

Plotting McCabe-Thiele Diagrams in Microsoft Excel for Non-Ideal Systems

Dr. John L. Gossage, Lamar University

John L. Gossage is an Associate Professor in the Dan F. Smith Department of Chemical Engineering at Lamar University. His main research areas are simulation, applied probability, and engineering education. He currently teaches simulation and kinetics classes at both the undergraduate and graduate levels, as well as undergraduate advanced analysis. He holds a Ph.D. in chemical engineering from Illinois Institute of Technology in Chicago, IL.

Plotting McCabe-Thiele Diagrams in Microsoft Excel for Nonideal Systems

Abstract

McCabe-Thiele diagrams are an indispensable aid to teaching distillation, as they graphically demonstrate key concepts (number of stages, feed location, minimum reflux ratio, *etc.*) related to operability of a distillation column. Their drawback is that they are difficult to draw freehand, and this difficulty is aggravated when the vapor-liquid equilibrium data for the system is represented by a set of equilibrium data points, as then the necessary first step is to construct the equilibrium curve. In this paper, a method is presented to automate the process of fitting binary equilibrium data to a smooth curve (using a cubic B-spline algorithm implemented via Visual Basic for Applications, or VBA) and then drawing the McCabe-Thiele diagram in Microsoft Excel. In this way, the effect of changes to the operating conditions can be easily demonstrated. Furthermore, the method will locate the azeotrope if the system has one.

The goals of this paper are to provide instructors a quick, automated method of generating a McCabe-Thiele diagram for a nonideal binary system to facilitate classroom instruction, to aid students in learning about and manipulating these diagrams, and to demonstrate how to integrate VBA calculations (including the cubic B-splines) into an Excel worksheet.

Notation

<u>Variable</u>	<u>Definition</u>
a, b, c, d	Cubic equation coefficients
A	The (generally) more volatile component
Azeo.	Azeotrope
B_j	Section “j” of CBS
CBS	Cubic B-Splines
D	Distillate molar flowrate
DV	Dependent variable
η, η	Murphree efficiency
N	Number of stages
IV	Independent variable
NCS	“Normal” Cubic Splines
q	heat required to vaporize one mole of feed at the entering conditions divided by the molar latent heat of vaporization of the feed
R	Reflux molar flowrate
RR	Reflux ratio = D/R
u	Parametric variable: $0 \leq u \leq 1$
x	Liquid mole fraction of A
x_j, y_j	Equilibrium data point “j”
$x_j(u), y_j(u)$	Interpolated CBS value
XK_j, YK_j	Knot “j” of CBS
y	Vapor mole fraction of A
z	Feed mole fraction of A

Subscripts

b	Bottoms
d	Distillate
f	Feed
j	Counter

Introduction and Literature Survey

Although modern process simulation packages (such as Aspen Plus or Pro/II) can quickly and efficiently perform distillation column calculations, they often confuse students who are just beginning to learn distillation basics, leading to questions such as, “Why can’t the reflux ratio be smaller than the minimum reflux ratio?”, or “Why does increasing the product purity require a larger number of stages?”. Part of this confusion is due to the fact that these process simulation packages do not produce McCabe-Thiele diagrams, which are indispensable tools for teaching binary distillation as they graphically demonstrate the relationships among key parameters (number of stages, feed location, reflux ratio, tops and bottoms purity, *etc.*) related to the design and operation of a distillation column¹.

The drawback to McCabe-Thiele diagrams is that they are difficult to draw freehand, and this difficulty is aggravated when the vapor-liquid equilibrium data for the system is represented not by a constant relative volatility but rather by a set of equilibrium data points, as then the necessary first step is to fit an equilibrium curve to the data. If this fitting is performed freehand, two problems arise: first, different hands will produce different curves, which will naturally alter the computed results (sometimes quite significantly); and second, the process is cumbersome and tedious, thereby consuming valuable class time.

Naturally, several authors have made available via the internet Excel (and other) worksheets to perform McCabe-Thiele analyses and provide graphical output of the results. Most of these^{1, 2, 3} are restricted to ideal binary systems that display constant relative volatility. Some authors have the same restrictions, but provide their work in Matlab format⁴ or as HTML⁵. At least two papers^{6, 7} have provided Excel worksheets for nonideal systems. In both cases, the methods use linear interpolation of the experimental data as an approximation to “fill in the gaps” between the data points.

This paper presents a method to automate the process of fitting binary equilibrium data to a smooth curve (using a cubic B-spline algorithm implemented via Visual Basic for Applications (VBA) in Microsoft Excel) and then drawing the McCabe-Thiele diagram in Microsoft Excel. In this way, the effect of changes to the operating conditions can be quickly and easily demonstrated to students in class. Additionally, the cubic B-spline method presented in this paper can also be applied to smoothing any type of data where the endpoints are known. Moreover, this use of cubic B-splines to smooth and fit experimental equilibrium data has never been published.

Furthermore, if the binary system in question contains an azeotrope, the spreadsheet will locate it. If Excel’s Solver add-in is used, the method can also solve trial-and-error problems: for example, to determine the necessary reflux ratio for a fixed number of stages in the column.

In addition, this paper provides the VBA code to find real roots of any cubic equation: such a function can also be useful in other Excel applications.

The inputs to the spreadsheet are the x - y equilibrium data, the feed composition and “ q -value” (usually, the liquid mole fraction of the feed: formally defined as the heat required to vaporize one mole of feed at the entering conditions divided by the molar latent heat of vaporization of the feed⁸ (equation 11.4-12, page 710)), the desired tops and bottoms purity, the reflux ratio, and the Murphree efficiency. The outputs are the location of the azeotrope (if present), the intersection point of the feed line with the equilibrium curve, the required number of stages, the optimal feed stage location, and the McCabe-Thiele diagram for the system.

For a good overview of the McCabe-Thiele method as applied to binary distillation problems, see Geankoplis⁸ pages 706-724.

For a good introduction into the intricacies of Microsoft Excel 2007 (which is the version used in this paper) see *Excel 2007, The Missing Manual*⁹. While newer versions of Excel (2010, 2013, and 2016 thus far) are available, this paper uses 2007 because it is readily available and has the necessary features. The worksheet has been tried in Excel 2010 and 2013 with no difficulties.

Gerald and Wheatley in *Applied Numerical Analysis*¹⁰ discuss cubic B-splines (pgs. 184-188) as well as normal cubic splines (pgs. 170-175).

The remainder of this paper is divided into the following topics:

1. Discussion of cubic B-splines and their advantages over normal cubic splines.
2. Tutorial on writing user-defined functions in Visual Basic for Applications (VBA).
3. Steps required to create the Excel worksheet to display the McCabe-Thiele diagram.

Cubic B-splines

To illustrate the problem of fitting experimental data, consider the ethanol/water data¹¹ (at 1 atm) shown in Table 1 and plotted in Figure 1:

#	x	y	T (K)
1	0.0010	0.0047	373.15
2	0.0061	0.0721	371.75
3	0.0145	0.1539	369.65
4	0.0237	0.2301	367.35
5	0.0310	0.2851	365.65
6	0.0490	0.3559	363.45
7	0.0652	0.4181	361.45
8	0.0968	0.4534	359.55
9	0.1394	0.5314	357.55
10	0.3261	0.6047	354.65
11	0.4635	0.6518	353.35
12	0.5413	0.6751	352.65

13	0.6856	0.7451	351.75
14	0.7760	0.8005	351.55
15	0.8403	0.8457	351.35
16	0.9037	0.9010	351.45
17	0.9725	0.9721	351.55
18	0.9804	0.9774	351.65

Table 1: Ethanol-Water Equilibrium Data¹¹

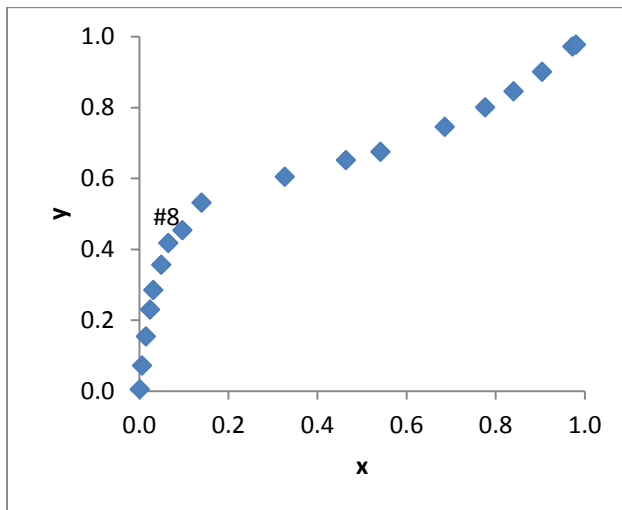


Figure 1: Equilibrium data for Ethanol-Water

As shown in Figure 1, the data appear to follow a smooth curve, except that data point #8 seems to be out of alignment: either too low or too far to the right. If the equilibrium curve is drawn using “normal” cubic splines, or “NCS”, (Gerald and Wheatley¹⁰ pages 170-175), which are required to pass through every data point, the curve appears as shown in Figure 2.

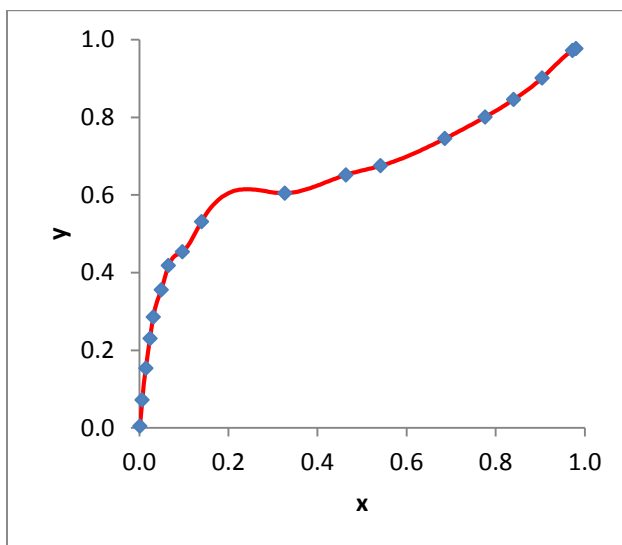


Figure 2: Equilibrium data fitted with NCS

As shown in Figure 2, the NCS fit seems fine for the first 7 data points, as well as for the last 8. However, for x-values from about 0.1 to 0.4, this proposed “equilibrium curve” exhibits some anomalies. Figure 3 is an expanded view of this anomalous region.

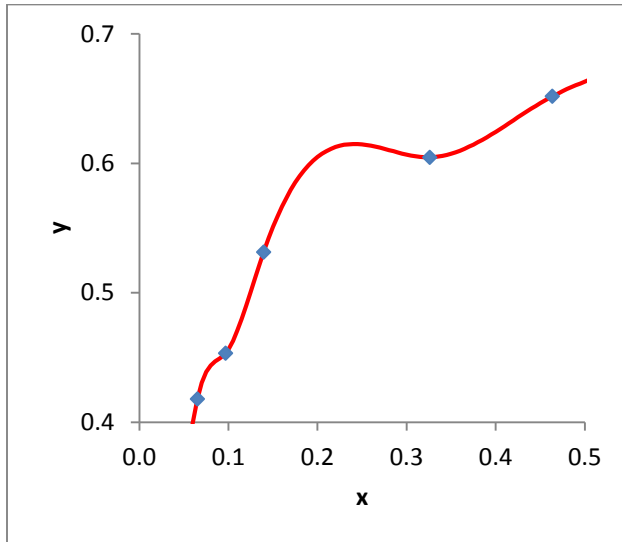


Figure 3: Expanded view of NCS fit

As shown in Figure 3, the first derivative of the NCS fit decreases rapidly as the curve approaches the data point #8 at about $x = 0.1$, and then it increases rapidly as the curve passes through that point. Even stranger, between 0.2 and 0.3 the first derivative becomes negative. These behaviors are not characteristic of an actual equilibrium curve, but both occur in this plot because of the requirement that the NCS pass through every data point.

Thus, a preferred fitting technique is one that follows the overall “form” of the data, but is not required to pass through each and every data point. In this way, slight inaccuracies in the data will be “smoothed out” by the fitted curve. B-splines have this desirable feature, since the “knots”, where two sections of the B-splines join, do not necessarily fall precisely on the data points.

Naturally, another approach is to fit the data to an equation of state thermodynamic model, such as Peng-Robinson or NRTL. However, no single model works well for all cases (that’s what keeps thermodynamicists busy), so the best choice will vary from system to system. That’s why in this paper a single approach is used for all cases.

Although B-splines can be of any order, in this paper 3rd order, or cubic, B-splines (CBS) are used. The CBS fit is formed by joining the various sections ($\mathbf{B}_1, \mathbf{B}_2, \text{etc.}$) of the curve. Each section \mathbf{B}_j is a cubic polynomial defined to span a region of space from one “knot”, located at (XK_j, YK_j) , to the next at (XK_{j+1}, YK_{j+1}) , where these knots are “near” (but not necessarily “on”) the associated data points (x_j, y_j) and (x_{j+1}, y_{j+1}) . At each knot, the two sections that join are required to have the same value as well as the same first and second derivatives: these requirements ensure that the resulting CBS fit is both continuous and smooth. Thus, the only difference between normal cubic splines and cubic B-splines is that, for the NCS, the “knots”

coincide with the given data points, while for CBS, the knots don't necessarily coincide with the data points.

Figure 4 shows the CBS fit for the equilibrium data. As in the previous figures, the blue diamonds are the original data points, and the red line is the fitted equilibrium curve. The black crosses in this figure show the values of the knots used in the CBS fit. As shown, the red line passes through the black crosses, but the black crosses do not necessarily coincide with the blue diamonds.

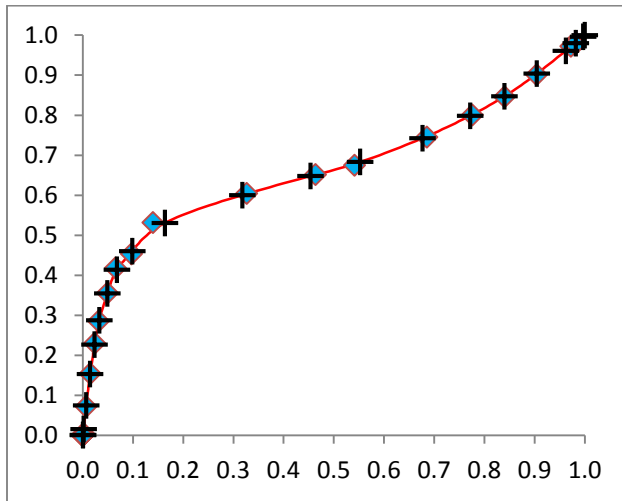


Figure 4: Equilibrium data fitted with CBS

As shown in Figure 4, the CBS fit is much smoother than the fit provided by NCS (in Figure 2), and the CBS fit does not display the anomalies exhibited by the NCS fit in Figure 3.

The location of each knot “ j ” (XK_j , YK_j) is calculated by the following equations in terms of the original equilibrium data x , y :

$$XK_j = \frac{x_{j-1} + 4 \cdot x_j + x_{j+1}}{6}$$

$$YK_j = \frac{y_{j-1} + 4 \cdot y_j + y_{j+1}}{6}$$

Thus, each knot “ j ” is a weighted average of the data values from the points “ $j-1$ ”, “ j ”, and “ $j+1$ ”. These knots are used below in the CBS algorithm for interpolating the values to create the equilibrium curve shown in Figure 4. Note that if an (x, y) point occurs three times in succession, then by necessity a knot will coincide with that data point: the CBS algorithm shown below uses this fact to ensure that the equilibrium curve passes exactly through the points $(0, 0)$ and $(1, 1)$.

As an example, to determine the value of y when $x = 0.5$, the algorithm first has to determine the particular section B_j where the equilibrium curve has an x -value of 0.5: in other words, it has to find the value j such that $XK_j \leq 0.5$ but $XK_{j+1} > 0.5$. If $XK_j = 0.5$ exactly, then the corresponding value of y is YK_j . If not, then the algorithm must interpolate to find the correct y .

With CBS, interpolation is performed in each section \mathbf{B}_j in terms on a parametric variable u , where $0 \leq u \leq 1$. The equations for interpolating x and y are these, where $x_j(u)$ and $y_j(u)$ are the interpolated values:

$$x_j(u) = \frac{(1-u)^3 \cdot x_{j-1}}{6} + \frac{(3 \cdot u^3 - 6 \cdot u^2 + 4) \cdot x_j}{6} + \frac{(-3 \cdot u^3 + 3 \cdot u^2 + 3 \cdot u + 1) \cdot x_{j+1}}{6} + \frac{u^3 \cdot x_{j+2}}{6}$$

$$y_j(u) = \frac{(1-u)^3 \cdot y_{j-1}}{6} + \frac{(3 \cdot u^3 - 6 \cdot u^2 + 4) \cdot y_j}{6} + \frac{(-3 \cdot u^3 + 3 \cdot u^2 + 3 \cdot u + 1) \cdot y_{j+1}}{6} + \frac{u^3 \cdot y_{j+2}}{6}$$

Note that knot “ j ” ($\mathbf{XK}_j, \mathbf{YK}_j$) is equal to $(x_j(0), y_j(0))$, and the next knot “ $j+1$ ” ($\mathbf{XK}_{j+1}, \mathbf{YK}_{j+1}$) is equal to both $(x_j(1), y_j(1))$ and $(x_{j+1}(0), y_{j+1}(0))$: this is how section \mathbf{B}_j spans the space from knot “ j ” to knot “ $j+1$ ” and smoothly joins up with the next section \mathbf{B}_{j+1} . Also note that these equations are valid only for u values in the range of $0 \leq u \leq 1$.

Thus, returning to the interpolation problem posed above, after the algorithm has determined the correct section \mathbf{B}_j , it must next find the value of u that makes $x = 0.5$ by solving a cubic equation (this is why the VBA code contains a function to find the real roots of a cubic polynomial), and then use that u to calculate the interpolated value of y by inserting that u value into the $y_j(u)$ equation given above. From the equation given above for $x_j(u)$, the cubic equation that must be solved is the following, where “ v ” is the value sought (in the present case, $v = 0.5$):

$$a \cdot u^3 + b \cdot u^2 + c \cdot u + d = 0$$

where

$$a = -x_{j-1} + 3 \cdot x_j - 3 \cdot x_{j+1} + x_{j+2}$$

$$b = 3 \cdot x_{j-1} - 6 \cdot x_j + 3 \cdot x_{j+1}$$

$$c = -3 \cdot x_{j-1} + 3 \cdot x_{j+1}$$

$$d = x_{j-1} + 4 \cdot x_j + x_{j+1} - 6 \cdot v$$

Naturally, due to the similarity of the $x_j(u)$ and $y_j(u)$ equations, a similar cubic equation (with the x 's replaced with y 's) can be written for the case where y is known and x is to be determined: this fact is useful for determining the stages, or “steps”, on the McCabe-Thiele diagram, since then the value of y is known and the value of x must be determined from the equilibrium curve.

Visual Basic for Applications (VBA)

The steps required to input the VBA code and create the Excel workbook, while not difficult, are numerous and exacting. They can be skipped by emailing a request for the Excel file to the author at che_prof@hotmail.com.

Before entering the code shown below, check that Excel has the “Developer” tab installed on the ribbon. In the Excel program, look at the tabs installed at the top of the ribbon to see if “Developer” is already installed: typically, its tab is just after the “View” tab. If not, install it (in Excel 2007) by clicking on the Office Button, selecting “Excel Options”, choosing “Popular”, and then checking the “Show Developer tab in the Ribbon” checkbox, as shown in Figure 5. Note that newer versions of Excel have somewhat different procedures for doing this.

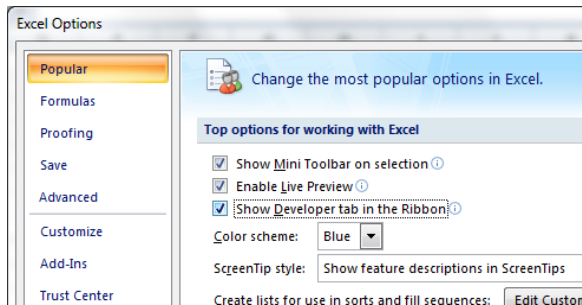


Figure 5: Installing “Developer” in Excel 2007

After “Developer” is installed, click on “Developer”, and then click on “Visual Basic” in the “Code” group to launch the Visual Basic for Applications (VBA) editor (alternatively, hit Alt-F11 to launch the VBA editor). If you have more than one Excel workbook open, navigate in the folder list (on the left of the VBA screen) to select “Microsoft Excel Objects” under the workbook where you want to enter the function. Then click on “Insert” and select “Module”: this will create a new module, “Module 1”, in the workbook: in VBA, modules are where VBA Functions are stored.

Move to the code window (on the right of the VBA screen) and enter the code shown at the end of this paper. Please note the following typographic conventions:

1. Lines beginning with an apostrophe are comments ignored by VBA.
2. Lines that end with a space followed by an underscore “_” are continued on the next line: *both the space and the underscore are absolutely required.*
3. Indentation is ignored by VBA.
4. Blank lines are ignored by VBA.

The code entered consists of two functions: *FXBS*, which implements the B-spline algorithm, and *FXCubic*, which is used to solve the cubic equation encountered in the *FXBS* algorithm.

In the *FXBS* code, “DV” and “IV” refer to the dependent and independent variables, respectively. To use *FXBS* for the example problem of determining y when $x = 0.5$, enter the formula “=*FXBS*(*ydata*, *xdata*, 0.5)” in a cell, where *ydata* and *xdata* are the ranges for the y and x equilibrium data (see Step 9 below for information on naming these data ranges). In this example, *xdata* is the independent variable (since its value is known to be 0.5), and *ydata* is the dependent variable. Similarly, the formula “=*FXBS*(*xdata*, *ydata*, 0.5)” will return the value of x that corresponds to $y = 0.5$ on the equilibrium curve. *FXBS* can also be used to search for an azeotrope by inputting any negative value for *IVvalue*: thus, the formula “=*FXBS*(*ydata*, *xdata*, -1)” will return the value of the azeotrope, if one exists.

The *FXCubic* code is an implementation of a method found on the Internet¹² that uses the Tschirnhaus transformation to form the depressed cubic, which is then solved using the trigonometric method of François Viète. The formula for using *FXCubic* is as follows: “=*FXCubic*(*a*, *b*, *c*, *d*, *xlow*, *xhigh*)”, where *a*, *b*, *c*, and *d* are the coefficients of the cubic polynomial, and in the case of multiple real roots, the code will return a root in the range of *xlow* to *xhigh*, if one exists.

Note: once the VBA code has been input into Excel, the file must be saved as an “Excel Macro-Enabled Workbook”, rather than as a normal “Excel Workbook”: if not, all the VBA code will be lost. Also, any time the workbook is re-opened, Excel *may* have to be told to “Enable Macros” or “Enable Content”: whether or not this step is needed depends on the computer’s security settings.

Excel Worksheet for McCabe-Thiele

The Excel spreadsheet can now be constructed with the use of the above equations and the VBA code. Here are the 65 simple steps in the process. *Note: some of these formulas appear in this paper on more than one line, but they should be entered on a single line with no “Enters”!*

1. **Enter the data shown below into the indicated cells.** To type the entries that span more than one row and/or column, select the appropriate cells, then click “Merge & Center” (on Home/Alignment). To enter the Greek letters and mathematical symbols, click Insert/Symbol. To draw borders around the cells, select the cells and click Home, then select the Borders icon in the Font box. To color the Input cells yellow, select the cells and use Home/Font/Fill Color.

	A	B	C	D
1	Inputs		Range	
2	zf	0.1	$x_b < z_f < x_d$	
3	q	0.8	$> -RR$	
4	x_d	0.85	$z_f < x_d < 1$	
5	x_b	0.01	$0 < x_b < z_f$	
6	RR	3		
7	η	1	$0 < \eta \leq 1$	
8				
9				
10	Outputs			
11	Azeo.		if present, else 0	
12	xf		Intersection of feed	
13	yf		line with op lines	
14	Stages		$N \leq 50$	
15	Nf		Feed Stage	

2. **Name cell B2 “zf”.** Click on cell B2, then enter the name “zf” in the Name Box, as shown below in Figure 6:

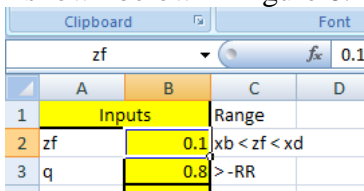


Figure 6: The name “zf” entered in Name Box

3. **Name the other variable cells in Column B.** Use the names “q”, “ x_d ”, “ x_b ”, “RR”, and “eta” in rows 3 to 7, and “Azeo” (no period!), “xf”, “yf”, “N”, and “Nf” in rows 11 to 15.

The purpose of naming these cells is so that the names can be used in the subsequent equations on the spreadsheet. Any mistakes made while entering the names can be corrected by clicking on Formulas/Name Manager (or Ctrl-F3).

4. Enter these headings in cells Q1 to S2:

	Q	R	S
1		Equilibrium Data	
2	Point #	x	y

5. Enter a 1 in cell Q3.

6. Enter the following formula in cell Q4: $=IF(R4<>"",Q3+1,"")$

7. Copy cell Q4 down to cell Q33.

8. Type the x-y data from Table 1 above into columns R and S. As the data are entered, Excel will automatically number the points in Column Q by means of the equations entered above in steps 7 and 8. Note: with the steps given below, the maximum number of data points is 31: if more are required, see step 50 below.

9. Select the x-data entered above (cells R3 to R20 for the present problem) and enter the name "xdata" in the Name Box; then select the y-data entered above (cells S3 to S20 for the present problem) and enter the name "ydata" in the Name Box. Note that if you change the number of points in the x-y data set, you'll have to use the Name Manager (invoked by Ctrl-F3) to edit the definitions of "xdata" and "ydata" to conform to the new ranges.

10. Enter the following formula in cell B11 to find the azeotrope:

$=FXBS(xdata,ydata,-1)$

For the data given, this should give a value of 0.8892... in cell B11.

11. Enter the following formula in cell B12 to calculate xf (by Gossage¹ Eq.'s 15 and 16):

$=IF(q=1,zf,(xd/(1+RR)+zf/(q-1))/(q/(q-1)-RR/(1+RR)))$

For the data given, this should give a value of 0.06053 in cell B12.

12. Enter the following formula in cell B13 to calculate yf (by Gossage¹ Eq. 17):

$=(xd+xf*RR)/(1+RR)$

For the data given, this should give a value of 0.25789 in cell B13.

13. Enter the following formula in cell B14 to calculate N, the number of stages:

$=MAX(A21:A71)-1+(VLOOKUP(MAX(A21:A71)-1,(A21:B71),2)-xb)/((VLOOKUP(MAX(A21:A71)-1,(A21:B71),2)-VLOOKUP(MAX(A21:A71),(A21:B71),2)))$

Note that this (and the next) step will give a result of "#N/A" until step 20 is completed.

This formula calculates the whole number of stages plus the fraction of the final stage needed to reach an x-value equal to xb. For the data given, this should (eventually, after step 20 is done) give a value of 22.53019 in cell B14.

14. Enter the following formula in cell B15 to calculate Nf, the feed stage:

$=INDEX(A21:B71,MATCH(xf,B21:B71,-1),1)+1$

For the data given, this should (eventually) give a value of 21 in cell B15.

Now the calculations for each stage can be performed.

15. Type the following data into the indicated cells on the spreadsheet, including the 0:

	A	B	C
19	Stages		
20	#	x	y

21	0		
22			

16. In cells B21 and C21, enter the following:

=xd

This sets both x_0 and y_0 equal to x_D .

17. In cell A22, enter this formula:

=IF(A21<>"", (IF(C21>xb, A21+1, "")), "")

This increments the Stage Number, *unless* the preceding Stage Number (cell A21) is blank or the previous value of y_i (cell C21) is less than x_B .

18. Type the following formula in cell B22 to calculate x_i (like Gossage¹ Eq.'s 18 and 20, but with x_{eq} replaced by the CBS-interpolated result):

=IF(A22<>"", (B21-eta*(B21-FXBS(xdata, ydata, C21))), "")

Note that if the current Stage Number (cell A22) is blank, this formula leaves the value of x_i blank as well.

19. Type the following formula in cell C22 to calculate y_i (by Gossage¹ Eq.'s 21 and 22):

=IF(A22<>"", (IF(B22>xf, (xd+B22*RR)/(1+RR),
((yf-xb)/(xf-xb)) * (B22-xb) +xb)), "")

Note that if the current Stage Number (cell A22) is blank, this formula leaves the value of y_i blank as well; otherwise, it calculates y_i by Gossage¹ Eq. 21 (if $x_i > x_f$) or Eq. 22.

20. Now copy cells A22 to C22 all the way down to C71. Click once on cell A22, then while holding the left mouse button, drag across to select cells A22 to C22, then type Ctrl-C to copy these three cells. Now while these cells are still selected, type Ctrl-G to open the “Go To” dialog box (as shown in Figure 7), enter “c71” in the Reference box, then, *while holding the Shift key*, click the OK button. This will select the entire range from A22 to C71. Now type Ctrl-V to paste the equations. Note: Excel provides other methods of performing this “copy & paste” task, but when the final destination is known, this method is convenient to use. This Excel “secret” is described by MacDonald⁹, page 82.

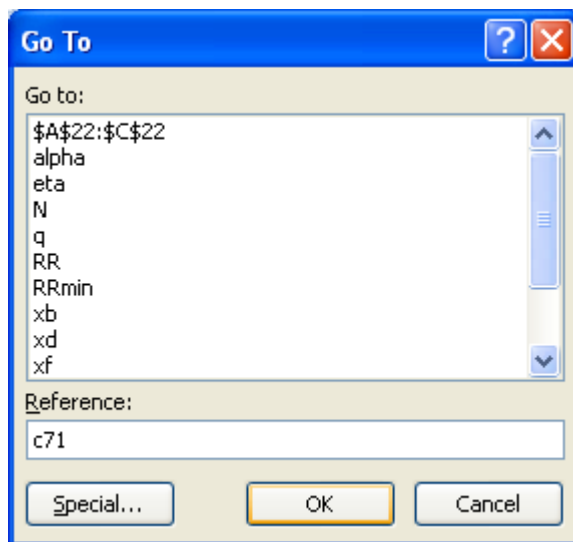


Figure 7: The “Go To” dialog box

21. Now check the Stage values to ensure that all the preceding formulas have been entered correctly:

	A	B	C
19	Stages		
20	#	x	y
21	0	0.85000	0.85000
22	1	0.84346	0.84509
23	2	0.83722	0.84042
24	3	0.83116	0.83587
25	4	0.82515	0.83136
26	5	0.81906	0.82680
27	6	0.81277	0.82208
28	7	0.80615	0.81711
29	8	0.79902	0.81177
30	9	0.79121	0.80591
31	10	0.78247	0.79935
32	11	0.77250	0.79187
33	12	0.76090	0.78317
34	13	0.74705	0.77279
35	14	0.72998	0.75998
36	15	0.70802	0.74351
37	16	0.67820	0.72115
38	17	0.63488	0.68866
39	18	0.56584	0.63688
40	19	0.42068	0.52801
41	20	0.15955	0.33216
42	21	0.04303	0.17203
43	22	0.01680	0.04335
44	23	0.00398	-0.01955
45			

Note that cell A45 is blank: that's because the y-value for the preceding Stage #23, -0.01955, is smaller than x_b , 0.01.

Now the data need to be arranged in a table so that Excel can plot the McCabe-Thiele diagram. The secret to drawing plots with multiple lines in Excel is that all the data should be in the same table (not absolutely required, but convenient), with the "x-axis" variables all in a single column. The various "y-axis" variables are then lined up versus the appropriate x-axis variables, with spaces left for unneeded values.

22. Type the following headings in Row 19, one per column, starting in Cell D19 and ending in Cell P19:

#, x, 45° line, Rect line, Strip line, Feed line, Eq line, Stages, Distillate, Feed, Bottoms, Orig Data, Azeotrope

Column E (the one labeled “x”) will be the x-axis variable in the plot. Columns F through N will be the nine different lines and Columns O and P will be the two sets of points that will be plotted to form the McCabe-Thiele diagram. Column D, labeled “#”, is for the Stage Numbers.

23. In cell L20 (under “Distillate”), enter a 0. In cells E20, E21, and L21, enter the following formula:

=xd

These four cells represent two (x,y) points to draw the dashed line from the x-axis up to the 45° line at $x=x_D$.

24. In cell M22 (two rows under “Feed”), enter a 0. In cells E22, E23, and M23, enter the following formula:

=zf

These four cells represent two (x,y) points to draw the dashed line from the x-axis up to the 45° line at $x=z_F$.

25. In cell N24 (four rows under “Bottoms”), enter a 0. In cells E24, E25, and N25, enter the following formula:

=xb

These four cells represent two (x,y) points to draw the dashed line from the x-axis up to the 45° line at $x=x_B$.

26. Enter “0” in cells F26 (six rows under “45° line”), and E26. Enter “1” in cells E27 and F27.

These four cells represent two (x,y) points that will be connected to draw the 45° line.

27. In cells E28 and G28 (eight rows under “Rect line”), enter this formula:

=xd

28. In cell E29 enter this formula:

=xf

29. In cell G29 enter this formula:

=yf

These four cells (from steps 27-29) represent two (x,y) points that will be connected to draw the Rectifying operating line.

30. In cell E30 enter this formula:

=xf

31. In cell H30 (ten rows beneath the “Strip line” heading) enter this formula:

=yf

32. In cells E31 and H31 enter this formula:

=xb

These four cells (from steps 30-32) represent two (x,y) points that will be connected to draw the Stripping operating line.

33. In cells E32 and I32 (twelve rows below the “Feed line”), enter this formula:

=zf

34. In cell E33 enter this formula:

=IF(q=1,xf,IF(AND((yf-1)/xf<q/(q-1),q/(q-1)<1),0,xf+(q-1)*(1-yf)/q))

For the data given, this should give a value of 0 in cell E33.

35. In cell I33 enter this formula:

=IF(q=1,1,IF(AND((yf-1)/xf<q/(q-1),q/(q-1)<1),yf-q*xf/(q-1),1))

For the data given, this should give a value of 0.5 in cell E34. These four cells (from steps 33-35) represent two (x,y) points that will be connected to draw the Feed line. The equation

for the feed line depends on the value of q , which is defined as follows⁸ (equation 11.4-12, page 710):

$$q = \frac{\left(\begin{array}{c} \text{heat needed to vaporize 1 mole} \\ \text{of feed at entering conditions} \end{array} \right)}{\left(\begin{array}{c} \text{molar latent heat of} \\ \text{vaporization of feed} \end{array} \right)}$$

Thus, $q=0$ means the feed is a saturated vapor; $q=1$ means the feed is a saturated liquid; $0 < q < 1$ means the feed is a saturated liquid-vapor mixture; $q < 0$ means the feed is a superheated vapor, and $q > 1$ means the feed is a subcooled liquid.

36. Now comes the Equilibrium curve. In cell E34 enter “0”. In cell J34 (fourteen rows beneath the “Eq Line” heading) enter the following formula:

=FXBS (ydata, xdata, E34)

37. In cell E35 enter the following formula:

= E34+0.005

38. Copy cell J34 into cell J35.

39. Now copy cells E35:J35 all the way down to cell J234. Use MacDonald’s “secret” explained above in step 20. This gives 201 points for Excel to use to draw the Equilibrium curve. *Note: if cell J234 says “#VALUE!”, manually enter the value of “1” into cell E234.*

Due to round-off error, adding 0.005 200 times can give a value slightly larger than 1.

40. Finally, the Stage data are transferred onto the table. In cell D235 enter this formula:

= A21

41. In cell E235 enter this formula:

= IF (D235<>"", VLOOKUP (D235, \$A\$21:\$C\$71, 2), 0)

42. In cell K235 enter this formula:

=IF (D235<>"", IF (D235<N, VLOOKUP (D235, \$A\$21:\$C\$71, 3), E235), 0)

43. In cell E236 enter this formula:

=IF (D235<>"", VLOOKUP (D235+1, \$A\$21:\$C\$71, 2), 0)

44. In cell K236 enter this formula:

=IF (D235<>"", IF (D235<N, VLOOKUP (D235, \$A\$21:\$C\$71, 3), E235), 0)

Yes, exactly the same formula is entered into both K235 and K236. One way to save some typing is to type the formula into K235 and then hit “Enter”. Then click once more on cell K235, move the cursor to the “Formula Entry” box (just to the right of the “Name Box”), highlight the entire formula, hit Ctrl-C to copy it, and then hit “Esc” (to avoid inadvertently changing the formula in K235). Then click on cell K236 and hit Ctrl-V to paste the formula.

45. Leave cell D236 blank. In cell D237 enter this formula:

= IF (D235<>"", (IF (D235<MAX (\$A\$21:\$A\$71), D235+1, "")), "")

Here, the Stage # will be incremented by 1 as long as the following two criteria are true: the previous Stage # (cell D235) is *not* blank, and the previous Stage # is less than the largest value among the Stages in Column A. If either criterion is not met, then the current Stage # will be blank. Note that if the current Stage # is blank, then the formulas given above in steps 41-44 will give values of “0”.

46. Highlight cells E235:K236, hit Ctrl-C to copy, then click cell E237 and hit Ctrl-V to paste. Row 235 represents the “tread” of the step, while row 236 is its “riser”.

47. Now copy cells D237:K238 all the way down to cell K336. Note that most of the rows will just contain 0’s: in the present problem, only rows 282 and above will be non-zero. The remaining rows are present for problems that contain many more stages (up to 50).

48. In cell E337 enter the following formula:

=R3

49. In cell O337 enter the following formula:

= (ISBLANK (S3) , "" , S3)

50. Now copy cells E337:O337 all the way down to cell O368. This will copy the Original Data into the array, leaving blank “Orig Data” values for any missing data, up to the maximum number of data points: 31. If more data points are required, simply extend the ranges as appropriate.

51. Enter the following formula into cells E368 and P369:

=Azeo

This will copy the value of the Azeotrope.

Excel Formula Explanation

Several of the formulas entered above make use of advanced built-in Excel functions: in particular, MAX, VLOOKUP, INDEX, and MATCH. Gossage¹ gives a detailed explanation of the use of these functions.

Drawing the McCabe-Thiele Diagram

Now that the data have been placed in the appropriate tabular form, Excel can now draw the McCabe-Thiele diagram. Here are the steps in the process.

52. Select cells E19:P369. A simple way is to click on cell P369, then use the “Go To” shortcut described above in step 20, using a Reference of E19.

53. Click the “Insert” tab on the Ribbon, and in the “Charts” region click the arrow below “Scatter” and select “Scatter with Straight Lines”. This will draw the chart.

54. Move the chart so its upper-left-hand corner is in cell F1. To move the chart, point the cursor at any edge of the chart (*not* the edge of the graph itself). The cursor will change to a four-way arrow. Click and drag the chart to the desired location.

55. Remove the horizontal gridlines. Click on any of the values on the y-axis: Excel will then select all of them. Right-click and select “Format Major Gridlines...”. Click “Close”. Now all the horizontal gridlines will be selected. Hit the “Delete” key to delete all of them.

56. Format the y-axis. Click on any of the values on the y-axis: Excel will then select all of them. Right-click and select “Format Axis...”. On “Axis Options”, after “Maximum” click the “Fixed” radio button and enter 1. After “Major unit” click the “Fixed” radio button and enter 0.1. Below “Axis Options” click “Number”. From the Category list choose Number and set the Decimal places to 1. Click “Close”.

57. Format the x-axis. Click on any of the values on the x-axis: Excel will then select all of them. Right-click and select “Format Axis...”. On “Axis Options”, after “Maximum” click the “Fixed” radio button and enter 1. After “Major unit” click the “Fixed” radio button and enter 0.1. Select “Number”. From the Category list choose Number and set the Decimal places to 1. Click “Close”.

58. Resize the chart. Click the resize handle (it looks like three dots arranged in a right triangle) at the bottom right of the chart: the cursor changes to a two-headed diagonal arrow. Click and drag so the bottom right corner of the chart falls in cell O18.

59. Make the “Distillate” line dashed rather than solid. Right-click on the line: Excel will then highlight Cells E20:E369, Cell L19, and Cells L20:L369 to indicate the selected data.

Select “Format Data Series...”. In the “Format Data Series” dialog box, select Line style, then select Dash type and choose “Round Dot”. Click “Close”.

60. **Repeat step 52 twice more: once for the Feed line, and again for the Bottoms line.** If the “Bottoms” line is hard to find, in the Ribbon, under “Chart Tools” select “Layout”, then in the “Current Selection” region, click the drop-down list and select Series “Bottoms”, then click the “Format Selection” and continue as above.

61. **Smooth the Equilibrium curve.** Right-click on the curve and select “Format Data Series...”. In the “Format Data Series” dialog box, select Line style, then check the “Smoothed line” box and click “Close”.

62. **Make both the 45° and Stages lines black.** Right-click on the 45° line, and select “Format Data Series...”, select Line color, then select the Solid line radio button. In the Color box, click the arrow and choose black. Repeat for the Stages line.

63. **Format the Orig Data series.** Select the Orig Data series. Under “Marker Options”, select “Built-in”, and for “Type”, select the diamond. Select “Marker Fill”, choose “Solid fill” and for “Color”, pick red. Select “Line Color”, pick “No line”, then “Close”.

64. **Repeat the above step for the Azeotrope series.** This time, choose a yellow square.

65. **Compare the results with those shown in Figures 8 and 9 to ensure that all of the steps have been completed satisfactorily.**

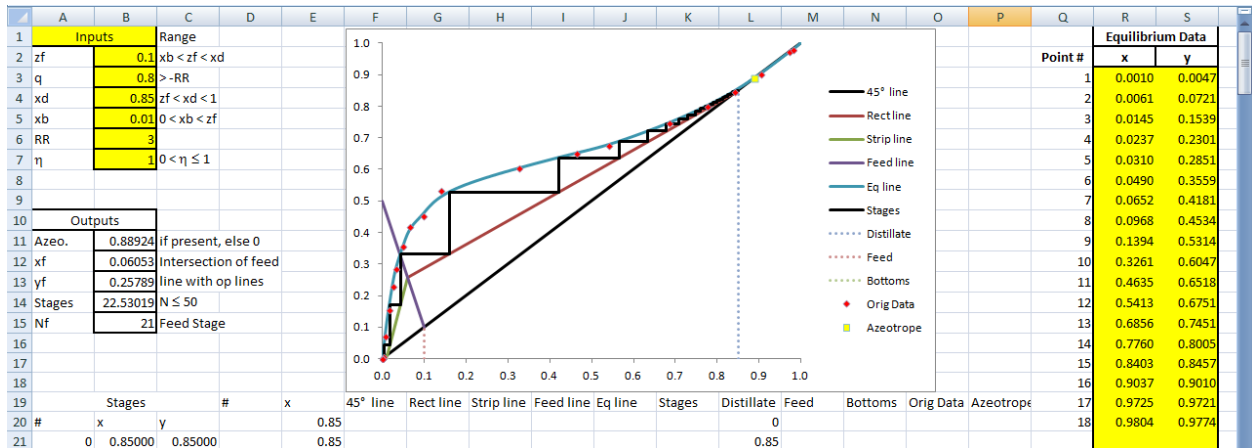


Figure 8: The completed spreadsheet

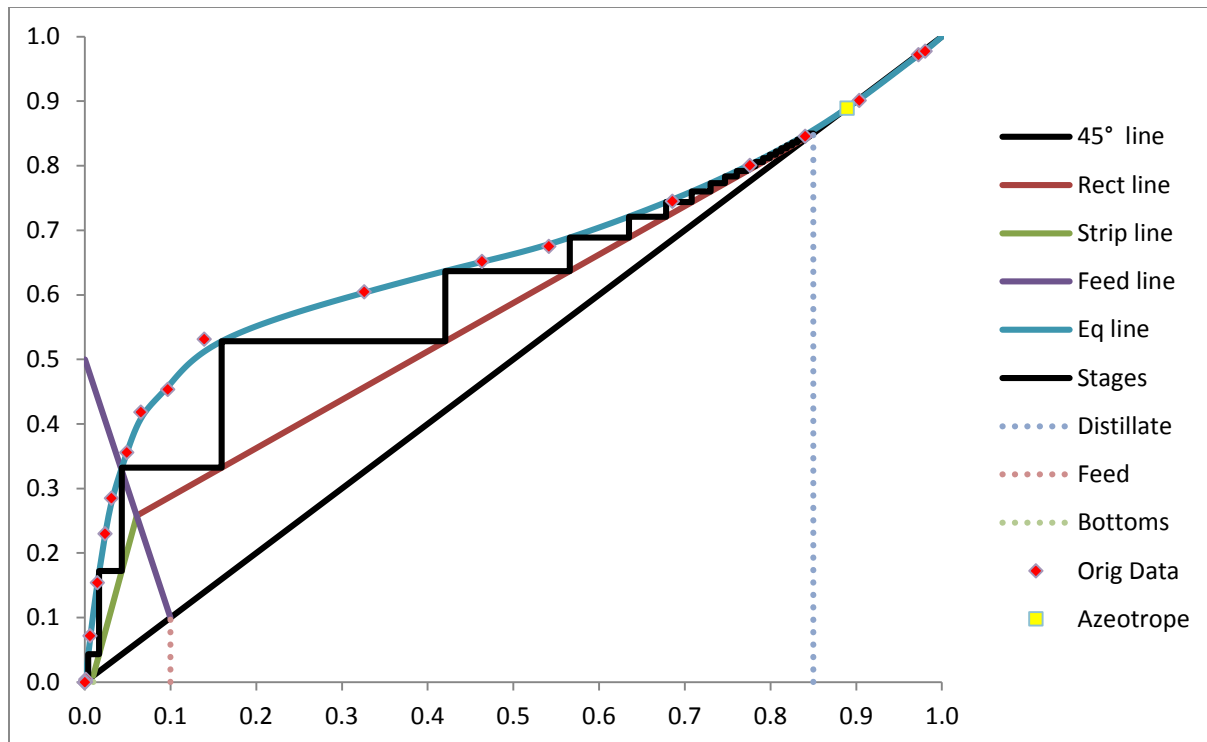



Figure 9: Expanded view: McCabe-Thiele diagram

Thus, for the example shown, the system has an azeotrope at an ethanol mole fraction of about 0.889 (Cell B11). The intersection of the feed line with the operating lines occurs at about (0.061, 0.258) (Cells B12 and B13). To achieve the desired separation for the given inlet conditions, a column with just over 22.5 (cell B14) equilibrium stages is required, with the feed entering at stage 21 (cell B15).

Using Solver for What-If Analysis

Excel's Solver add-in can be used to perform "what-if" analysis: for example, for the same feed conditions and desired tops and bottoms purities above, what Reflux Ratio would be required for an existing distillation column with 30 stages?

If Excel's "Solver" add-in is not installed, follow these steps in Excel 2007 (in newer versions of Excel, replace steps a and b with a. Click "File" and b. Click "Options"):

- Click the Office Button: 
- Click "Excel Options"
- Select "Add-Ins"
- Click "Go"
- Choose "Solver" and click "Ok"

Once Solver is installed, click on the Data tab in the Ribbon, then from the Analysis group select Solver: the Solver Parameter dialog box then appears. In the Set Objective box, type the letter N (which earlier in step 3 of "Excel Worksheet for McCabe-Thiele" was defined as the name for

the Number of Stages). Click the radio button for “Value of” and enter 30 in the box. In the By Changing Variable Cells box enter the name RR (which also earlier was defined as the name for the Reflux Ratio). Finally, click the Solve button and Solver will quickly determine that a RR of just over 2.48 is required for a 30-stage column.

Limitations

The method used to construct the spreadsheet has a few limitations:

1. The equilibrium x - y data must be arranged in ascending order.
2. The x - y data needn't contain either (0,0) or (1,1), as these will be accounted for automatically in the *FXBS* code.
3. The maximum number of x - y data points is 31, though this can be increased as explained in step 50.
4. The maximum number of stages is 50. This number can be increased¹ by changing the range references (those that end in “71” would have to be increased) and expanding the plotting array.
5. For binary systems with two azeotropes, the *FXBS* code will find only the larger one. Some binary systems have multiple azeotropes: two such systems¹³ are butanoic acid (1) /butyl butanoate (2) (at 93.33 kPa, $y_1 = 0.6532$ and $y_1 = 0.8639$), and C_6F_6 (1) / C_6H_6 (2) (at 101.33 kPa, $y_1 = 0.7600$ and $y_1 = 0.1832$). For such systems, a work-around is to find the first azeotrope, then reverse the order of the species, recalculate the equilibrium data, and then find the second azeotrope.

Conclusions

This paper provides a method to use Microsoft Excel, in conjunction with Excel's Visual Basic for Applications, to calculate and display the McCabe-Thiele diagram for a binary nonideal distillation, given the equilibrium x - y data, the feed conditions, the desired tops and bottoms purities, and the reflux ratio. The algorithm presented automatically smooths and fits the equilibrium data to an equilibrium curve by means of the cubic B-spline method. Furthermore, if the binary system contains an azeotrope, the algorithm locates it. This paper thus provides a quick and easy method to produce McCabe-Thiele diagrams for nonideal binary distillation: a boon to instructors and students alike.

Bibliography

1. Gossage, John L., “Plotting McCabe-Thiele Diagrams in Microsoft Excel”, *Computers in Education*, Volume 6, Number 3, pages 20-30, July-September 2015.
2. <http://excelcalculations.blogspot.com/2012/05/mccabe-thiele-binary-distillation-excel.html>; retrieved 12/17/2015.

3. http://chemsof.com/mccabe_thiele_diagram.html; retrieved 12/17/2015.
4. <http://www.mathworks.com/matlabcentral/fileexchange/4472-mccabe-thiele-method-for-an-ideal-binary-mixture/content/McCabeThiele.m>; retrieved 12/17/2015.
5. <http://checalc.com/calc/binary.html>; retrieved 12/17/2015.
6. Burns, Mark A. and James C. Sung, "Design of Separation Units Using Spreadsheets", *Chemical Engineering Education*, pages 62-69, Winter 1996.
7. Mathias, Paul M., "Visualizing the McCabe-Thiele Diagram", *Chemical Engineering Progress*, pages 36-44, December 2009.
8. Geankoplis, Christie John, *Transport Processes and Separation Process Principles*, 4th Edition, Prentice-Hall Professional Technical Reference, Upper Saddle River, NJ, 2003.
9. MacDonald, Matthew, *Excel 2007: The Missing Manual*, Pogue Press; Revised edition, 2007.
10. Gerald, Curtis F., and Patrick O. Wheatley, *Applied Numerical Analysis, Seventh Edition*, Pearson Addison Wesley, New York, 2004.
11. Orjuela, A.; Yanez, A. J.; Vu, D. T.; Miller, D. J.; Lira, C. T., "Phase equilibria for reactive distillation of diethyl succinate Part I. System diethyl succinate + ethanol + water", *Fluid Phase Equil.*, Volume 290, pages 63-67, 2010.
12. <http://www.1728.org/cubic2.htm>; retrieved 08/25/2015.
13. Gmehling, J., J. Menke, J. Krafczyk, K. Fischer, J.-C. Fontaine, and H. V. Kehiaian, "Azeotropic Data for Binary Mixtures", pages 6-210 to 6-228 in *CRC Handbook of Chemistry and Physics, 92nd Edition*, edited by William M. Haynes, Taylor & Francis Group, Boca Raton, FL, 2011.

Acknowledgments

The author wishes to acknowledge two Lamar University colleagues who provided assistance with the present work: Dr. Peyton Richmond, who suggested the use of cubic B-splines for this project and provided the Gerald & Wheatley reference¹⁰, and Dr. Daniel Knight, for his help on permissible shapes for the equilibrium curve.

VBA Code

```
Public Function FXBS(DVrange As Range, IVrange As Range, _
    IVvalue As Double) As Double
    Dim DV() As Double
    Dim IV() As Double
    Dim DVk() As Double
    Dim IVk() As Double
    Dim u As Double
    Dim j As Long
    Dim n As Long
    Dim newn As Long
    Dim a As Double
    Dim b As Double
```

```

Dim c As Double
Dim d As Double
Dim flow As Double
Dim fhigh As Double

If IVvalue = IVvalue ^ 2 Then
' If true, then IVvalue = 0 or 1
    FXBS = IVvalue
    Exit Function
End If

n = DVrange.Rows.Count
newn = n

ReDim DV(n + 5), IV(n + 5), DVk(1 To n + 4), IVk(1 To n + 4)

For j = 0 To 2
    DV(j) = 0
    IV(j) = 0
    DV(j + n + 3) = 1
    IV(j + n + 3) = 1
Next j

For j = 3 To n + 2
    DV(j) = DVrange(j - 2)
    IV(j) = IVrange(j - 2)
    If DV(j) = 1 Then newn = newn - 1
Next j

n = newn

For j = n + 4 To n + 3 Step -1
    DVk(j) = (DV(j - 1) + 4 * DV(j) + DV(j + 1)) / 6
    IVk(j) = (IV(j - 1) + 4 * IV(j) + IV(j + 1)) / 6
    If IVk(j) <= IVvalue Then GoTo CubicDef
Next j

fhigh = IVk(n + 3) - DVk(n + 3)

For j = n + 2 To 1 Step -1
    DVk(j) = (DV(j - 1) + 4 * DV(j) + DV(j + 1)) / 6
    IVk(j) = (IV(j - 1) + 4 * IV(j) + IV(j + 1)) / 6
    If IVk(j) <= IVvalue Then GoTo CubicDef
    If IVvalue < 0 Then
' IVvalue < 0 indicates an azeotrope search
        flow = IVk(j) - DVk(j)
        If fhigh * flow <= 0 Then GoTo CubicDef
        fhigh = flow
    End If
Next j

' IVk(j) <= IVvalue < IVk(j+1)

```

```

' Define the cubic equation: a*u^3 + b*u^2 + c*u + d = 0

CubicDef:
a = -IV(j - 1) + 3 * IV(j) - 3 * IV(j + 1) + IV(j + 2)
b = 3 * IV(j - 1) - 6 * IV(j) + 3 * IV(j + 1)
c = -3 * IV(j - 1) + 3 * IV(j + 1)
d = IV(j - 1) + 4 * IV(j) + IV(j + 1) - 6 * IVvalue

If IVvalue < 0 Then
' IVvalue < 0 indicates an azeotrope search
  a = a - (-DV(j - 1) + 3 * DV(j) - 3 * DV(j + 1) + DV(j + 2))
  b = b - (3 * DV(j - 1) - 6 * DV(j) + 3 * DV(j + 1))
  c = c - (-3 * DV(j - 1) + 3 * DV(j + 1))
  d = d - (DV(j - 1) + 4 * DV(j) + DV(j + 1) - 6 * IVvalue)
End If

' Now find value of u that solves the cubic eq'n

u = FXCubic(a, b, c, d, 0, 1)

FXBS = ((1 - u) ^ 3 * DV(j - 1) +
(3 * u ^ 3 - 6 * u ^ 2 + 4) * DV(j) +
(-3 * u ^ 3 + 3 * u ^ 2 + 3 * u + 1) * DV(j + 1) +
u ^ 3 * DV(j + 2)) / 6

End Function

Public Function FXCubic(a As Double, b As Double, c As Double, _
d As Double, xlow As Double, xhigh As Double) As Double
Dim f As Double
Dim g As Double
Dim h As Double
Dim i As Double
Dim j As Double
Dim k As Double
Dim m As Double
Dim n As Double
Dim p As Double
Dim r As Double
Dim s As Double
Dim t As Double
Dim u As Double
Dim x As Double
Dim eps As Double

eps = (a ^ 2 + b ^ 2 + c ^ 2 + d ^ 2) * 0.0000000001
If eps = 0 Then
  FXCubic = 0
  Exit Function
End If

If Abs(a) < eps Then

```

```

' We have the quadratic  $b*x^2 + c*x + d = 0$  instead of a cubic
  If Abs(b) < eps Then
    ' We have the linear equation  $c*x + d = 0$ 
      FXCubic = -d / c
      Exit Function
    End If
    FXCubic = (-c + (c ^ 2 - 4 * b * d) ^ (1 / 2)) / (2 * b)
    If FXCubic >= xlow And FXCubic <= xhigh Then Exit Function
    FXCubic = (-c - (c ^ 2 - 4 * b * d) ^ (1 / 2)) / (2 * b)
    Exit Function
  End If

' Apply the method shown at http://www.1728.org/cubic2.htm

f = (3 * c / a - b ^ 2 / a ^ 2) / 3
g = (2 * b ^ 3 / a ^ 3 - 9 * b * c / a ^ 2 + 27 * d / a) / 27

' If f = g = 0, then all three roots are real and equal
If (Abs(f) < eps And Abs(g) < eps) Then
  FXCubic = (Abs(d / a) ^ (1 / 3)) * Sgn(d / a)
  Exit Function
End If

h = g ^ 2 / 4 + f ^ 3 / 27

' If h <= 0, three real roots; if h > 0, only one real root

If h <= 0 Then
  i = Sqr(g ^ 2 / 4 - h)
  j = i ^ (1 / 3)
  x = -g / (2 * i)
  ' k = Acos(x), but VBA has no built-in Acos function
  k = Atn(-x / Sqr(-x * x + 1)) + 2 * Atn(1)
  FXCubic = 2 * j * Cos(k / 3) - b / (3 * a)
  If FXCubic >= xlow And FXCubic <= xhigh Then Exit Function
  m = Cos(k / 3)
  n = 3 ^ (1 / 2) * Sin(k / 3)
  p = -b / (3 * a)
  FXCubic = -j * (m + n) + p
  If FXCubic >= xlow And FXCubic <= xhigh Then Exit Function
  FXCubic = -j * (m - n) + p
  Exit Function
End If

' h > 0, so only one real root

r = -g / 2 + Sqr(h)
If (r <> 0) Then s = (Abs(r) ^ (1 / 3)) * Sgn(r) Else s = 0
t = -g / 2 - Sqr(h)
If (t <> 0) Then u = (Abs(t) ^ (1 / 3)) * Sgn(t) Else u = 0
FXCubic = (s + u) - b / (3 * a)

```

End Function