Bo Svensmark, svensmark@plen.ku.dk
Analytical Chemistry, PLEN, KU, 2017-2024

Tutorial for the Excel spreadsheet ChromSimLC

This sheet and the associated Visual Basic Macros (VBA) can be used to predict chromatograms for 140+ compounds on 4+ commercial HPLC-columns with ACN or MeOH as eluent. Plus (+) means that you can extend the range by adding literature values for compounds and columns, but some formulas must be corrected accordingly.

ChromSimLC works on both Mac and Windows computers. A version with a Fortran subroutine is about double as fast as the VBA version, but it has not been updated from 32 to 64 bit yet.

You do not need to know about the Solvent Parameter model (Abraham's model) and the Linear Solvent Strength model (LSS) to use this spreadsheet - but the chromatograms are calculated using the Solvation Parameter model to calculate the retention factors for up to 11 compounds on 2 columns at a time. The Linear Solvent strength mathematics is used to calculate the chromatograms for isothermal or gradient elution LC in Excel. The VBA macro simulates the chromatograms by digital simulation using the plate theory for chromatography, and this allows simulations where the log(retention factor)'s are not a linear function of eluent strength.

The Excel sheet can only calculate for a simple linear gradient, whereas the VBA subroutine can simulate any gradient you want (but you must enter gradient parameters manually in sheet *Macro input*).

There are 5 chromatograms in the sheet:

1) **Linear green:** Linear log(k) vs eluent strength ϕ
2) **Polynomal green:** First, second or third order polynomial fit of log(k) vs ϕ
3) **Linea blue:** Linear log(k) vs ϕ
4) **Polynomial blue:** First, second or third order polynomial fit of log(k) vs ϕ
5) **Reference:** May be one of the chromatograms 1-4 or any input you like.

All user input are in the white cells. If you want to modify the other cells you must unprotect the sheet first.

1) Predict retention times with a linear model for *log(t)* vs ϕ (Excel sheet LC)

This is the simplest function and is very fast. But – the predictions will only good in the case where *log(k)* vs *T* is strictly linear in the whole gradient range.

You can either:

- Compare 2 different columns with the same gradient
- Compare 2 different gradients on the same column
- Compare 2 different gradients on 2 different columns (may be used for optimization of HPLC)

To use this function:

1) **Enter names of the analytes in the cells C6:C16.** Each name must be identical to the name of one of the 140+ compounds in the table in the sheet *Compounds.* The only exception is that the case of the letters does not matter, i.e. both *Hexadecane* and *hexadecane* is ok. You will get a warning if the spelling is wrong.
2) **Enter the relative concentrations cells B6:B16**. Usually use 1 for a compound and 0 for rows without data. You can also use this to remove or amplify a compound for peak identification.
3) **Enter the names of the 2 columns in cell E20 and E22** and press the *Update column parameters button.* The name must be identical to one of the 4+ columns you can see in cell R6:R26. If the names are not found in the column sheet, unfortunately another column will be chosen, but the warning ***Please update column parameters!*** will not go away.
4) **Enter the parameters for two gradients in** F20:N20 for column 1 (green chromatograms) and F22:N22 for column (blue chromatogram) respectively. The meaning of the parameters should explain them self (for the parameter *Beta-ratio* see next point). Note though, that only a single linear gradient is allowed, but you may add an isocratic start by adding the isocratic time to tD, the dwell time. Also, if the chromatogram is longer than the program (B22 > J20 or J22) an isocratic part is added at this time. You can see the gradients in the plot to the right. If ϕ *end* <= ϕ *start* an isocratic run will be calculated and *tD* and *tG* are not used.
5) **Beta-ratio.** The Solvent Strength Parameters are for the chemical properties of the columns, but the same column type can be made with different physical dimensions, i.e. capillary diameter and amount of stationary phase. So if your column has other properties than the columns used for the table values, all retention factors will change by a constant ratio, which you can enter here. Ratio > 1 means more retention.
6) **The amplification and offset** (*Amplify* and *Offset*) are used to amplify and position the chromatograms in the graph. Try different values to see the effect. Use 0 to hide a curve.
7) **Finally, you may enter the retention times for a reference chromatogram** in the cells E6:F16. If you do not have a reference chromatogram (e.g. experimental values) you can set one of the calculated chromatograms as a reference by use of the button to the right of the graph. To remove the reference, just delete the values in E6:F16. If you only have retention times for a reference, you may copy the widths from one of the calculated chromatograms for comparison.
8) **The chromatograms can be seen in the graph**, and the retention times, peak widths, relative retention etc. in the tables below it B51:I79. The Column differences in cell M16:N16 and P6:Q26 are relative to Column 1 according to the Solvation Parameter model.
8) **Please note that the peak order in the tables may be different from the elution order**. In this case it is advised to sort the input according to the elution order – so press the button for sorting of the input just above the graph. The sorting of the input is according to the reference chromatogram.

Now you can try to change the input and see the results immediately. Try making a program for the blue chromatogram. Start with the same parameters as the green chromatogram and try different variations. You may also play with the other parameters for the columns, intensity and offset.

If you want to optimize the gradient you may get help by looking in the plots of *log(k)* vs ϕ in the sheet *SPmodel.* Here you can also see a linear and a quadratic fit to judge if the linear fit seems good enough.

Advanced users: In cell A110:O132 there is a scheme for a *What-If-Analysis*, i.e. you can get many results for variation of a single input parameter. It is a little complicated, but just follow the instructions in the spreadsheet and/or use the help function in Excel for *Data Table.* This function seems not to work unless you unprotect the sheet first: Go to *Review* and click *Unprotect sheet*. Protect the sheet again afterwards.

If you want to predict chromatograms using a nonlinear *log(k)* vs ϕ relation, you must use the *nonlinear model*, see next section.

2) Predict retention times with a nonlinear model for *log(t)* vs ϕ (Excel sheet LC + VBA)

One simulation takes a few seconds to one minute depending on the computer power and the parameters for the simulation (see *observations* below). The predictions can be much more accurate than the linear calculations in Excel in the case **when *log(k)* vs ϕ is not strictly linear** in the whole gradient range.

To use this function:

1) **Enter the order of the polynomial fit of *log(k) vs T*** in cell L21 and L23 for the green and blue column respectively.
2) **Press the button *Update green* or *Update blue*.** The macros associated with these buttons simulate the chromatograms and returns the resulting retention times and peak widths in cell B68:I78.
2) **All chromatograms can be seen in the graph**, and the retention times, peak widths, relative retention etc. in the table below it B53:I78.
4) **If you change any run parameter or the polynomial order** a warning will be shown in cell J10 or J12 asking you to ***Update green*** **or** ***Update blue.*** This indicates that the retention times in chromatogram #2 and #4 do not correspond to the parameters in the spreadsheet. After the macro have been run the cells will indicate that this is ***ok*** **or** ***ok.***
5) That's all.

The simulations will use the same parameters for the compounds, the two columns and the temperature program as the Excel calculations.

If the simulation takes too long time, press *Escape* and then press *Ok* in the message window. The program may stop with an error message if the simulation is too long. Close the spreadsheet and reduce the number of theoretical plates for the simulation in the sheet *Macro input* as explained below.

Now you can try to change the input parameters and see the results after pressing the *Update* button whenever the Update warning is given. You may see the following observations if you try:

3) Observations: VBA simulations *vs* Excel calculations

Use the same column for both chromatograms, in cell E20 and E22. Then make an isocratic run

for column 1 and a temperature program for column 2. Fit the parameters according to the analytes you have chosen so you have two reasonable good chromatograms. Use a relative short *tmax* in cell B22, i.e. 10-20 min. For the isothermal run, some peaks may have a longer retention time, but just forget about them in the graph.

Then set the polynomial order equal to 1 in L21 and L23 and run the simulation by pressing *Update* button. Now both Excel and simulation will predict the chromatograms for identical runs, i.e. a linear fit of log(k) vs ϕ. But you can see small differences in the numbers when you compare the results from Excel in cell B53:I63 and the results from Fortran in cells B68:I78, as explained below.

The peak widths in the beginning of the chromatograms are somewhat smaller in the VBA simulation than in the Excel calculations. The correct ones are the VBA numbers based on the plate theory. In the Excel calculation the number of theoretic plates *N* is assumed to be independent of the retention factor, but in reality the peak width depends on the retention factor by a factor sqrt(*k*/(*k*+1)). This is normally not observed in experimental chromatograms because extra-column broadening affects the early eluting narrow peaks more than later eluting broader peaks, thus compensating the theoretical effect. For late eluting peaks the widths may also be broader because of the nonlinear fit though.

The peak retention times for the same peak are slightly different for some of the peaks. This is because the simulation is only an approximation and the error in the approximation depends on the number of theoretic plates used in the simulation. The error is inverse proportional to the number of plates *N*, but unfortunately the CPC-time for the simulation depends on the number of plates squared N^2. For N = 200 in the simulation the error is ~1 % and the simulation of 11 analytes and *tmax* = 40 minutes takes about 10 s on my PC. You may change the number of plates used in the simulation in the sheet *Macro input* in cells A2 and/or L2. Try N = 100 and 500 in addition to 200 and see the effect (*N* max is 1000 at present).

The number of plates in the simulation is *NOT* identical to the number of theoretical plates *N* in cell B21 in the sheet *GC.* Actually the simulation is performed with relative few plates (cell A2 and L2 in *Fortran input*) and then the simulated peak widths are corrected by a factor sqrt(*N(simulation)/N(real)*).

For peaks eluted after the end time of a temperature program i.e. if there is an isotherm hold at the final temperature, the retention times calculated by Excel are wrong and the VBA simulated retention times are again correct. This is because the retention times here cannot be calculated by the Linear Solvent Strength model, so I have made a homemade approximation in Excel, and this approximation is obviously not very good – so if you need accurate results here you can only trust the simulations.

The table values for the columns and the compounds in the sheets *Compounds* and *Columns* are taken from literature and the references are given in sheets.

3) Complex temperature programs

You can simulate almost any temperature program by the simulation, but then you need to modify the sheet *Macro input*, and the simulation will no longer automatically simulate the same program as given in sheet *LC*. If you want to simulate a complex program, please save the spreadsheet under a new name by *Save As…* before proceeding – then you can always reload the original Excel sheet for normal operation.

1) Save the spreadsheet under a new name, *Save As…*.
2) Unprotect the sheet *Macro input* (Go to *Review* and click *Unprotect sheet*)
3) Input the times for the program in increasing order in the cells starting in E2 and proceeding down to cells E3-E4-etc. and the corresponding temperatures in the cells starting in F2.
4) Run the subroutine by pressing the *Update* button in sheet *LC.*
5) Note that the parameters and the plot of the gradient in the sheet *LC* do NOT reflect the gradient used in the simulation. You must use the data in the sheet *Macro input* if you want to plot the gradient.