is the superficial liquid flow.

15.5 Column Design

Tray layout parameters that specify a complete design (for the calculation of mass transfer coefficients and pressure drops) are shown in Table 15.3. For packings only the column diameter and bed height are design parameters, other parameters (such as void fraction, nominal packing diameter, etc.) are fixed once the type of packing has been selected. The packed bed height must be specified since it determines the desired separation and the capacity.

A very important parameter in tray column design is the system factor (SF). It represents the uncertainty in design correlations with regard to phenomena that are currently still not properly modeled, such as foaming.

Different design methods can be employed:

Fraction of flooding; this is the standard design method for trays, we have employed a modified version of the method published by Barnicki and Davis (1989).

Pressure drop; this is the usual design method for packed columns, but is very useful as well for tray design with pressure drop constraints.

other design methods can be thought of: ones that minimize pressure drop and cost, or maximize flexibility and efficiency. *ChemSep* has a modular structure to allow different design methods to be implemented.

Table 15.3: Tray layout data

General (sieve) tray layout data:	
Column diameter	Active area
Number of flow passes	Total hole area
Tray spacing	Downcomer area
Liquid flow path length	Weir length
Hole diameter	Weir height
Hole pitch	Deck thickness
Downcomer clearance	
Additional data for bubble caps:	
Cap diameter	Slot area
Slot height	Riser area
Skirt clearance	Annual area
Additional data for valves:	
Closed Loss K	Open Loss K
Eddy Loss C	Ratio Valve Legs
Valve Density	Valve Thickness
Fraction Heavy Valves	Heavy Valve Thickness

15.5.1 Tray Column Design: Fraction of flooding

The first task in this approach to tray design is to assign all layout parameters to consistent values corresponding to the required capacity defined by the fraction of flooding and current flowrates. These defaults function as starting points for subsequent designs.

The initial free area ratio is taken to be 15% of the active area. The active area is determined from a capacity factor calculation with internals specific methods (for sieve and bubble-cap trays the default is Fair's correlation by Ogboja and Kuye (19), and the Glitsch method is used for valve trays). The tray spacing is initially set to the default value (of 0.5m) and the downcomer area is calculated according to a method in the Glitsch manual (limited by a minimum time residence check). From the combined areas the column diameter is computed. The number of liquid passes on a tray is initially set by the column diameter; under 5ft one pass, under 8ft two, 10ft three, under 13ft four, else five passes. With the number of passes and the column diameter the total weir length is computed. Once the weir length is determined the liquid weir load is checked, if too high the number of passes is incremented and a new weir length is evaluated until the weir load is below a specified maximum.

Initial weir height is taken as 2", but limited to a maximum of 15 % of the tray spacing. For notched or serrated weirs the notch depth is a third of the weir height. For serrated weirs the angle of serration is 45° . Circular weirs have diameters 0.9 times the weir length. The hole diameter is set to 3/16" for sieve trays and tray thickness 0.43 times the hole diameter (or 1/10"). The hole pitch is computed from the free area ratio and hole diameter according to a triangular pitch. The default downcomer clearance is 1.5" but is limited by the maximum allowed downcomer velocity according to the Glitch method, de-rated with the system factor. The clearance is set to be at least 0.5" lower than the weir height to maintain a positive liquid seal but is limited to a minimum of 0.5".

For bubble-cap trays the cap diameter is 3" for column diameters below 4.5 *ft* and 4" for above. The hole diameter can vary between 60% to 71% of the cap diameter, and default taken as 70%. Default skirt clearance is 1" with minimum of 0.5" and maximum of 1.5". The slot height can vary in between 0.5" and 1.5", where the default is 1" for cap diameters below 3.5" and 1.25" for larger cap diameters. The pitch can vary from 1.25" to half the flow path length (minimum number of rows is two), with the default value set to 1.25".

Valve trays are initialized to be Venturi orifice uncaged, carbon steel valves of 3*mm* thick with 3 legs (see Kister, 1992, p312). The hole diameter is 1" for column smaller than 4.5*ft*, otherwise 2". No double weight valves are present.

The second task in the fraction of flooding method consists of finding the proper free area ratio ($\beta = A_h/A_b = \text{hole area} / \text{active area}$) so that no weeping occurs. This ratio can vary between a minimum of 5% (for stable operation) and a maximum of 20%. To test whether weeping occurs, we use the correlation of Lockett and Banik (1984): $Fr_{hole} > 2/3$. The method requires all liquid heights to be evaluated at weep rate conditions. This task is ignored for bubble-cap trays. The weep test is done at weeping conditions, with a weep factor at 60 % (this can be changed). Calculation of the liquid heights was described in Section 16.2. If weeping occurs at the lower bound for the free area ratio, a flag is set for the final task to adapt the design.

The final task consists of evaluating all liquid heights at normal conditions and to do a number of checks:

- vapor distribution (for bubble-cap trays),
- weeping (for sieve trays/valve trays),
- hydraulic flooding,
- excessive liquid entrainment,

- froth height limit, and
- excessive pressure drop

If a check fails, the design is modified to correct the problem, according to the adjustments shown in Table 15.4 after which new areas are calculated from the capacity correlations. Part of this task is also to keep the layout parameters within certain bounds to maintain a proper tray design. Finally, the number of iterations for the design method is checked against a maximum (default 30) to prevent a continuous loop.

The adjustment factors f_1 , f_2 , and f_3 are percentile in/decrements, normally set at 5, 2, and 1 %. These factors – together with all the default, lower, and upper settings that are used in the design routine – are stored in a “design file” (TDESIGN.DEF, described in Section ??) that can be tailored to handle specific kinds of designs and columns. This allows the selection of different methods for capacity and hydrodynamic calculations as well. Also the fraction that the flows need to change before a re-design is issued can be changed in this manner together with other design criteria. The design file must be in the current directory for the nonequilibrium program to use it, otherwise the normal defaults will be used. To obtain the best results, it would be best to have different TDESIGN.DEF files for distillation and absorption. However, a compromise between these operations was struck and *ChemSep* currently has one file to handle both.

15.5.2 Tray Column Design: Pressure drop

The same design routine for a tray design at a specific fraction of flood may also be used to design a tray with a certain *maximum* pressure drop, using a default fraction of flood of 75%. Note that this method of designing trays does not fix the pressure drop: it only applies a maximum *allowed* pressure drop over the tray. If the tray design results in a pressure drop larger than that specified the layout is adjusted according to the steps for excessive pressure drop per Table 15.4. To obtain a layout which has a lower pressure drop is to raise the bubbling area, in effect lowering the fraction of flood. However, the weir height is also lowered and the hole diameter is increased (albeit with a smaller factor).

15.6 Comparison with Experimental Distillation Data

Here, we illustrate the performance of the nonequilibrium model by simulating several systems for which experimental data is available in the literature. We focus our

Table 15.4: Tray design checks and adjustments

Problem	Test	Adjustments
Bubble cap vapor distribution	$h_{lg}/h_d > 0.5$	$p + f_1$, $h_{skirt} + f_2$, $h_{slot} + f_3$, $d_h - f_3$
Weeping	$Fr_h/(2/3) < 1 - f_a$ $f_{ree} < 0.05$ $A_b < A_{bf}$: else:	$A_b = A_{bf}$ $W_f + f_1$ $A_b - f_1$ $d_h - f_3$ $h_w - f_3$ $t_v + f_2$ (vt)
Hydrodynamic (downcomer) flooding)	$T_s < h_{db}/FF$	$T_s + f_1$ $A_d + f_1$ $h_w + f_2$ $h_c + f_3$
Excessive liquid entrainment		$A_b + f_1$ $T_s + f_1$ $d_h - f_2$ $h_w - f_3$
Froth height limit	$h_f > 0.75T_s$	$A_b + f_1$ $T_s + f_2$ $h_w - f_3$
Excessive pressure drop	$g\rho h_{wt} > \Delta p_{max}$	$A_b + f_1$ $h_w - f_1$ $d_h + f_2$ $p + f_1$ (bc) $h_{skirt} + f_2$ (bc) $h_{slot} + f_3$ (bc)
Excessive vapor entrainment		$A_d + f_1$

attention on ternary systems; only in mixtures with more than two species can the molecular interaction between them influence the mass transfer resulting in different individual component Murphree efficiencies.

The complete process of comparing experimental data with the current version of *ChemSep* has been automated and provides a test for the correct behavior of the sim-

ulator. We hope to build up an extensive database of distillation experiments and welcome additional data. The experimental literature data is entered in a database file that consists of keywords and the data. Then we carry out simulations and store each simulation in a sep-file. With the use of some utility programs we convert the simulation information and compare it to the experimental measured values. We can create parity plots and calculate average and maximum errors.

15.6.1 Ternary Distillation Experiments

The experiments discussed below were carried out at total reflux in columns with bubble-cap trays.

Vogelpohl (1979) has reported some results for the distillation of two non-ideal systems: acetone, methanol, and water as well as methanol, isopropanol, and water. The experiments were done in a column with 38 bubble-cap trays of 0.3 m in diameter. Due to the ease of separation, only up to 13 trays were active for the experimental runs. The experimental data clearly shows that the component Murphree efficiencies are unequal; indeed, in the acetone-methanol-water system the composition of methanol passes through a maximum in the column and the efficiency for this component becomes unbounded. Vogelpohl shows that the assumption of equal component efficiencies gives rise to large differences between the predicted and measured composition profiles.

The simulations were done using the total reflux mode of the nonequilibrium model. For total reflux specifications one must use a column configuration with a condenser and reboiler plus one feed. Either the distillate flow is set to zero and a reboiled vapor flow is specified, or, the bottoms flow is set to zero and a reflux flowrate is specified. Only these specifications of the column operation will trigger the total reflux mode of the column simulator. In this mode the feed specifications are employed as the specification of the vapor or liquid compositions on a specific stage. The compositions (and stage) that are to be fixed are set by the feed compositions and stage. Thus, if we know the compositions of the vapor leaving the reboiler we specify a feed to the reboiler and set the feed component flows equal to the known mole fractions. To indicate that these are vapor mole fractions I set the vapor fraction of the feed as 1. On the other hand, if the liquid reboiler compositions are known we would specify the feed vapor fraction as 0.

For Vogelpohl's data the compositions of the vapor leaving the (total) reboiler were specified to match the measured values. The reboiler vapor flowrate was specified at $0.65 \text{ kmol}/h$ resulting in an effective F-factor around 0.1. The calculated fraction of flood was around 20 %. Table 15.5 shows the UNIQUAC interaction parameters used

for the acetone, methanol, and water column. These interaction parameters were fitted to binary VLE data. The bubble-cap trays were specified with 0.3 *m* tray diameter, tray spacing as 0.2 *m*, bubbling area of 0.06008 *m*², a weir height of 0.03 *m*, and a flow path width of 0.24 *m*. The other layout parameters are assigned/computed by the simulator. The AIChE MTC model was used together with the vapor plug flow and liquid mixed flow models.

Table 15.5: UNIQUAC interaction parameters (cal/mol) for the Acetone (1) - Methanol (2) - Water (3) system

Components i-j	A_{ij}	A_{ji}
acetone - methanol	403.8524	-84.2364
acetone - water	698.7989	-110.382
methanol - water	-337.129	549.2958

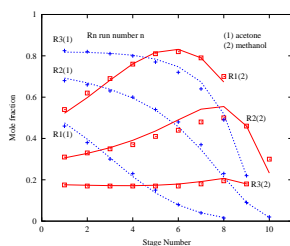


Figure 15.1: Experimental compositions (points) and the predicted composition profiles (lines) for the acetone-methanol-water system.

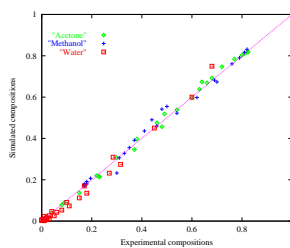


Figure 15.2: Parity plot for the acetone-methanol-water system.

For the methanol, isopropanol, and water column UNIQUAC interaction parameters were taken from DECHEMA (p. 575) that were fitted to ternary VLE data, see Ta-

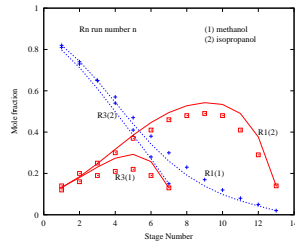


Figure 15.3: Experimental compositions (points) and the predicted composition profiles (lines) for the methanol-isopropanol-water system.

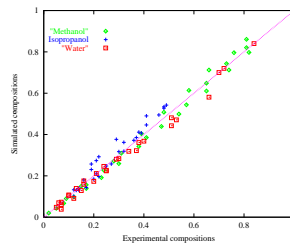


Figure 15.4: Parity plot for the methanol-isopropanol-water system.

ble 15.6. The same bubble-cap tray layout, column specifications, and model selections were used as for the acetone-methanol-water column.

Figure 15.1 shows the measured compositions and the simulated mole fraction profiles for the acetone-methanol-water system. Figure 15.2 shows that the nonequilibrium model does an excellent job in predicting the mole fractions for this very nonideal system. Figure 15.3 and Figure 15.4 show that the methanol-isopropanol-water system is predicted less well. We suspect this may be due to the interaction parameters of the activity coefficients. These simulations are extremely sensitive to these parameters, especially as isopropanol is going through a maximum in concentration. The average and maximum discrepancies in the predicted and experimentally measured compositions are shown in Table 15.8.

Ternary distillation experiments using acetone, methanol, and ethanol were performed by Free and Hutchison (1960) in a column with 7 bubble-cap trays of 0.1016 m in diameter. They also find that equal Murphree efficiencies cannot explain the behavior of this system. Twelve runs were conducted covering different regions of the ternary composition triangle. The column they used had a tray diameter of 0.1016 m, bub-

Table 15.6: UNIQUAC interaction parameters (cal/mol) for the Methanol (1) - Isopropanol (2) - Water (3) system

Components i-j	A_{ij}	A_{ji}
methanol - isopropanol	754.216	-457.107
methanol - water	670.563	-442.713
isopropanol - water	355.536	28.874

bling area of 0.00689 m^2 , 0.015 m weir height and a flow path length of 0.0813 m . The F-factor was assumed to be around $0.9 \sqrt{\text{kg/m}^3/\text{s}}$ and a vapor boilup of 0.65 kmol/h was specified, resulting in fraction of flood around 35 %. Table 15.7 shows the UNIQUAC Q prime interaction parameters were used (Prausnitz et al., 1980). The UNIQUAC Q prime activity coefficient model gives excellent results for these experiments. Figure 15.5 shows the data and the predicted column profiles for some of the experimental runs. The agreement is excellent, as can also be seen from the parity plot, Figure 15.6.

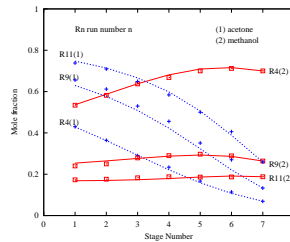


Figure 15.5: Comparison of experimental compositions (points) and the predicted composition profiles (lines) for the acetone-methanol-ethanol system.

Table 15.7: UNIQUAC interaction parameters (cal/mol) for the Acetone (1) - Methanol (2) - Ethanol (3) system

Components i-j	A_{ij}	A_{ji}
acetone - methanol	359.10	-96.90
acetone - ethanol	404.49	-131.25
methanol - ethanol	660.19	-292.39

All these simulations were done by specifying the bottom compositions in the column, just as was done by Krishnamurthy and Taylor (1985). The errors in Table 15.8 would

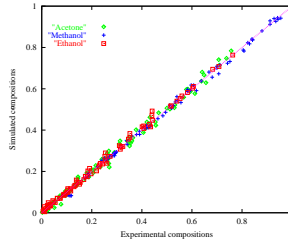


Figure 15.6: Comparison of experimental and predicted mole fractions for the acetone-methanol-ethanol system.

be reduced if we would specify the compositions in the middle of the column, as the errors in the predicted mole fractions often accumulate from stage to stage.

Table 15.8: Summary of the average and maximum discrepancies between model prediction and experimental measurement

System	No. of Runs	No. of Samples	Average Error	Maximum Error
acetone - methanol - water	3	22	0.015	0.070
methanol - isopropanol - water	3	27	0.024	0.087
acetone - methanol - ethanol	12	67	0.010	0.051

15.6.2 Significance of Multicomponent Interaction Effects

Figure 15.7 is a comparison for the acetone-methanol-water experiments of the nonequilibrium model with an ‘equal diffusivity’ model. This model uses a single average value of the diffusion coefficients for each component. It yields equal component stage efficiencies and, therefore, corresponds to the conventional equilibrium stage model with efficiencies. The comparison clearly shows that the predicted top compositions of the ‘equal diffusivity’ model are qualitatively different from the experimental values.

This can be explained by inspecting the back-calculated Murphree component efficiencies of the nonequilibrium model, see Figure 15.8. We see that the efficiencies of water and acetone differ by about 10%! This explains the observed 10% difference in top compositions. The methanol efficiencies show an even more weird behavior: they become *unbounded* between plate 6 and 7. When we look at the compositions, we see that the reason for this is that methanol goes through a maximum around stage

6. Whenever a component goes through an extreme in composition (the driving force becomes zero) and there is still mass transfer occurring (however little) than the efficiencies are unbounded. If we had more stages in the column, we would observe that the methanol efficiency below this maximum stays at the higher value of the Murphree efficiency of water. Apparently, the *direction* of the mass transfer is important!

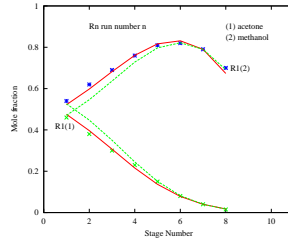


Figure 15.7: Experimental compositions (points) and the predicted composition profiles for the acetone-methanol-water system using the nonequilibrium model (solid lines) and the ‘equal diffusivity’ model (broken lines).

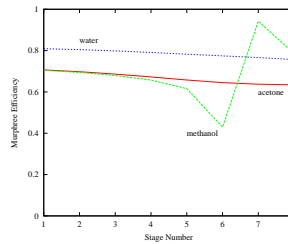


Figure 15.8: Back-calculated Murphree component efficiencies for the acetone-methanol-water system from the nonequilibrium simulation.

15.6.3 Binary Distillation Experiments: Mass Transfer Coefficients and Flow Models

To illustrate the behavior of the various tray MTC models as well as the different flow models we include here some comparisons with experimental data of the FRI by Yanagi and Sakata (1979, 1981). Two systems were used in these tests: the cyclohexane - *n*-heptane system at pressures of 28, 34, and 165 *kPa*, and the *i*-butane - *n*-butane system at pressures of 1138, 2056, and 2756 *kPa*. The experiments were carried out in sieve tray columns operated at total reflux. Nonequilibrium simulations

Table 15.9: Sieve Plate Dimensions of FRI Column

Column diameter (m)	1.2
Tray spacing (m)	0.61
Sieve plate material	316SS
Plate thickness (mm)	1.5
Hole diameter (mm)	12.7
Hole pitch (mm)	30.2
Weir length (m)	0.94
Weir height (mm)	25.4,50.8
Downcomer clearance (mm)	22,38
Effective bubbling area (m^2)	0.859
Hole area (m^2)	0.118

were done for a column with the same tray design (the design parameters are summarized in Table 15.9) at total reflux (Kooijman, 1995). The Murphree efficiencies were calculated for each component on each tray from the results of a simulation and averaged (over those trays not adjacent to condenser/reboiler). Simulations were done using different combinations of flow models:

- Mixed vapor - Mixed liquid
- Plug flow vapor - Mixed liquid
- Plug flow vapor - Plug flow liquid

Simulations were carried out with flows that go from 20% to 100% of flooding. Figure 15.9 shows some of these results with the Chan and Fair (1984) mass transfer coefficients correlations. The experimental efficiencies show a decline at low and high fractions of flooding, probably due to weeping and liquid entrainment. The Chan and Fair model includes a quadratic dependence of N^V on the fraction of flooding in order to account for the decrease in mass transfer at both low and high fractions of flooding. For this reason the Chan and Fair method usually describes the mass transfer (and hence, the efficiencies) better than the other mass transfer coefficient models.

Note that the Mixed-Mixed flow model underpredicts the efficiencies, as is true for the Plug flow vapor - Mixed flow liquid model. The Plug-Plug flow model fits the experimental efficiencies quite well. This should not be too surprising as this data was actually used in the development of the Chan and Fair correlations. We are, however, using the correlations in a nonequilibrium model rather than in an efficiency calculation.

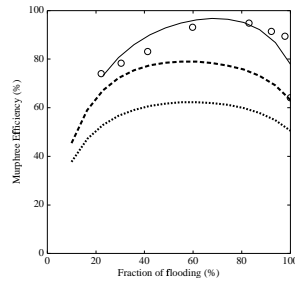


Figure 15.9: Murphree efficiencies for different flow models for a (8% hole area) sieve tray column with the cyclohexane - *n*-heptane system operating at 165 *kPa*. Mixed-Mixed flow (thick dotted line), Plug flow vapor - Mixed flow Liquid (thick dashed line), and Plug-Plug flow (solid line). Mass transfer coefficients from the Chan and Fair correlation..

Four different methods for estimating the binary mass transfer coefficients for sieve trays were tested: AIChE (1958), Chan and Fair (1984), Zuiderweg (1982), and Chen and Chuang (1994). Figure 15.10 shows the results for the *i*-butane/*n*-butane tests using the plug-plug flow model. In general, the AIChE and the Chan-Fair correlations behave in similarly, except for the strong dependence on the fraction of flooding of the Chan-Fair model. This is not surprising since both use the same expression for the liquid mass transfer coefficients. The Zuiderweg and Chen-Chuang models predict higher efficiencies as they have higher values for the liquid number of transfer units than the first two correlations. For this test they perform well; in many other tests these two models tend to overpredict the Murphree efficiencies.

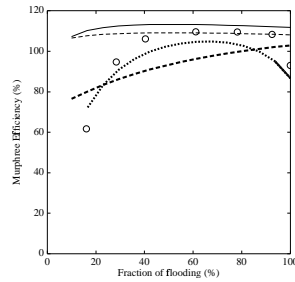


Figure 15.10: Murphree efficiencies for different Mass Transfer Coefficient models for a (14% hole area) sieve tray column with *i*-butane - *n*-butane system operating at 1138 *kPa*. Chan and Fair (thick dotted line), AIChE (thick dashed line), Zuiderweg (dashed line), and Chen and Chuang (solid line)..

The Chan-Fair correlation tends to describe the overall behavior of the Murphree efficiencies better than the other methods considered. However, its formulation causes it always to have a maximum efficiency at 60% fraction of flood. This limits the model to sieve trays only and to the range of fractions of flooding where the quadratic term is positive (0-1.2). Presumably, the fall-off in tray performance at low and high fractions of flooding is due to increases in weeping or entrainment at these extreme flows. It is not clear to us that mass transfer coefficient correlations should account for these effects. Rather, we suggest that these effects should be separated. We have also encountered situations where the Chan and Fair correlations provide negative mass transfer coefficients because the flows are outside its range. Not only are negative mass transfer coefficients physically meaningless, they may prevent the program that implements our nonequilibrium model from converging to a solution! Despite these problems with the Chan and Fair method we think its limitations are less serious (from our perspective) than are the limitations of other methods and, for now, it is our method of choice.

Symbol List

Latin Symbols

a_d	interfacial area density (m^2/m^3)
a	area (m^2)
A_h	hole area (m^2)
A_b, A_{bub}	bubbling area (m^2)
A_d	downcomer area (m^2)
B	packing base (m)
c	number of components, molar concentration ($kmol/m^3$)
Ca	Capillary number
d_h	hole diameter (m)
d_{eq}	equivalent diameter
D	binary diffusivity coefficient (m^2/s)
D_c	column diameter (m)
D_e	eddy dispersion coefficient (m^2/s)
f_1, f_2, f_3	design adjustment factors
F_p	packing factor ($1/m$)
F_s	F factor $F_s = U_v \sqrt{\rho_V} (kg^{0.5}/m^{0.5}/s)$
FF	fraction of flooding
FP	flow parameter $FP = M_L/M_V \sqrt{\rho_t^V/\rho_t^L}$

Fr	Froude number
g	gravitational constant, $9.81 (m/s^2)$
h_c	clearance height under downcomer (m)
h_{cl}	clear liquid height (m)
h_d	dry tray pressure drop height (m)
h_{db}	downcomer backup liquid height (m)
h_f	froth height (m)
h_i	liquid height at tray inlet (m)
h_{lg}	liquid gradient pressure drop height (m)
h_l, h_L	liquid pressure drop height (m)
h_{ow}	height of liquid over weir (m)
h_r	residual pressure drop liquid height (m)
h_{wt}	wet tray pressure drop liquid height (m)
h_w	weir height (m)
h_{udc}	liquid height pressure loss under downcomer (m)
k	binary mass transfer coefficient (m/s)
Le	Lewis number ($Le = Sc/Pr$)
M_w	molecular weight ($kg/kmol$)
N	number of transfer units, NTU
P	perimeter (m)
p	hole pitch (m), pressure (Pa)
Δp	pressure drop (Pa)
ΔP_{max}	maximum design pressure drop ($Pa/tray$ or Pa/m)
Pr	Prandtl number
Q	volumetric flow (m^3/s)
Re	Reynolds number
S	packing side (m)
Sc	Schmidt number
SF	system derating factor
t	residence time (s)
t_v	valve thickness (m)
T	temperature (K)
T_s	tray spacing (m)
u, U	velocity (m/s)
V	vapor flow rate ($kmol/s$)
We	Weber number
W_l	weir length (m)
x	liquid mole fraction
y	vapor mole fraction
z	mole fraction

Z tray flow path length (m)

Greek Symbols

α fraction liquid in froth
 β fractional free area $\beta = A_h/A_b$,
 ϵ void fraction
 Γ liquid flow per perimeter
 ϕ fractional entrainment
 ρ density (kg/m^3)
 σ surface tension (N/m)
 η viscosity ($Pa \cdot s$)
 λ thermal conductivity ($W/m/K$)

Superscripts

I interface
 L liquid
 P phase P
 V vapor

Subscripts

b, bub bubbling
 c critical, contact
 eff effective
 $fl, flood$ at flooding conditions
 i component i
 j stage j ,
component j
 h hole
 MW minimum wetting
 p packing
 $spec$ specified

t total

Abbreviations

bc bubble-caps

References

Bubble Tray Design Manual: Prediction of Fractionation Efficiency, AIChE, New York (1958).

P.E. Barker, M.F. Self, "The evaluation of Liquid Mixing Effects on a Sieve Plate using Unsteady and Steady-State Tracer Techniques", *Chem. Eng. Sci.*, Vol. **17**, (1962) p. 541.

S.D. Barnicki, J.F. Davis, "Designing Sieve-Tray Columns, Part 1: Tray Design", *Chem. Engng.*, Vol. **96**, No. 10 (1989) pp. 140–146.

S.D. Barnicki, J.F. Davis, "Designing Sieve-Tray Columns, Part 2: Column Design and Verification", *Chem. Engng.*, November (1989) pp. 202–212.

D.L. Bennett, R. Agrawal, P.J. Cook, "New Pressure Drop Correlation for Sieve Tray Distillation Columns", *AIChE J.*, Vol. **29** (1983) pp. 434–442.

R. Billet, M. Schultes, "Advantage in correlating packed column performance", *ICHEME Symp. Ser.* No. 128 (1992) p. B129.

J.L. Bravo, J.R. Fair, "Generalized Correlation for Mass Transfer in Packed Distillation Columns", *Ind. Eng. Chem. Process Des. Dev.*, Vol. **21** (1982) pp. 162–170.

J.L. Bravo, J.A. Rocha, J.R. Fair, "Mass Transfer in Gauze Packings", *Hydrocarbon Processing*, January (1985) pp. 91–95.

J.L. Bravo, J.A. Rocha, J.R. Fair, "Pressure Drop in Structured Packings", *Hydrocarbon Processing*, March (1986) pp. 45–48.

J.L. Bravo, J.A. Rocha, J.R. Fair, "A Comprehensive Model for the Performance of Columns Containing Structured Packings", *I. ChemE. Symp. Ser.*, No. 128 (1992) pp.

A439–A457.

H. Chan, J.R. Fair, “Prediction of Point Efficiencies on Sieve Trays. 1. Binary Systems”, *Ind. Eng. Chem. Process Des. Dev.*, Vol. bf 23 (1984) pp. 814–819.

G.X. Chen and K.T. Chuang, “Prediction of Efficiencies for Sieve Trays in Distillation”, *Ind. Eng. Chem. Res.*, Vol. 32 (1993) pp. 701–708.

I.J. Harris, “Optimum Design of Sieve Trays”, *Brit. Chem. Engng.*, Vol. 10, No. 6 (1965) p. 377.

G.A. Hughmark, “Models for Vapour Phase and Liquid Phase Mass Transfer on Distillation Trays”, *AIChE J.*, Vol. 17, No. 6 (1971) p. 1295.

F. Kastanek, “Efficiencies of Different Types of Distillation Plate”, *Coll. Czech. Chem. Comm.*, Vol. 35 (1970) p. 1170.

H.Z. Kister, *Distillation Design*, McGraw-Hill, New York (1992).

G.F. Klein, *Chem. Engng.*, May 3 (1982) p. 81.

M. Leva, *Tower Packings and Packed Tower Design*, The U.S. Stoneware Company (1951).

M. Leva, “Reconsider Packed-Tower Pressure-Drop correlations”, *Chem. Eng. Prog.* January (1992) p. 65.

M.J. Lockett, *Distillation Tray Fundamentals*, Cambridge University Press (1986).

M.J. Lockett, S. Banik, “Weeping from Sieve Trays”, *Ind. Eng. Chem. Process Des. Dev.*, Vol. bf 25 (1986) p. 561.

E.E. Ludwig, *Applied Process Design for Chemical and Petrochemical Plants*, Vol. 2, 2nd Ed., Gulf Pub. Co., Houston, TX, (1979).

O. Ogboja, A. Kuye, “A Procedure for the Design and Optimisation of Sieve Trays”, *Trans. I. Chem. E.*, Vol. 68, Part A (1990) pp. 445-452.

K. Onda, H. Takeuchi, Y. Okumoto, “Mass Transfer Coefficients Between Gas and Liquid Phases in Packed Columns”, *J. Chem. Eng. Jap.*, Vol. 1, No. 1 (1968) p. 56.

M. Prado, J.R. Fair, “Fundamental Model for the Prediction of Sieve Tray Efficiency”, *Ind. Eng. Chem. Res.*, Vol. 29, No. 6 (1990) pp. 1031–1042.

M. Prado, PhD Thesis, Austin, TX (1986).

R.H. Perry and D. Green, *Perry's Chemical Engineering Handbook*, 6th edition, section 18, Liquid-Gas System (1984) pp. 18-8 – 18-12.

B.D. Smith, *Design of Equilibrium Staged Processes*, McGraw-Hill, New York (1963).

J. Stichlmair, A. Mersmann, "Dimensioning Plate Columns for Absorption and Rectification", *Chem. Ing. Tech.*, Vol. **45**, No. 5 (1978) p. 242.

J. Stichlmair, J.L. Bravo, J.R. Fair, *Gas. Sep. Purif.*, Vol. **3**, (1989) p. 19.

R. Taylor, R. Krishna, *Multicomponent Mass Transfer*, Wiley NY (1993)

P.C. Wankat, *Separations in Chemical Engineering - Equilibrium Staged Separations*, Elsevier (1988) pp. 420–428.

F.J. Zuiderweg, "Sieve Trays - A View of the State of the Art", *Chem. Eng. Sci.*, Vol. **37** (1982) pp. 1441–1461.

Chapter 16

Design Models for Packed Columns

This chapter serves as a guide to the correlations in *ChemSep* for estimating mass transfer coefficients and the pressure drop in packed distillation (and absorption) columns. The methods used in *Design Mode* to determine column design parameters also are discussed in detail. The chapter ends with the comparison of experimental distillation data and nonequilibrium model simulations.

16.1 Mass Transfer Coefficient Correlations

Table 16.1 provides a summary of the available correlations for trays and packings; the various correlations are discussed in more detail below. Recommended models are shown in boldface.

16.1.1 Random Packings

OTO-68 Onda *et al.* (1968) [parameters a_p , d_p , σ_c] developed correlations of mass transfer coefficients for gas absorption, desorption, and vaporization in random packings. The vapor phase mass transfer coefficient is obtained from

$$k^V = A \text{Re}_V^{0.7} \text{Sc}_V^{0.333} (a_p D^V) (a_p d_p)^{-2} \quad (16.1)$$

Table 16.1: Available mass transfer coefficient correlations per internals type

Dumped packing	Structured packing
Onda 68	Bravo 85
Bravo 82	Bravo 92
Billet 92	Billet 92
...	...

where $A = 2$ if $d_p < 0.012$ and $A = 5.23$ otherwise. Vapor and liquid velocities are calculated from

$$u_V = VM_w^V / \rho^V A_t \quad (16.2)$$

$$u_L = LM_w^L / \rho^L A_t \quad (16.3)$$

and Reynolds and Schmidt numbers defined by:

$$\text{Re}_V = \frac{\rho^V u_V}{(\eta^V a_p)} \quad (16.4)$$

$$\text{Re}_L = \frac{\rho^L u_L}{(\eta^L a_p)} \quad (16.5)$$

$$\text{Sc}_V = \frac{\eta^V}{(\rho_t^V D^V)} \quad (16.6)$$

$$\text{Sc}_L = \frac{\eta^L}{(\rho_t^L D^L)} \quad (16.7)$$

The liquid phase mass transfer coefficient is

$$k^L = 0.0051(\text{Re}_L^I)^{2/3} \text{Sc}_L^{-0.5} (a_p d_p)^{0.4} (\eta^L g / \rho^L)^{1/3} \quad (16.8)$$

where Re_L^I is the liquid Reynolds number based on the interfacial area density

$$\text{Re}_L^I = \frac{\rho^L u_L}{(\eta^L a_d)} \quad (16.9)$$

The interfacial area density, a_d (m^2/m^3), is computed from

$$a_d = a_p (1 - \exp(-1.45(\sigma_c/\sigma)^{0.75} \text{Re}_L^{0.1} \text{Fr}_L^{-0.05} \text{We}_L^{0.2})) \quad (16.10)$$

where

$$\text{Fr}_L = \frac{a_p u_L^2}{g} \quad (16.11)$$

$$\text{We}_L = \frac{\rho^L u_L^2}{a_p \sigma} \quad (16.12)$$

BF-82 Bravo and Fair (1982) [parameters a_p , d_p , σ_c] used the correlations of Onda *et al.* for the estimation of mass transfer coefficients for distillation in random packings but proposed an alternative relation for the interfacial area density:

$$a_d = 19.78(\text{Ca}_L \text{Re}_V)^{0.392} \sqrt{\sigma} H^{-0.4} a_p \quad (16.13)$$

where H is the height of the packed section and Ca_L is the capillary number

$$\text{Ca}_L = u_L \eta^L / \rho^L \sigma \quad (16.14)$$

Since the interfacial area density is used in the calculation for the liquid phase Reynolds number the Bravo and Fair method will predict different mass transfer coefficients for the liquid phase.

BS-92 Billet and Schultes (1992) [parameters a_p , ϵ , C_{fl} , C_h , C_p , C_v , C_l] describe an advanced empirical/theoretical model which is dependent on the pressure drop/holdup calculation (C_h , C_p , C_{fl}). The correlation can be used for both random and structured packings. Vapor and liquid phase coefficients are adjusted by parameters C_v and C_l , bringing the total number of parameters to five. There are trends in the parameters that can be observed from the tabulated data. Unfortunately, no such generalization was done by Billet, making use of the model dependent on the availability of the parameters or experimental data. The mass transfer coefficients are computed from

$$k^L = C_l \left(\frac{g \rho_l}{\eta_l} \right)^{1/6} \sqrt{\frac{D^L}{d_h}} \left(\frac{u_L}{a_p} \right)^{1/3} \quad (16.15)$$

$$k^V = C_v \left(\frac{1}{\sqrt{\epsilon - h_t}} \right) \sqrt{\frac{a}{d_h}} D^V (\text{Re}_V)^{3/4} (\text{Sc}_V)^{1/3} \quad (16.16)$$

with Reynolds and Schmidt numbers calculated as in the method of Onda *et al.*. The hydraulic diameter d_h is

$$d_h = 4\epsilon/a_p \quad (16.17)$$

and the liquid holdup fraction, h_t , is calculated as described below under the pressure drop section. The interfacial area density is given by:

$$a_d = a_p (1.5/\sqrt{a_p d_h}) (u_L d_h \rho^L / \eta^L)^{-0.2} (u_L^2 \rho^L d_h / \sigma)^{0.75} (u_L^2 / g d_h)^{-0.45} \quad (16.18)$$

16.1.2 Structured Packings

BRF-85 Bravo *et al.* (1985) [parameters a_p , ϵ , B , h_c , S , D_{eq} , θ] published correlations for structured packings. This method is based on the assumption that the

surface is completely wetted and that the interfacial area density is equal to the specific packing surface: $a_d = a_p$. The Sherwood number for the vapor phase is

$$\text{Sh}_V = 0.0338 \text{Re}_V^{0.8} \text{Sc}_V^{0.333} \quad (16.19)$$

and is defined by

$$\text{Sh}_V = \frac{k^V d_{eq}}{D^V} \quad (16.20)$$

The equivalent diameter of a channel is given by

$$d_{eq} = Bh_c [1/(B + 2S) + 1/2S] \quad (16.21)$$

where B is the base of the triangle (channel base), S is the corrugation spacing (channel side), and h_c is the height of the triangle (crimp height). The vapor phase Reynolds number is defined by

$$\text{Re}_V = \frac{d_{eq} \rho^V (u_{V,eff} + u_{L,eff})}{\eta^V} \quad (16.22)$$

The effective velocity of vapor through the channel, u_{Ve} , is

$$u_{V,eff} = u_V / (\epsilon \sin \theta) \quad (16.23)$$

(u_V is the superficial vapor velocity, ϵ the void fraction, and θ the angle of the channel with respect to the horizontal). The effective velocity of the liquid is

$$u_{L,eff} = \frac{3\Gamma}{2\rho^L} \left(\frac{(\rho^L)^2 g}{3\eta^L \Gamma} \right)^{1/3} \quad (16.24)$$

where Γ is the liquid flow rate per unit of perimeter

$$\Gamma = \rho^L u_L / P \quad (16.25)$$

where P is the perimeter per unit cross-sectional area, computed from

$$P = (4S + B) / Bh_c \quad (16.26)$$

The penetration model is used to predict the liquid phase mass transfer coefficients with the exposure time assumed to be the time required for the liquid to flow between corrugations (a distance equal to the channel side):

$$t_L = S / u_{L,eff} \quad (16.27)$$

$$k^L = 2 \sqrt{\frac{D^L}{\pi t_L}} \quad (16.28)$$

BRF-92 Bravo *et al.* (1992) [parameters a_p , ϵ , S , θ , F_{se} , K_2 , C_e , $dPdz_{flood}$] developed a theoretical model for modern structured packings. Four parameters can be supplied. However, the authors advise using a fixed value for the surface renewal correction (C_e), normally 0.9. They provide a relation for parameter K_2 as well:

$$K_2 = 0.614 + 71.35S \quad (16.29)$$

The mass transfer calculations depend on the pressure drop and holdup calculation. The effective area can be adjusted with the surface enhancement factor F_{se} , and the liquid resistance with a correction on the surface renewal following the penetration model (parameter C_e). Effective velocities are computed with

$$u_{L,eff} = \frac{u_L}{\epsilon h_t \sin \theta} \quad (16.30)$$

$$u_{G,eff} = \frac{u_V}{\epsilon(1 - h_t) \sin \theta} \quad (16.31)$$

where h_t is the fractional liquid holdup (see below at the section on pressure drop calculation). Reynolds numbers and liquid mass transfer coefficient is now calculated as in Bravo *et al.* (1985) but with

$$t_L = C_e S / u_{L,eff} \quad (16.32)$$

The vapor phase mass transfer coefficient is obtained from

$$k^V = 0.054 \left(\frac{D^V}{S} \right) \text{Re}_V^{0.8} \text{Sc}_V^{1/3} \quad (16.33)$$

where the equivalent diameter is replaced with the channel side S and a different coefficient is used. The assumption of a completely wetted packing is dropped. Instead, the interfacial area density is given by

$$a_d = F_t F_{se} a_p \quad (16.34)$$

$$F_t = \frac{29.12 (\text{We}_L \text{Fr}_L)^{0.15} S^{0.359}}{\text{Re}_L^{0.2} \epsilon^{0.6} (\sin \theta)^{0.3} (1 - 0.93 \cos \gamma)} \quad (16.35)$$

where $\cos \gamma$ is equal to 0.9 for $\sigma < 0.0453$, otherwise it is computed from

$$\cos \gamma = 5.211 10^{-16.835\sigma} \quad (16.36)$$

Note that a switch point different from that used by Bravo *et al.* (1992) is employed to guarantee continuity in $\cos \gamma$.

BS-92 Billet and Schultes (1992) [parameters a_p , ϵ , C_{fl} , C_h , C_p , C_v , C_l] developed a model for both random and structured packings, see the section on random packings above.

Table 16.2: Pressure drop correlations per internals type

Dumped packing	Structured packing
Fixed	Fixed
Ludwig 79	Billet 92
Leva 92	Bravo 86
Billet 92	Stichlmair 89
Stichlmair 89	Bravo 92
...	...

16.2 Pressure Drop Models

There are as many ways to compute tray pressure drops as there to estimate mass transfer coefficients. For packings there is a move away from the use of generalized pressure drop charts (GPDC) to more theoretically based correlations. We have chosen to employ the most recently published models. For packings there are at least 7 methods available (see Table 16.2). For packings operating above the loading point ($FF > 0.7$) we advise the use of models that take the correction for the liquid holdup into account, such as the SBF-89. However, disadvantage is that these models can have complex (imaginary) solutions, especially at high fractions of flood. This can cause non-convergence! The Lev-92 model can be an alternative for it includes a dependence on the liquid flow rate to simulate the increased pressure drop at loading conditions. When the pressure drop is specified as fixed, it is assumed zero.

16.2.1 Random packing Pressure Drop Correlations

For packings the vapor and liquid mass flow per cross sectional area (kg/m^2s) and velocities (m/s) are:

$$L_a = LM^L/A_t \tag{16.37}$$

$$V_a = VM^V/A_t \tag{16.38}$$

$$u_L = L_a/\rho^L \tag{16.39}$$

$$u_V = V_a/\rho^V \tag{16.40}$$

Lud-79 Ludwig (1979) [parameters A, B] supplied a simple empirical equation for

the pressure drop requiring two fitted parameters (see Wankat, 1988, 420–428):

$$\frac{\Delta p}{\Delta z} = 3.281\ 242A \frac{(0.2048V_a)^2}{(0.06243\rho^V)} 10^{B(0.06243L_a)} \quad (16.41)$$

where 3.281 242, 0.2048, and 0.06243 are conversion factors so that we can use A and B parameters from Wankat. Its accuracy is limited since the influence of physical properties such as viscosity or surface tension on A and B are not included. Even more, the fitted parameters can be flow regime dependent. The loading regime is not well described with the simple exponential term.

Lev-92 Leva (1992) [parameter F_p] devised a modified version of the Generalized Pressure Drop Correlation (GPDC) presented long ago by Leva (1951). The GPDC has been the standard design method for decades. Some modifications that were actually simplifications made the GPDC lose its popularity. The function worked back from the GPDC and limiting ($L_a = 0$) behavior is (in SI units):

$$\frac{\Delta p}{\Delta z} = 22.3F_p(\eta^L)^{0.2}\phi V_a^2 \frac{10^{0.035L_a\phi}}{g\rho^V} \quad (16.42)$$

with $\phi = \rho_{water}/\rho_l = 1000/\rho_l$. This expression is similar to the Ludwig (1979) equation with corrections for the influence of the liquid density and viscosity. The only parameter is the packing factor F_p that can be obtained from dry pressure drop experiments (see Leva, 1992) or estimated from the specific packing area over the void fraction cubed. Again, the loading regime is not well described with the simple exponential term. This model is the default method for random packings if no model is specified, since it requires only the packing factor.

SBF-89 Stichlmair *et al.* (1989) [parameters a_p , ϵ , C_1 , C_2 , C_3] published a semi-empirical method from an analogy of the friction of a bed of particles and the pressure drop. It contains a correction for the actual void fraction corrected for the holdup, that is dependent on the pressure drop. It is, therefore, an iterative method. It is suitable for both random and structured packings, but there are few published parameters for structured packings. The pressure drop is

$$\frac{\Delta p}{\Delta z} = 0.75f_0(1 - \epsilon_p)\rho^V * U_V^2 / (d_p\epsilon_p^{4.65}) \quad (16.43)$$

where the void fraction of the irrigated bed, equivalent packing diameter, Reynolds number, and friction factor for a single particle are:

$$\epsilon_p = \epsilon - h_t \quad (16.44)$$

$$d_p = 6(1 - \epsilon_p)a_p \quad (16.45)$$

$$\text{Re}_V = u_V d_p \rho^V / \eta^V \quad (16.46)$$

$$f_0 = C_1/\text{Re}_V + C_2/\sqrt{\text{Re}_V} + C_3 \quad (16.47)$$

The iterations are started by assuming a dry bed for which $\epsilon_p = \epsilon$ and the holdup fraction is computed with the liquid Froude number:

$$\text{Fr}_L = u_L^2 a_p / g \epsilon^{4.65} \quad (16.48)$$

$$h_t = 0.555 \text{Fr}_L^{1/3} \quad (16.49)$$

The liquid holdup is limited to 0.5 in order to handle flooding.

BS-92 Billet and Schultes (1992) and Billet's monograph (1979) [parameters a , ϵ , C_{fl} , C_h , C_p] include a model with a comprehensive list of packing data and fitted parameters. The method corrects for the holdup change in the loading regime but employs an empirical exponential term, and is not iterative.

Packing dimension, hydraulic diameter and F-factor are

$$d_p = 6(1 - \epsilon)/a_p \quad (16.50)$$

$$d_h = 4\epsilon/a_p \quad (16.51)$$

$$F_s = u_V \sqrt{\rho^V} \quad (16.52)$$

Liquid Reynolds and Froude number are

$$\text{Re}_L = u_L \rho_L / \eta^L a_p \quad (16.53)$$

$$\text{Fr}_L = u_L^2 a_p / g \quad (16.54)$$

If $\text{Re}_L < 5$ then

$$q = C_h \text{Re}_L^{0.15} \text{Fr}_L^{0.1} \quad (16.55)$$

else

$$q = 0.85 C_h \text{Re}_L^{0.25} \text{Fr}_L^{0.1} \quad (16.56)$$

$$h_{l,1} = \left(\frac{12 \eta^L a_p^2 u_L}{\rho_L g} \right)^{1/3} \quad (16.57)$$

$$h_{l,2} = h_{l,1} q^{2/3} \quad (16.58)$$

$$h_{l,fl} = 0.3741 \epsilon \left(\frac{\eta^L \rho_w}{\eta_w \rho_L} \right)^{0.05} \quad (16.59)$$

$$\epsilon_{fl} = \left(\frac{u_L}{u_V} \right) \sqrt{\frac{\rho^L}{\rho^V}} \left(\frac{\eta^L}{\eta^V} \right)^{0.2} \quad (16.60)$$

$$\epsilon_{fl} = g / (C_{fl}^2 \epsilon_{fl}^{-0.39}) \quad (16.61)$$

$$u_{v,fl} = \sqrt{2g / \epsilon_{fl} (\epsilon - h_{l,fl})^{1.5}} \sqrt{h_{l,fl} / a_p} \sqrt{\rho^L / \rho^V} / \sqrt{\epsilon} \quad (16.62)$$

if $u_V > u_{V,fl}$ then $h_t = h_{l,fl}$ else

$$h_t = h_{l,2} + (h_{l,fl} - h_{l,2})(u_V/u_{V,fl})^{13} \quad (16.63)$$

The pressure drop is then

$$K_1 = 1 + (2/3)(1/1 - \epsilon)(d_p/D_c) \quad (16.64)$$

$$\text{Re}_V = u_V d_p \rho^V / (1 - \epsilon) \eta^V K_1 \quad (16.65)$$

$$\phi_{l1} = C_p (64/\text{Re}_V + 1.8/\text{Re}_V^{0.08}) \exp(\text{Re}_L/200)(h_t/h_{l,1})^{0.3} \quad (16.66)$$

$$\frac{\Delta p}{\Delta z} = \phi_{l1} (a_p / (\epsilon - h_t))^3 (F_s^2/2) K_1 \quad (16.67)$$

16.2.2 Structured packing Pressure Drop Correlations

BRF-86 Bravo *et al.* [parameters ϵ , S , $\sin(\theta)$, C_3] compute the pressure drop from an empirical correlation with one fitted parameter, called C_3 . This model is unsuitable for pressure drop correlations in the loading regime ($FF > 0.7$). The pressure drop per height of packing is:

$$\frac{\Delta p}{\Delta z} = (0.171 + 92.7/\text{Re}_V)(\rho^V u_{V,eff}^2/d_{eq}) \left(\frac{1}{(1 - C_3 \sqrt{\text{Fr}})} \right)^5 \quad (16.68)$$

where

$$u_{V,eff} = u_V / (\epsilon \sin \theta) \quad (16.69)$$

$$\text{Re}_V = \frac{d_{eq} \rho^V u_{V,eff}}{\eta^V} \quad (16.70)$$

$$\text{Fr}_L = u_L^2 / d_{eq} g \quad (16.71)$$

SBF-92 Stichlmair *et al.* (1989) [parameters a , ϵ , C_1 , C_2 , C_3] published a semi-empirical method, see the section on pressure drop of random packed columns above.

BRF-92 Bravo *et al.* (1992) [parameters a_p , ϵ , S , θ , K_2 , $dPdz_{flood}$] developed a theoretical model developed for modern structured packings. Two parameters need to be supplied for pressure drop calculations, however, the K_2 parameter was fitted by the authors. The pressure of flooding ($dPdz_{flood}$) can be easily obtained from data or via Kister's correlation and the packing factor. The model includes an iterative method with a dependence of the liquid holdup on the pressure drop (and vice versa). The Weber, Froude, and Reynolds numbers are

$$\text{We}_L = u_L^2 \rho^L S / \sigma \quad (16.72)$$

$$\text{Fr}_L = u_L^2 / (Sg) \quad (16.73)$$

$$\text{Re}_L = u_L S \rho^L / \eta^L \quad (16.74)$$

The effective g (as a function of h_t) is obtained from:

$$g_{eff} = \left(1 - \frac{dPdZ}{dPdZ_{flood}}\right) \frac{(\rho^L - \rho^V)}{\rho^L} g \quad (16.75)$$

Then F_t (see above), h_t , and $dPdZ$ are computed

$$h_t = \left(\frac{4F_t}{S}\right)^{2/3} \left(\frac{3\eta^L u_L}{\rho^V \sin \theta \epsilon g_{eff}}\right)^{1/3} \quad (16.76)$$

$$A = \frac{0.177\rho^V}{S\epsilon^2(\sin \theta)^2} \quad (16.77)$$

$$B = \frac{88.774\eta^V}{S^2\epsilon \sin \theta} \quad (16.78)$$

$$\frac{\Delta p}{\Delta z} = (Au_V^2 + Bu_V) \left(\frac{1}{1 - K_2 h_t}\right)^5 \quad (16.79)$$

The calculation is repeated until pressure drop converges or when it becomes larger than the pressure drop at flood. There can be problems converging this method.

BS-92 Billet and Schultes (1992) and Billet's monograph (1979?) [a_p , ϵ , C_{fl} , C_h , C_p] See the section on pressure drop of random packed columns above.

16.3 Entrainment and Weeping

Entrainment can occur in packings and cause a rapid loss of efficiency due to back-mixing but currently is not accounted for. Weeping does not occur in packed columns.

16.4 Packing Flooding and Minimum Wetting

The fraction of flooding for packings is computed by dividing the superficial gas velocity by the gas velocity at flood. The latter is found in an iterative process from the pressure drop correlation (keeping the liquid to vapor ratio constant!) and the specified pressure drop of flood. If no flood pressure drop is specified it is computed from the packing factor with the Kister and Gill correlation. If no pressure drop model is selected, the Leva's method is used, which also has the packing factor as only parameter.

The minimum operating conditions, similar to the weep point for tray operation, is set to the minimum wetting rate predicted by Schmidt (1979). Schmidt calculates a liquid falling number

$$C_L = 1633.6(0.062428\rho_L)(1000\sigma)^3/(1000\eta_L)^4 \quad (16.80)$$

and the minimum wetting rate is (in $USgal/min/ft^2$):

$$Q_{MW} = \frac{0.3182C_L^{2/9}(1 - \cos \phi)^{2/3}}{\sqrt{(1 - T_L)a_p/0.3048}} \quad (16.81)$$

where ϕ is the liquid contact angle and T_L is the shear stress number:

$$T_L = 0.9FF^{-2.8} \quad (16.82)$$

with FF as the fraction of flood. The contact angle for metals is 10° (0.1745 rad). To obtain contact angles for other packings the contact angle is assumed inverse proportional to the critical surface tension, σ_c . For metals we have $\sigma_c = 0.075$, so the contact angle is computed with

$$\phi = 0.1745 \left(\frac{0.075}{\sigma_c} \right) \quad (16.83)$$

With this approach we obtain higher minimum wetting rates for plastic packings than for metal packings. The minimum operating or 'weep' fraction WF is then taken as

$$WF = \frac{Q_{MW}}{u_L} \quad (16.84)$$

where u_L is the superficial liquid flow.

16.5 Column Design

For packings only the column diameter and bed height are design parameters, other parameters (such as void fraction, nominal packing diameter, etc.) are fixed once the type of packing has been selected. The packed bed height must be specified since it determines the desired separation and the capacity.

16.5.1 Packed Column Design: Fraction of flooding

For packed columns only the column diameter is to be estimated. Default packing data are used for all parameters that are not specified; values of 1" metal Pall rings

for random packed sections and of Koch Flexipack 2 (316SS) for structured sections. To determine the packed column diameter, the diameter that gives rise to the flooding pressure drop (as specified) is computed using the selected pressure drop model. The resulting diameter is corrected for the fraction of flooding and the system factor:

$$D_c = \frac{D_{c,flood}}{\sqrt{FF SF}} \quad (16.85)$$

This makes the resulting column diameter depend on the selected pressure drop model. If no pressure drop model is selected the Leva (1992) model is selected (which is a function only of the packing factor). If no pressure drop at flood is specified, it is estimated with Kister and Gill correlation (1992):

$$\Delta P_{flood} = 0.115 F_p^{0.7} \quad (16.86)$$

(with pressure drop as liquid height and English units!). This correlation is a function only of the packing factor but has been tested on a wide range of packings and an accuracy of 15 %. As long as the packing factor is known, this design method will not fail.

16.5.2 Packed Column Design: Pressure drop

Tray design based on a specified pressure drop is done as discussed above but with a default fraction of flooding of 75 %. However, the specified pressure drop functions as a maximum allowed pressure drop per tray. No adjustment is done if the pressure drop is below this specified pressure drop.

Packed column design automatically finds the diameter leading to the specified pressure drop (with the selected pressure drop model). This is done by using a linear search technique as the different packing pressure drop correlations can behave quite erratically. The maximum allowed pressure drop is the flooding pressure drop as specified or computed from Kister's correlation and the packing factor. If the pressure drop is specified to be very low the column diameter might converge to unrealistic diameters. A zero or larger than flooding pressure drop specification results in a 70 % fraction of flooding design.

Symbol List

Latin Symbols

a_d	interfacial area density (m^2/m^3)
a	area (m^2)
A_h	hole area (m^2)
A_b, A_{bub}	bubbling area (m^2)
A_d	downcomer area (m^2)
B	packing base (m)
c	number of components, molar concentration ($kmol/m^3$)
Ca	Capillary number
d_h	hole diameter (m)
d_{eq}	equivalent diameter
D	binary diffusivity coefficient (m^2/s)
D_c	column diameter (m)
D_e	eddy dispersion coefficient (m^2/s)
f_1, f_2, f_3	design adjustment factors
F_p	packing factor ($1/m$)
F_s	F factor $F_s = U_v \sqrt{\rho_V}$ ($kg^{0.5}/m^{0.5}/s$)
FF	fraction of flooding
FP	flow parameter $FP = M_L/M_V \sqrt{\rho_V^V/\rho_t^L}$
Fr	Froude number
g	gravitational constant, 9.81 (m/s^2)
h_c	clearance height under downcomer (m)
h_{cl}	clear liquid height (m)
h_d	dry tray pressure drop height (m)
h_{db}	downcomer backup liquid height (m)
h_f	froth height (m)
h_i	liquid height at tray inlet (m)
h_{lg}	liquid gradient pressure drop height (m)
h_l, h_L	liquid pressure drop height (m)
h_{ow}	height of liquid over weir (m)
h_r	residual pressure drop liquid height (m)
h_{wt}	wet tray pressure drop liquid height (m)
h_w	weir height (m)
h_{udc}	liquid height pressure loss under downcomer (m)
k	binary mass transfer coefficient (m/s)
Le	Lewis number ($Le = Sc/Pr$)

M_w	molecular weight ($kg/kmol$)
N	number of transfer units, NTU
P	perimeter (m)
p	hole pitch (m), pressure (Pa)
Δp	pressure drop (Pa)
ΔP_{max}	maximum design pressure drop ($Pa/tray$ or Pa/m)
Pr	Prandtl number
Q	volumetric flow (m^3/s)
Re	Reynolds number
S	packing side (m)
Sc	Schmidt number
SF	system derating factor
t	residence time (s)
t_v	valve thickness (m)
T	temperature (K)
T_s	tray spacing (m)
u, U	velocity (m/s)
V	vapor flow rate ($kmol/s$)
We	Weber number
W_l	weir length (m)
x	liquid mole fraction
y	vapor mole fraction
z	mole fraction
Z	tray flow path length (m)

Greek Symbols

α	fraction liquid in froth
β	fractional free area $\beta = A_n/A_b$,
ϵ	void fraction
Γ	liquid flow per perimeter
ϕ	fractional entrainment
ρ	density (kg/m^3)
σ	surface tension (N/m)
η	viscosity ($Pa \cdot s$)
λ	thermal conductivity ($W/m/K$)

Superscripts

<i>I</i>	interface
<i>L</i>	liquid
<i>P</i>	phase <i>P</i>
<i>V</i>	vapor

Subscripts

<i>b, bub</i>	bubbling
<i>c</i>	critical, contact
<i>eff</i>	effective
<i>fl, flood</i>	at flooding conditions
<i>i</i>	component <i>i</i>
<i>j</i>	stage <i>j</i> , component <i>j</i>
<i>h</i>	hole
<i>MW</i>	minimum wetting
<i>p</i>	packing
<i>spec</i>	specified
<i>t</i>	total

Abbreviations

bc	bubble-caps
----	-------------

References

Bubble Tray Design Manual: Prediction of Fractionation Efficiency, AIChE, New York (1958).

P.E. Barker, M.F. Self, "The evaluation of Liquid Mixing Effects on a Sieve Plate using Unsteady and Steady-State Tracer Techniques", *Chem. Eng. Sci.*, Vol. **17**, (1962) p. 541.

- S.D. Barnicki, J.F. Davis, "Designing Sieve-Tray Columns, Part 1: Tray Design", *Chem. Engng.*, Vol. **96**, No. 10 (1989) pp. 140–146.
- S.D. Barnicki, J.F. Davis, "Designing Sieve-Tray Columns, Part 2: Column Design and Verification", *Chem. Engng.*, November (1989) pp. 202–212.
- D.L. Bennett, R. Agrawal, P.J. Cook, "New Pressure Drop Correlation for Sieve Tray Distillation Columns", *AIChE J.*, Vol. **29** (1983) pp. 434–442.
- R. Billet, M. Schultes, "Advantage in correlating packed column performance", *ICHEME Symp. Ser.* No. 128 (1992) p. B129.
- J.L. Bravo, J.R. Fair, "Generalized Correlation for Mass Transfer in Packed Distillation Columns", *Ind. Eng. Chem. Process Des. Dev.*, Vol. **21** (1982) pp. 162–170.
- J.L. Bravo, J.A. Rocha, J.R. Fair, "Mass Transfer in Gauze Packings", *Hydrocarbon Processing*, January (1985) pp. 91–95.
- J.L. Bravo, J.A. Rocha, J.R. Fair, "Pressure Drop in Structured Packings", *Hydrocarbon Processing*, March (1986) pp. 45–48.
- J.L. Bravo, J.A. Rocha, J.R. Fair, "A Comprehensive Model for the Performance of Columns Containing Structured Packings", *I. ChemE. Symp. Ser.*, No. 128 (1992) pp. A439–A457.
- H. Chan, J.R. Fair, "Prediction of Point Efficiencies on Sieve Trays. 1. Binary Systems", *Ind. Eng. Chem. Process Des. Dev.*, Vol. **23** (1984) pp. 814–819.
- G.X. Chen and K.T. Chuang, "Prediction of Efficiencies for Sieve Trays in Distillation", *Ind. Eng. Chem. Res.*, Vol. **32** (1993) pp. 701–708.
- I.J. Harris, "Optimum Design of Sieve Trays", *Brit. Chem. Engng.*, Vol. **10**, No. 6 (1965) p. 377.
- G.A. Hughmark, "Models for Vapour Phase and Liquid Phase Mass Transfer on Distillation Trays", *AIChE J.*, Vol. **17**, No. 6 (1971) p. 1295.
- F. Kastanek, "Efficiencies of Different Types of Distillation Plate", *Coll. Czech. Chem. Comm.*, Vol. **35** (1970) p. 1170.
- H.Z. Kister, *Distillation Design*, McGraw-Hill, New York (1992).
- G.F. Klein, *Chem. Engng.*, May 3 (1982) p. 81.
- M. Leva, *Tower Packings and Packed Tower Design*, The U.S. Stoneware Company

(1951).

M. Leva, "Reconsider Packed-Tower Pressure-Drop correlations", *Chem. Eng. Prog.* January (1992) p. 65.

M.J. Lockett, *Distillation Tray Fundamentals*, Cambridge University Press (1986).

M.J. Lockett, S. Banik, "Weeping from Sieve Trays", *Ind. Eng. Chem. Process Des. Dev.*, Vol. 25 (1986) p. 561.

E.E. Ludwig, *Applied Process Design for Chemical and Petrochemical Plants*, Vol. 2, 2nd Ed., Gulf Pub. Co., Houston, TX, (1979).

O. Ogboja, A. Kuye, "A Procedure for the Design and Optimisation of Sieve Trays", *Trans. I. Chem. E.*, Vol. 68, Part A (1990) pp. 445-452.

K. Onda, H. Takeuchi, Y. Okumoto, "Mass Transfer Coefficients Between Gas and Liquid Phases in Packed Columns", *J. Chem. Eng. Jap.*, Vol. 1, No. 1 (1968) p. 56.

M. Prado, J.R. Fair, "Fundamental Model for the Prediction of Sieve Tray Efficiency", *Ind. Eng. Chem. Res.*, Vol. 29, No. 6 (1990) pp. 1031-1042.

M. Prado, PhD Thesis, Austin, TX (1986).

R.H. Perry and D. Green, *Perry's Chemical Engineering Handbook*, 6th edition, section 18, Liquid-Gas System (1984) pp. 18-8 - 18-12.

B.D. Smith, *Design of Equilibrium Staged Processes*, McGraw-Hill, New York (1963).

J. Stichlmair, A. Mersmann, "Dimensioning Plate Columns for Absorption and Rectification", *Chem. Ing. Tech.*, Vol. 45, No. 5 (1978) p. 242.

J. Stichlmair, J.L. Bravo, J.R. Fair, *Gas. Sep. Purif.*, Vol. 3, (1989) p. 19.

R. Taylor, R. Krishna, *Multicomponent Mass Transfer*, Wiley NY (1993)

P.C. Wankat, *Separations in Chemical Engineering - Equilibrium Staged Separations*, Elsevier (1988) pp. 420-428.

F.J. Zuiderweg, "Sieve Trays - A View of the State of the Art", *Chem. Eng. Sci.*, Vol. 37 (1982) pp. 1441-1461.

Chapter 17

Design Models for Extraction Columns

This chapter deals with the application of the nonequilibrium model to the modelling of extraction columns. In such operations the two phases present are both liquids instead of a liquid and a vapor as in the case of distillation, stripping, or absorption. This requires fundamentally different mass transfer coefficients and flow models, as well as a completely new design method.

17.1 Introduction

Nonequilibrium modelling of extraction uses essentially the same model as described in Chapter 14 (see Lao et al., 1989). In extraction, of course we have light and a heavy liquid phases, where the light liquid behaves as the vapor with, of course, liquid-like properties. If the heavy phase (L) is lighter than the light phase (V) the program stops. However, either phase (that is, L or V) can be the dispersed phase. The user must specify which is the disperse phase, since this changes the design and the calculation of MTC's. Currently sieve trays, structured and random packed columns, rotating disc contacters, and spray columns are supported as internals (as well as equilibrium stages with a specified stage efficiency). The K-values must be the liquid-liquid extraction model, which uses activity coefficients. The energy balance can be ignored (Enthalpy=None) or included. In case it is ignored the column temperature is dictated by that of the feeds, and linear interpolation is used to provide a column temperature

profile. A specific temperature profile can be imposed if the energy balance is ignored and the user initializes all of the temperatures. Default values for the total interfacial area and mass transfer coefficients are: $A_i = 100 \text{ m}^2$, $k_d = 10^{-5}$ and $k_c = 10^{-4} \text{ m/s}$. Mass transfer in coalescing layers and jet zones is neglected (they could be modeled by a special stage for packed/RDC columns). Thus, only the drop rise zone is taken into account for mass transfer. Current limitations of the nonequilibrium extraction model are:

- No efficiencies are back-calculated (yet),
- Limited number of mass transfer coefficient correlation,
- No comparisons of simulations with experiment have been performed,
- Backflow and dispersion calculations do not work properly (yet).

The nonequilibrium extraction work still requires more work. There is an abundance of correlations for extraction but no benchmarks have been done to indicate what are the right models.

17.2 Sieve Tray Columns

ChemSep will attempt to design the extraction column if no design is specified. This sieve tray design method is adapted from course notes by R. Krishna.

17.2.1 Design

The default free area ratio is 5%, tray spacing is 0.4 m , and the clearance under the downcomer a quarter of the tray spacing. There is no weir and the hole diameter is set by default to:

$$x = \sqrt{\frac{\sigma}{\Delta\rho g}} \quad (17.1)$$

$$d_h = 1.8x \quad (17.2)$$

but d_h is limited (if supplied) by:

$$0.5x < d_h < \pi x \quad (17.3)$$

and the practical limits (overriding):

$$3mm < d_h < 8mm \quad (17.4)$$

The hole velocity is computed with:

$$Eo = \frac{\Delta\rho g d_h^2}{\sigma} \quad (17.5)$$

$$We = 4.33 Eo^{-0.26} \quad (17.6)$$

$$U_h = \sqrt{\frac{We\sigma}{\rho_d d_h}} \quad (17.7)$$

If the hole velocity is less than 0.15 (m/s) then its design value is kept at 0.15 (m/s). The Froude number is computed from

$$Fr = \frac{U_h^2}{g d_h} \quad (17.8)$$

For Eo is less than 0.4 the Sauter mean droplet diameter is computed by:

$$d_p = Eo^{-0.4} \left(2.13 \left(\frac{\Delta\rho}{\rho_d} \right)^{0.67} + \exp(-0.13Fr) \right) d_h \quad (17.9)$$

otherwise

$$d_p = Eo^{-0.42} (1.24 + \exp(-Fr^{0.42})) d_h \quad (17.10)$$

The hole area is

$$A_h = \frac{Q_d}{U_h} \quad (17.11)$$

The ratio of the hole area over the active area (free area ratio, f) is limited to lie between 1 and 20 %.

$$A_a = A_h / f \quad (17.12)$$

The hole pitch can be computed if the hole diameter and free area ratio are known. The downcomer velocity can be computed if a minimum droplet diameter, d_{min} , is assumed that should not be entrained. This diameter is taken to be 0.5 mm and the downcomer velocity is the velocity of the continuous phase, U_c :

$$U_c = 0.249 d_{min} \left(\frac{(g\Delta\rho)^2}{\rho_c \eta_c} \right)^{0.33} \quad (17.13)$$

With U_c known we can compute the downcomer area:

$$A_d = Q_c / U_c \quad (17.14)$$

The total area is equal to two downcomer areas plus the active area and 0.5 % area for support etc.:

$$A_t = (A_a + 2A_d)/0.995 \quad (17.15)$$

With the total tray area known the column diameter can be computed. The net area for the disperse phase, A_n , and the disperse velocity, U_d , are:

$$A_n = A_A + A_d \quad (17.16)$$

$$U_d = \frac{Q_d}{A_n} \quad (17.17)$$

Next, the dispersed phase velocity holdup and slip velocity are computed. The slip velocity (V_s) is estimated to be one tenth of the dispersed phase velocity. This makes the dispersed phase holdup equal to one tenth since it is defined as

$$\phi_d = \frac{U_d}{V_s} \quad (17.18)$$

The slip velocity (a function of the dispersed phase holdup and needs to be obtained iteratively) can be calculated from:

$$V_s = \sqrt{2.725gd_p \left(\frac{\Delta\rho}{\rho_c} \right) \left(\frac{(1 - \phi_d)}{(1 + \phi_d^{0.33})} \right)^{1.834}} \quad (17.19)$$

After the dispersed phase holdup is computed (it depends on V_s), it is checked to ensure that it lies between 1 and 20 % for standard operational conditions. If it is too small the free area ratio is increased, if it is too large the free area ratio is decreased (each by 5 %) until it is within the desired range.

The Weber number

$$We = \rho_d U_h^2 d_p / \sigma \quad (17.20)$$

must be larger than 2 to ensure all holes produce drops (i.e. to avoid inactive holes, see Seibert and Fair, 1988).

The height of the coalesced layer is (Treybal, 1980):

$$h_c = \frac{(U_h^2 - U_d^2)\rho_d}{2gC_d^2\Delta\rho} + \frac{4.5U_c\rho_c}{2g\Delta\rho} + \frac{6\sigma}{d_p g\Delta\rho} \quad (17.21)$$

(with $C_d = 0.67$). The first term is height to overcome flow through the orrifices. The second term is for friction losses due to contraction/expansion on entry/exit (0.5 + 1.0 velocity heads) and change of direction (2 times 1.5 velocity heads). The third term is for the interfacial tension effects at the holes. The height needs to be larger than 4 cm (to ensure safe operation). If not, then the hole diameter is decreased by 5 % and we repeat the procedure from the hole velocity calculation, Eq. (17.5).

This design is for a one pass sieve tray, and the flow path length, L_f is computed from geometric relations. The weir length is (segmental downcomer):

$$W_l = A_a/L_f \quad (17.22)$$

The tray thickness is defaulted to a tenth of an inch. To prevent entrainment of droplets, the flow under the downcomer is only allowed to be 50 % higher than the downcomer velocity. If higher, then the downcomer clearance is enlarged until this requirement is met. The tray spacing is adjusted so that the coalesced layer and coalescence zone divided over the length of the downcomer equals the fraction of flooding (multiplied with the system factor).

The reported fraction of flooding is to the ratio of the height of the coalesced layer over the height of the downcomer (according to Seibert and Fair the flooding calculation is within 20 %). The lower operating limit is the ratio of two over the Weber number (to guarantee proper droplet formation).

17.2.2 Mass Transfer Coefficients

The 'Handlos-Baron-Treybal' method is used. The hole velocity U_h , Eu , Fr , net area A_n , Sauter mean drop diameter d_p , disperse velocity U_d , slip velocity V_s , disperse phase holdup ϕ_d , h_c , and h_z are computed as above (but with fixed design parameters). The interfacial area per unit of volume is

$$A_i = \frac{6\phi_d}{d_p} \quad (17.23)$$

and the drop rise zone (where mass transfer is assumed to take place):

$$h_{drop} = t_s - h_c \quad (17.24)$$

where t_s is the tray spacing. The volume for mass transfer on a tray is

$$V_i = A_n h_{drop} \quad (17.25)$$

The mass transfer coefficients for transport from the dispersed phase are (Handlos and Baron, 1957):

$$k_d = \frac{0.00375V_s}{(1 + \eta_d/\eta_c)} \quad (17.26)$$

and for transport from the continuous phase are (Treybal, 1963):

$$k_{c,ij} = 0.725\text{Re}_c^{-0.43}(1 - \phi_d)V_s\text{Nu}_c^{-0.58} \quad (17.27)$$

with

$$\text{Re}_c = d_p V_s \rho_c / \eta_c \quad (17.28)$$

$$\text{Nu}_c = \eta_c / \rho_c D_{ij}^c \quad (17.29)$$

Note that k_d is not a function of the diffusion coefficient and, thus, is the same for all components.

17.3 Packed Columns

Column design and calculation of mass transfer coefficients is done the same way for structured packings and random packings, following the methods and correlations as outlined by Seibert and Fair (1988).

17.3.1 Design

For mass transfer from the continuous phase to the dispersed phase we have $x = 1$ for the calculation of the Sauter mean drop diameter:

$$d_p = 1.15x \sqrt{\frac{\sigma}{\Delta \rho g}} \quad (17.30)$$

The slip velocity of a single droplet at zero disperse phase holdup is given by

$$V_s^o = \sqrt{\frac{4\Delta \rho g d_p}{3\rho_c C_d}} \quad (17.31)$$

where $C_d = 0.38$ (for high values of Reynolds). Static disperse holdup is:

$$\phi_{ds} = \frac{0.076 a_p d_p}{\xi} \quad (17.32)$$

where a_p is the packing area and ξ the packing void fraction. The static holdup area and total area are:

$$a_s = 60.076 a_p \quad (17.33)$$

$$a = a_p + a_s \quad (17.34)$$

The tortuosity is defined as

$$\zeta = \frac{a d_p}{2} \quad (17.35)$$

The superficial velocity of the continuous phase at the flood point is

$$e = \cos\left(\frac{\pi\zeta}{4}\right) \quad (17.36)$$

$$U_{cf} = \frac{0.192\xi V_s^o}{(1.08 + (Q_d/Q_c)/e^2)} \quad (17.37)$$

This needs to be corrected for the fraction of flooding (and system factor):

$$U_c = SF \times FF \times U_{cf} \quad (17.38)$$

to give the net area

$$A_n = \frac{Q_c}{U_c} \quad (17.39)$$

from which the packed column diameter (D_c) can be calculated.

The reported fraction of flooding is the quotient of computed U_c to U_{cf} as discussed above. The dispersion coefficients are given by (see Section 21 of Perry and Green, 1984):

$$\log \frac{E_d}{V_d d_p} = 0.046 \frac{V_c}{V_d} + 0.301 \quad (17.40)$$

$$\log \frac{E_c}{V_c d_p} = 0.161 \frac{V_c}{V_d} + 0.347 \quad (17.41)$$

where d_p is the packing diameter.

17.3.2 Mass Transfer Coefficients

The method of Seibert and Fair (1988) is used. The phase velocities are computed by

$$U_c = \frac{Q_c}{A_n} \quad (17.42)$$

$$U_d = \frac{Q_d}{A_n} \quad (17.43)$$

The drop diameter d_p , slip velocity V_s^o , area a , static holdup area a_s , and tortuosity ζ are calculated as above. Then the dispersed phase holdup, ϕ_d , is determined iteratively (starting at 0.1.) from:

$$f(\phi_d) = \exp\left(\frac{-6\phi_d}{\pi}\right) \quad (17.44)$$

$$\phi_d = \frac{U_d}{\xi(V_s^o f(\phi_d) - U_c)e^2} \quad (17.45)$$

Then the slip velocity is

$$V_s = V_s^o f(\phi_d)e + (1 - e)U_c \quad (17.46)$$

since $U_d = V_s^o f(\phi_d)$. The mass transfer coefficient for the dispersed phase is computed by:

$$\phi = \frac{\sqrt{Sc_d}}{(1 + \eta_d/\eta_c)} \quad (17.47)$$

$$\phi > 6 : k_{d,ij} = \frac{0.023V_s}{\sqrt{Sc_d}} \quad (17.48)$$

$$\phi < 6 : k_d = \frac{0.00375V_s}{(1 + \eta_d/\eta_c)} \quad (17.49)$$

$$Sc_d = \frac{\eta_d}{\rho_d D_{d,ij}} \quad (17.50)$$

$$(17.51)$$

If ϕ is larger than 6 the Laddha and Degaleesan correlation is used otherwise the Handlos-Baron method. For the mass transfer coefficient in the continuous phase:

$$Sh_c = 0.698 Re_c^{0.5} Sc_c^{0.4} (1 - \phi_d) \quad (17.52)$$

$$k_{c,ij} = \frac{Sh_c D_{c,ij}}{d_p} \quad (17.53)$$

$$Re_c = \frac{\rho_c V_s d_p}{\eta_c} \quad (17.54)$$

$$Sc_c = \frac{\eta_c}{\rho_c D_{c,ij}} \quad (17.55)$$

The interfacial area per unit volume is:

$$a_i = \frac{6\xi\phi_d}{d_p} \quad (17.56)$$

The total interfacial area in a stage is the stage height times the net area times the interfacial area per unit volume:

$$a_{i,tot} = a_i A_n h_{stage} \quad (17.57)$$

17.4 Rotating Disk Contactors

This design method is based on material in the Handbook of Solvent Extraction (Chapter 13.1) and course notes by R. Krishna.

17.4.1 Design

The phase ratio α is

$$\alpha = \frac{Q_d}{Q_c} \quad (17.58)$$

The maximum stable drop diameter is

$$u_0 = 0.9 \frac{(g\Delta\rho)^{5/21} \sigma^{6/21}}{\rho_c^{10/21} \rho_d^{1/21}} \quad (17.59)$$

$$d_{p,max} = \frac{\sigma}{\rho_c u_0^2} \quad (17.60)$$

A stable drop diameter is selected as half of the maximum diameter

$$d_p = 0.5d_{p,max} \quad (17.61)$$

and the require power input ($P_i = N^3 R^5 / HD^2$) is computed

$$e = \left(\frac{0.25 \left(\frac{\sigma}{\rho_c} \right)^{0.6}}{d_p} \right)^{2.5} \quad (17.62)$$

$$P_i = \frac{\pi e}{4C_p} \quad (17.63)$$

($C_p = 0.03$ for $Re > 10^5$). If no column diameter is known, an estimate is made from assuming a cross-sectional area for a combined velocity of 0.05 m/s with:

$$A_c = (Q_c + Q_d)/0.05 \quad (17.64)$$

The required rotational speed (using these standard ratios) is

$$N = \left(\frac{P_i \left(\frac{0.1}{0.6^5} \right)}{D_c^2} \right)^{0.33} \quad (17.65)$$

The slip velocity can be calculated using a correlation from Kung and Beckman (1961):

$$\frac{V_s \eta_c}{\sigma} = \left(\frac{\Delta\rho}{\rho_c} \right)^{0.9} \left(\frac{S}{R} \right)^{2.3} \left(\frac{H}{R} \right)^{0.9} \left(\frac{R}{D} \right)^{2.6} \left(\frac{g}{RN^2} \right) \quad (17.66)$$

The dispersed phase holdup at flood is determined from

$$\phi_d = \frac{\sqrt{\alpha^2 + 8\alpha} - 3\alpha}{4(1 - \alpha)} \quad (17.67)$$

from which the continuous phase velocity at flood can be determined with

$$U_{c,f} = V_s(1 - \phi_d)^2(1 - 2\phi_d) \quad (17.68)$$

Correction for fraction of flooding (and system factor) gives

$$U_c = SF \times FF \times U_{c,f} \quad (17.69)$$

from which the column area and diameter can be calculated

$$A_c = \frac{Q_c}{U_c} \quad (17.70)$$

The rotor diameter R , stator diameter S , and the height of the compartment have standard ratios with respect to the column diameter (D_c)

$$R = 0.6D_c \quad (17.71)$$

$$S = 0.7D_c \quad (17.72)$$

$$H = 0.1D_c \quad (17.73)$$

These ratios help determine the size of the column. Below a Reynolds number of 10^5 C_p becomes a function of the Reynolds number. Normally RDC's are operated in the regime above 10^5 so the Reynolds number is computed by

$$\text{Re}_d = \frac{\rho_d N R^2}{\eta_d} \quad (17.74)$$

and a smaller diameter is selected (and the calculations repeated) if necessary. On re-design the layout of the stage with the largest diameter is used for the entire section.

The reported fraction of flooding is the quotient of computed U_c over $U_{c,f}$ as discussed above. The operating velocity is proportional to the slip velocity and so inverse proportional to the square of the rotation speed. One of the design rules was to keep the dispersed phase Reynolds number larger than 10^5 so the lower operating limit is defined as:

$$\left(\frac{10^5}{\text{Re}_d} \right)^2 \quad (17.75)$$

Stemerding *et al.* (1963) gave a correlation for the axial dispersion coefficient for the continuous phase

$$\frac{E_c}{V_c H} = 0.5 + 0.012NR(S/D)^2/V_c \quad (17.76)$$

The dispersed phase dispersion coefficient is set to twice this number.

17.4.2 Mass Transfer Coefficients

The method of 'Kronig-Brink-Rowe' is used. Phase ratio α , energy input P_i (from N , R , H , and D_c) are computed as above. The drop diameter is computed from

$$C_p = 0.03 \quad (17.77)$$

$$e = \frac{4C_p P_i}{\pi} \quad (17.78)$$

$$d_p = \frac{0.25 (\sigma/e)^{0.6}}{\rho_c^{0.4}} \quad (17.79)$$

The dispersed phase holdup ϕ_d is calculated iteratively and the slip velocity is determined as described above (with 17.66). The mass transfer coefficients are:

$$\text{Sh}_d = 10.0 \quad (17.80)$$

$$k_{d,ij} = \frac{\text{Sh}_d D_{d,ij}}{d_p} \quad (17.81)$$

$$\text{Sh}_c = 2 + 0.42 \text{Re}_c^{0.62} \text{Sc}_c^{0.36} \quad (17.82)$$

$$k_{c,ij} = \frac{\text{Sh}_c D_{c,ij}}{d_p} \quad (17.83)$$

with

$$\text{Re}_c = \frac{\rho_c d_p^{1.33} e^{0.33}}{\eta_c} \quad (17.84)$$

$$\text{Sc}_c = \frac{\eta_c}{\rho_c D_{c,ij}} \quad (17.85)$$

The interfacial area per unit volume is

$$a_i = \frac{6\phi_d}{d_p} \quad (17.86)$$

Alternatively, the "Rose-Kintner-Garner-Tayeban" method can be used:

$$\text{Sh}_c = 0.6 \sqrt{\text{Re}_c} \sqrt{\text{Sc}_c} \quad (17.87)$$

$$b = d_p^{0.225} / 1.242 \quad (17.88)$$

$$\omega = \frac{8\sigma b n(n+1)(n-1)(n+2)}{d_p (n+1)\rho_d + n\rho_c} \quad (17.89)$$

$$k_d = 0.45 \sqrt{D_{d,ij} \omega} \quad (17.90)$$

where $n = 2$, and d_p is in *cm* for the calculation of b and ω .

17.5 Spray Columns

This design method is adapted from Jordan (1968) and Lo *et al.* (1983).

17.5.1 Design

The height of a stage in a spray column is set to the default value of 0.4 *m* and the hole diameter in the distributor to 0.005 *m*. The hole velocity (U_o) in the distributor is set to 0.1 *m/s* from which the total hole area is then:

$$A_o = Q_d/U_o \quad (17.91)$$

The droplet diameter can be calculated from (Vedaiyan *et al.*, 1972):

$$d_p = 1.592 \left(\frac{U_o^2}{2gd_o} \right)^{-0.067} \sqrt{\frac{\sigma}{g\Delta\rho}} \quad (17.92)$$

The flood velocity of the continuous phase is (Treybal, 1963):

$$U_{cf} = \frac{0.3894\Delta\rho^{0.28}}{[0.2165\eta_c^{0.075}\sqrt{\rho_c} + 0.2670d_p^{0.056}\sqrt{\rho_d\alpha}]^2} \quad (17.93)$$

where $\alpha = Q_d/Q_c$. The dispersed phase holdup at flood is

$$\phi_{df} = \frac{\sqrt{\alpha^2 + 8\alpha} - 3\alpha}{4(1 - \alpha)} \quad (17.94)$$

The velocity of the continuous phase is then

$$U_c = FF \times SF \times U_{cf} \quad (17.95)$$

and the column area

$$A_c = Q_c/U_c \quad (17.96)$$

from which the column diameter can be calculated. (The column area must also be larger than the total hole area, if not, the column area is set to four times the hole area.)

The fraction of flooding reported is calculated as

$$FF = \frac{U_c}{SF \times U_{cf}} \quad (17.97)$$

where U_{cf} is computed as in the spray column design and $U_c = A_c/Q_c$. No lower operating limit is calculated. The dispersion coefficient for the continuous phase is (see Section 21 of Perry and Green, 1984):

$$\frac{E_c}{V_c H} = 7.2\sqrt{U_d D_c} \quad (17.98)$$

Since the dispersion coefficient for the dispersed phase is unknown it is set equal to that for the continuous phase.

17.5.2 Mass Transfer Coefficients

The transition drop size below which droplets become stagnant is calculated from

$$P = \frac{\rho_c^2 \sigma^4}{g \eta_c^4 \Delta \rho} \quad (17.99)$$

$$d_{p,t} = 7.25 \sqrt{\frac{\sigma}{g \Delta \rho P^{0.15}}} \quad (17.100)$$

The drop terminal velocity is (Satish *et al.*, 1974):

$$V_t = 1.088 \left(\frac{U_o^2}{2g d_o} \right)^{-0.082} \left(\frac{\sigma g \Delta \rho}{\rho_c^2} \right)^{1/4} \quad (17.101)$$

With the continuous operating and flood velocities the fraction of flooding is calculated and then the dispersed phase holdup is

$$\phi_d = FF \phi_{df} \quad (17.102)$$

and the slip velocity

$$V_s = (1 - \phi_d) V_t \quad (17.103)$$

If the drops are stagnant ($d_p < d_{p,t}$) the dispersed phase MTC is computed from

$$k_{d,ij} = 18.9 D_{d,ij} / d_p \quad (17.104)$$

otherwise the Handlos-Baron correlation (1957) is used:

$$k_d = \frac{0.00375 V_s}{(1 + \eta_d / \eta_c)} \quad (17.105)$$

For the continuous phase MTC we use (see Section 21 of Perry and Green, 1984)

$$k_c = 0.725 \text{Re}_c^{-0.43} \text{Sc}_c^{-0.58} (1 - \phi_d) V_s \quad (17.106)$$

where

$$\text{Re}_c = d_p V_s \rho_c / \eta_c \quad (17.107)$$

$$\text{Sc}_c = \frac{\eta_c}{\rho_c D_{c,ij}} \quad (17.108)$$

The interfacial area for mass transfer per unit of volume is

$$A_i = \frac{6 \phi_d}{d_p} \quad (17.109)$$

17.6 Modeling Backflow

The backflows in an extraction column may be estimated from the dispersion coefficients with:

$$\alpha_d = \frac{E_d}{V_d H} - 0.5 \quad (17.110)$$

$$\alpha_c = \frac{E_c}{V_c H} - 0.5 \quad (17.111)$$

where α is the fractional backflow ("entrainment") in the stage, and H is the stage height.

For spray columns (see Section 21 of Perry and Green, 1984):

$$E_c = 7.2\sqrt{V_d D_c} \quad (17.112)$$

For packed columns:

$$\log \frac{E_d}{V_d d_F} = 0.046 \frac{V_c}{V_d} + 0.301 \quad (17.113)$$

$$\log \frac{E_c}{V_c d_F} = 0.161 \frac{V_c}{V_d} + 0.347 \quad (17.114)$$

For a RDC:

$$E_c = 0.5 H V_c + 0.012 R N H \left(\frac{S}{D_c} \right) \quad (17.115)$$

$$E_d = F E_c \quad (17.116)$$

where F is calculated from

$$F = \frac{4.210^5}{D_c^2} \left(\frac{V_d}{h} \right)^{3.3} \quad (17.117)$$

F must be greater than or equal to one. Krishna uses:

$$E_c = \frac{0.5 H U_c}{(1 - \Phi_d)} + 0.012 R N H \left(\frac{S}{D_c} \right) \quad (17.118)$$

$$E_d = \frac{0.5 H U_d}{\Phi_d} + 0.024 R N H \left(\frac{S}{D_c} \right) \quad (17.119)$$

Symbol List

Latin Symbols

a_p packing area per unit volume (m^2/m^3)

a_s	static holdup area per unit volume (m^2/m^3)
A_a	total tray active area (m^2)
A_d	downcomer area (m^2)
A_i	interfacial area per unit volume (m^2/m^3)
A_h	total tray hole area (m^2)
A_n	netto tray area (m^2), $A_n = A_a + A_d$
A_t	total tray area (m^2)
C_d	drag coefficient
D	binary diffusion coefficient (m^2/s)
D_c	column diameter (m)
d_e	effective drop diameter (m) ?
d_h	hole diameter (m)
d_{min}	minimum droplet diameter (m)
d_p	Sauter mean drop diameter (m)
E_o	Eotvos number ($\Delta\rho g d_h / \sigma$)
f	free area ratio (A_h/A_a)
F	molar flow ($kmol/s$)
Fr	Froude number ($U_h^2/g/d_h$)
FF	fraction of flooding
g, g_c	gravitational constant, 9.81 (m/s^2)
H	RDC compartment height (m)
h_c	height of coalesced layer (m)
h, h_{drop}	height of drop rising zone (m)
h_{stage}	stage height for packed column (m)
k	binary mass transfer coefficient (m/s)
M_w	molecular weight ($kg/kmol$)
N	rotation speed (rad/s)
Nu	Nusselt number
Pe	Peclet number
P_i	power input (?)
Q	volumetric flow (m^3/s)
R	rotor diameter (m)
Re	Reynolds number
S	inner stator diameter (m)
Sc	Schmidt number
Sh	Sherwood number
SF	system derating factor
t	contact time (s)
t_s	tray spacing (m)
U_c, U_d	continuous, dispersed phase velocity (m/s)
U_{cf}	continuous phase superficial velocity at flood (m/s)

U_h	hole diameter (m/s)
V_i	tray volume for interfacial mass transport (m^3)
V_s	slip velocity (m/s)
V_s^o	slip velocity at zero disperse phase holdup (m/s)
We	Weber number ($\rho_d U_h^2 d_p / \sigma$)
W_l	weir length (m)

Greek Symbols

α	phase ratio (Q_d/Q_c)
ρ	mass density (kg/m^3)
ϕ_d	disperse phase holdup fraction
ϕ_{ds}	static disperse phase holdup fraction
σ	interfacial tension (N/m)
η	liquid viscosity ($Pa.s$)
μ	kinematic viscosity (η/ρ)
ζ	tortuosity
κ	
δ	

Subscripts

c	continuous phase
d	disperse phase, downcomer
i	interface, component i
j	component j

References and Further Reading

F.H. Garner, M. Tayeban, *Anal. Real Soc. Espan. Fis. Quim. (Madrid)*, **B56** (1960) pp. 479.

- A.E. Handlos, T. Baron, "Mass and Heat Transfer from Drops in Liquid-Liquid Extraction", *AIChE J.*, Vol. **3** (1957) pp. 127–136.
- D.G. Jordan, *Chemical Process Development*, Part 2, John Wiley, New York (1968).
- R. Krishna, *Design of Liquid-Liquid Extraction Columns*, University of Amsterdam, NL (1993).
- R. Kronig, J.C. Brink, "The Theory of Extraction from Falling Droplets", *Appl. Sci Res.*, Vol. **A2** (1950) pp. 142–154.
- M. Lao, J. P. Kingsley, R. Krishnamurthy and R. Taylor, "A Nonequilibrium Stage Model of Multicomponent Separation Processes VI: Simulation of Liquid-Liquid Extraction", *Chem. Eng. Comm.*, Vol. **86** (1989) pp. 73–89.
- G.S. Laddha, T.E. Degaleesan, *Transport Phenomena in Liquid Extraction*, McGraw-Hill (1978).
- T.C. Lo, M.H.I. Baird, C. Hanson, *Handbook of Solvent Extraction*, John Wiley, NY (1983).
- R.H. Perry and D. Green, *Perry's Chemical Engineering Handbook*, 6th Ed. (1984).
- P.N. Rowe, K.T. Claxton, J.B. Lewis, *Trans. Inst. Chem. Eng.*, **43**, T14 (1965).
- P.M. Rose, R.C. Kinter, "Mass Transfer from Large Oscillating Drops", *AIChE J.*, Vol. **12**, No. 3 (1966) p. 530.
- E.Y. Kung, R.B. Beckman, "Dispersed-Phase Holdup in a Rotating Disk Extraction Column", *AIChE J.*, Vol. **7**, No. 2 (1961) pp. 319-324.
- L. Satish, T.E. Degaleesan, G.S. Laddha, *Indian Chem. Eng.*, Vol. **16** (1974) p. 36.
- A.F. Seibert, J.R. Fair, "Hydrodynamics and Mass Transfer in Spray and Packed Liquid-Liquid Extraction Columns", *Ind. Eng. Chem. Res.*, **27**, No.3 (1988) pp. 470–481.
- S. Stermerding, E.C. Lumb, J. Lips, "Axiale Vermischung in einer Drehscheiben-Extraktions Kolonne", *Chem. Ing. Tech.*, Vol. **35** (1963) pp. 844–850.
- R.E. Treybal, *Mass Transfer Operations*, 3rd ed., McGraw-Hill, New York (1980)
- R.E. Treybal, *Liquid Extraction*, 2nd ed., McGraw-Hill, New York (1963).
- S. Vedaiyan, T.E. Degaleesan, G.S. Laddha, H.E. Hoelscher, "Some Performance

Characteristics of Spray Columns”, *AIChE J.*, Vol. **18** (1972) pp. 161–168.

Chapter 18

Nonequilibrium Distillation Dynamics

