Using the Berkeley Madonna Language

1. A Short Guide to Berkeley Madonna

Computer Requirements

Berkeley Madonna is available for PCs with Windows and for the Apple Computers. The book was prepared with Berkeley Madonna Version 8.3.18 running on Windows. Presently, a newer version 10 is available for Windows and Apple. More information with downloads can be found on the following website:

- www.berkeleymadonna.com

From this website you can also download **Berkeley Madonna User's Guide** for both versions. Here, you will find a detailed description of Berkeley Madonna.

.

Running Programs

To our knowledge, Madonna is by far the easiest simulation software to use, as can be seen on the Screenshot Guide in this Appendix for versions 8 and 10. This guide is made to cover issues most important for this book.

Under the Madonna Help, Examples, you will find a file called 5 Minute Tutorial.mmd. You should definitely go through this if Madonna is new for you.

All simulation examples are found on the Wiley Websites for both presently available versions of Madonna and can be freely downloaded from there.

Running an example typically involves the following steps:

- Open Berkeley Madonna and open a prepared program file.
- Go to Model/Equations on the menu and study the equations and program logic.

- Adjust the font and size to suit by first moving to the program listing and using commands in the Edit drop-down menu.
- Go to Parameters/Parameter Window on the menu and see how the values are set. They may be different than in the program. Those with a * can be reset to the original values. Also, if necessary, here the integration method and its parameters (DT, STOPTIME, DTMAX, DTMIN, TOLERANCE, etc.) values can be changed.
- Decide which plot might be interesting, based on the discussion in the text.
- You can also split the windows to simultaneously see two graphs.
- Go to Graph/New Window and then Graph/Choose Variables to select data for each axis. All calculated results on the left side of the equations are available and can be selected. Alternatively, you can also double click on the drawing area to add or remove variables for recording. The axes scaling can be modified by double clicking on the axis numbering.
- Run the program by clicking on Run or Green button with right arrow.
- Adjust the graph by setting the legend with the legend button. Perhaps put one of the variables on the right side of the graph with Graph/Choose Variables.
- Possibly select the range of the axes with Graph/Axis Settings. Choose colors or line types with the buttons.
- Decide on further runs. It is most common to want to compare runs for different values of the parameters. This is usually done with Parameters /Batch Runs and also with Parameters/Define Sliders. If the overlay button is set then more than one set of runs can be graphed on top of the first set. Sometimes more than one parameter needs to be set; this is best done with changes done in the Parameters/Parameter Window, with an overlay graph if desired.
- If there are too many curves to compare in one graph, you can also split the graph window to display two graphs simultaneously. Here one graph can be locked and the other used to make alternative runs.

As seen at the end of the Screenshot Guide, Parameter Plot runs are very useful to display the steady-state values as a function of the values of one parameter. For this, one needs to be sure that the STOPTIME is sufficient to reach steady-state for all the runs.

When running a program with arrays, as found in the finite-differenced examples, the X axis can be set with [i] and the Y axes with the variables of interest. The resulting graph is a plot of the variable values at the STOPTIME in all of the array sections. For equal-sized segments, this is the equivalent of a plot of the variables versus distance. If the steady-state has been reached then the graph gives the steady-state profile with distance. More on running programs is found in Sec. 2 of the Appendix.

Special Programming Tips

Berkeley Madonna, like all programming languages, has certain functions and characteristics that are worth noting and that do not appear elsewhere in this book.

Editing text

The very convenient built-in editor is usually satisfactory. Also the program can be written with a Word processor and saved as a text-only file with the suffix ".mmd". Madonna can then open it directly.

Finding programming errors. Look at a table output of the variables

Sometimes programs do not run because of errors in the program that cause integration problems. Some hint as to the location of the error can often be found by making an output table of all the calculated variables. This is done by going to Graph/New Window and then Graph/Choose Variables and selecting all the variables. Then the program is run and the table button is chosen. Inspection of all the values in the table during the first one or two time intervals will usually lead to an isolation of the problem for those values that are marked in red with NAN (not a number). Also, values going negative can be found easily here and often indicate an integration error. Sometimes this can be overcome with a limit function of the form, limit X>=0 that should, however, applied with care since it may hide program errors.

An experienced user's hint is that some occurring errors lead to occasionally difficult tracing of errors. Such typical errors are mix-up of the characters + and - or * and /. These will not lead to formal errors that would be directly alerted by Madonna. The resulting numerical solutions will then be either way off or sometimes numerical aberrations are not so easily seen.

Is a bracket missing?

Madonna tests for bracket pairs, and a missing bracket will be indicated.

Setting the axes. Watch the range of values.

Remember that each Y axis can have only one range of values. This means that you must choose the ranges so that similarly sized values are located on the same axis. In some cases, a log scale may be useful to show values of different order of magnitude simultaneously.

Are there bugs or imperfections in Madonna?

Yes, there might be some that have not yet been discovered. You may find some or you may have some special wishes for improvements. The Madonna developers in Berkeley, California would be glad to receive your suggestions. See their homepage to contact them.

Making a pulse input to a process.

This can be done it two ways: Either use the pre-programmed PULSE function (see the program CSTRPULSE for an example) or use an IF-THEN-ELSE statement to turn a stream on and off.

Making a more complex conditional control of a program.

In general, the IF-THEN-ELSE conditional statement form is used, combined with the inequality, and, or possibilities as found in the HELP. This can involve a switching from one equation to another within this statement. Another way is to use flags or constants that take values of 0 or 1 and are multiplied by terms in the equations to achieve the desired results. Nesting of multiple IF statements is possible:

```
V=IF(Disk<1 AND P>1.9)
        THEN 0.85*KV*P/SQRT(TR+273)
                ELSE IF (Disk<1 AND P<=1.9 AND P>1.1)
       THEN KV*P/SQRT(TR+273)*SQRT(1+(1/P)*(1/P))
```

Parameter estimation to fit parameters to data.

For fitting sets of data to one or more parameters the data can be imported as a text file and fitted by going to Parameters/Curve Fit. The Edit/Preferences/Graph Window provides the possibility of having the data as open circles.

Optimisation of a variable.

There is optimization available under Parameters/Optimize, but if it is something simple with one or two parameters, then sliders can also be effectively used. If the value of a maximum is sought as a function of a single parameter value, then the Parameter Plot for maximum value can be used.

Finding the influence of two parameters on the steady state?

A Parameter Plot choosing the “final” value can be used to find the influence of one variable on the steady state. The second parameter can be changed in the Parameter Window and additional parametric runs made and plotted with an overlay plot. Thus, it is possible to obtain a sort of contour plot with a series of curves for values of the second parameter. Unfortunately, no automatic contour plot is yet possible.

Nice looking results are not always correct.

A warning! It is possible to obtain results from a program that at first glance seems OK. Always make sure that the same results are obtained when DT is reduced by a factor of 10 or when a different integration method is used. Plotting all the variables may reveal oscillations that indicate integration errors. These may not be detectable on plots of a few variables.

Setting the integration method and its parameters?

It is recommended to choose the automatic step-size method AUTO and to set equal values of DT and DTMAX. Run the integration once and reduce both parameters by one-half and run again. If the results are good, try to improve the speed by increasing both parameters. Finally, it should be possible to set DTMAX higher than DT, but sometimes the resulting curves are not smooth if DTMAX is too high. In most cases, good results are obtained with AUTO and DT set to about 1/1000 of the smallest time constant.

If no success is found with AUTO, then try STIFF and adjust by the same procedure. Oscillations can sometimes be seen by zooming in on a graph; often these are a sign of integration problems. Sometimes some variables look OK but others oscillate, so look at all of them if problems arise. Unfortunately, there is no perfect recipe, but fortunately Madonna is very fast, so the trial-and-error method usually works out.

Checking results by mass balance

For continuous processes, checking the steady-state results is very useful. Algebraic equations for this can be added to the program, such that both sides became equal at steady state. For batch systems, all the initial mass must equal all the final mass, not always in mols but in kg. Expressed in mols the stoichiometry must be satisfied.

What is a “Floating point exception”?

This error message comes up when something does not calculate correctly, such as dividing by zero. This is a common error that occurs when equations contain a variable in the denominator that is initially zero. Often it is possible to add a very small number to it, so that the denominator is never exactly zero. These cases can usually be located by outputting a table of all the variables.

Plotting variables with distance and time.

Stagewise and finite-differenced models involve changes with time and distance. When the model is written in array form the variable can be plotted as a function of the array index. This is done by choosing an index variable for the Y axis and the [ ] symbol for the X-axis. The last value calculated is used in the plot, which means that if the steady-state has been reached then it is a steady-state profile with distance. An example is given in the “Screenshot Guide” in Sec. 2 of the Short Guide to Berkeley Madonna on the Wiley Websites and in the example CELLDIFF.

Writing your own plug-in functions or integration methods.

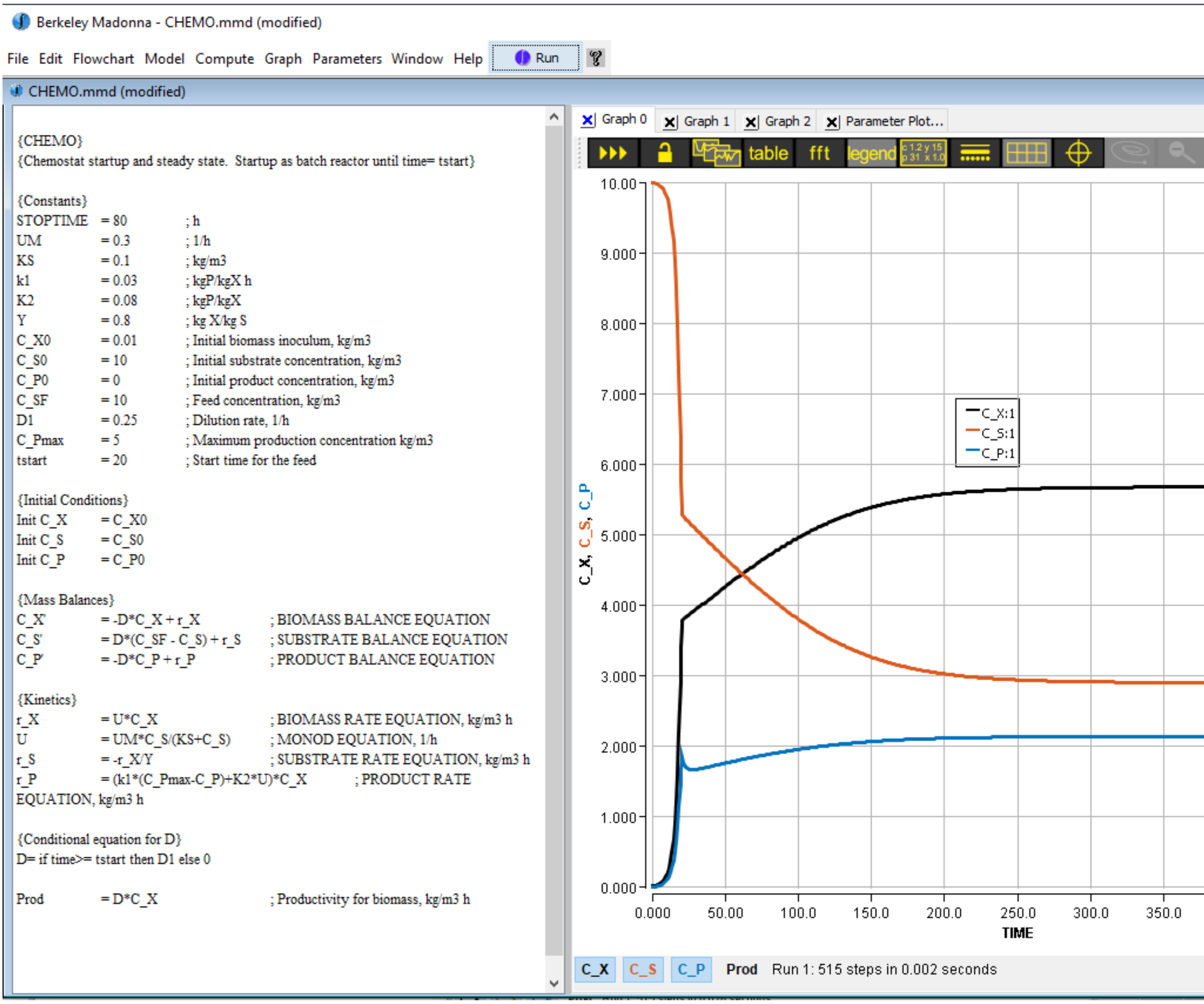
Information on using C or C++ for this can be obtained by making contact through the Berkeley Madonna homepage.

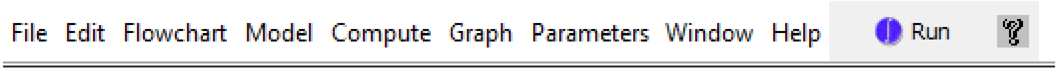
2. Screenshot Guide to Berkeley Madonna, Version 10

Working with downloads from Wiley Websites.

It is recommended to download a single example and then open it from the already started Berkeley Madonna Version 10. The Berkeley Madonna Version 10 *.zip file package requires unpacking first after downloading.

This guide is intended as a supplementary introduction to Berkeley Madonna, Version 10.2





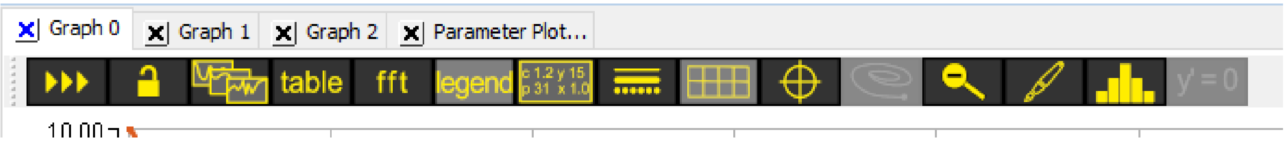
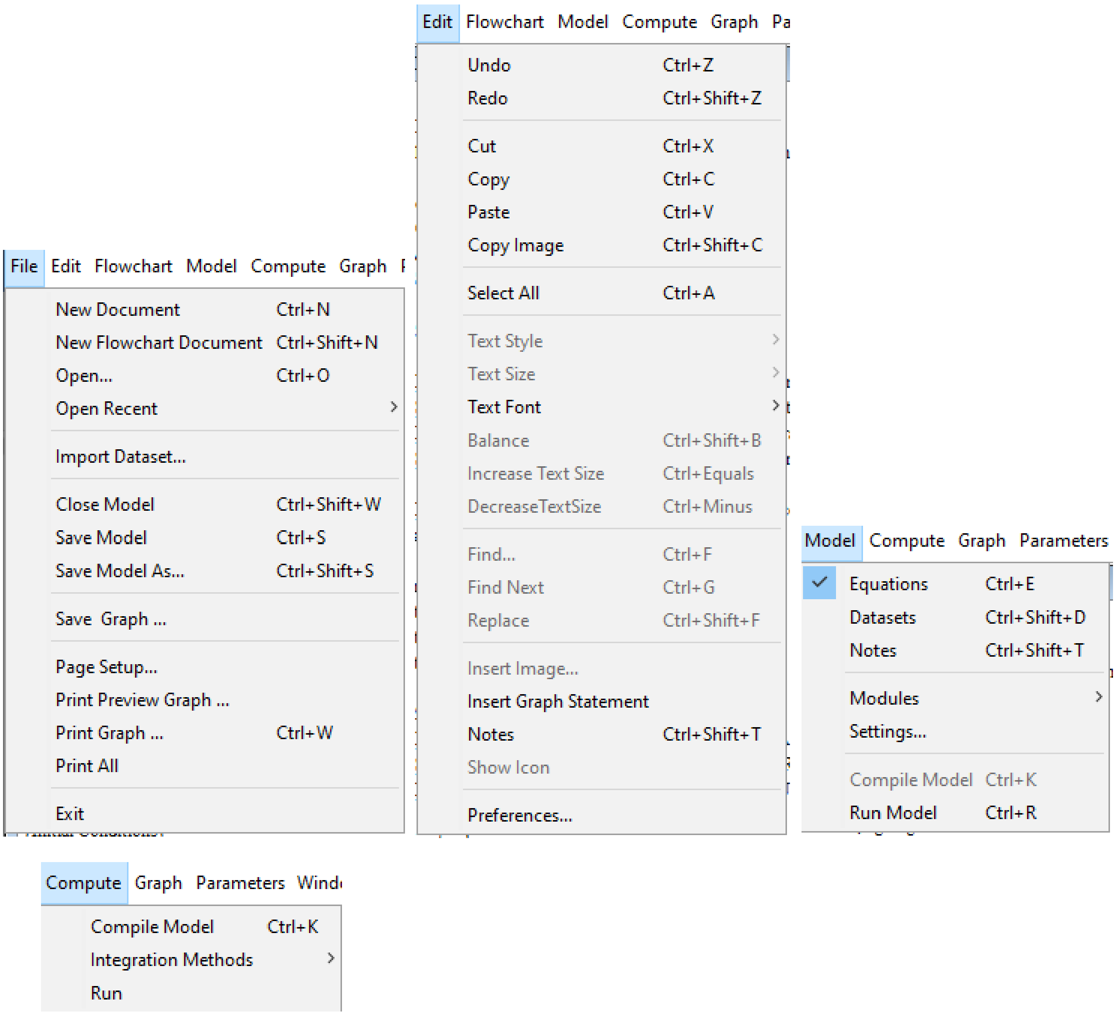


Figure 1. The example CHEMO has been opened and directly shows this screen with a subwindow for the program listing, one for graphics and one showing parameter values (right).

The Menu (From left: File, Edit, Flowchart active only for flowchart programs, Model, Compute, Graph, Parameters, Window and Help)

Graph Windows and Buttons (From left: Run, Lock, Overlay, Table, Fast Fourier Transform, Legend, Graph Parameters, Line appearance, Grid, Readout, Initial Conditions, Zoom out, Note, Histogram and Nullclines).

Not used in the book: Fast Fourier Transform, Initial Conditions and Nullclines.



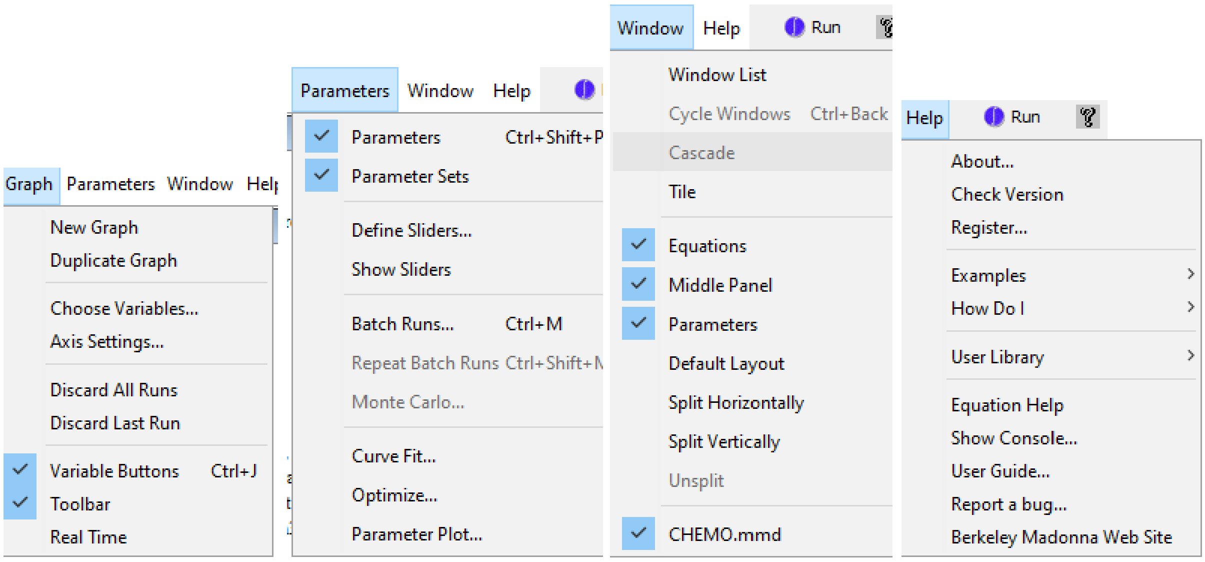


Figure 2. The Berkeley Madonna menus are shown above. The RUN button allows running a program.

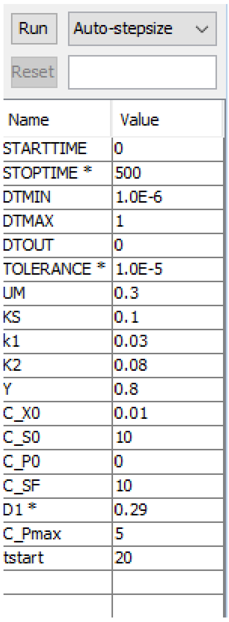


Figure 3. Seen here is the **Parameter Window**. Parameters can be changed here without changing the program listing.

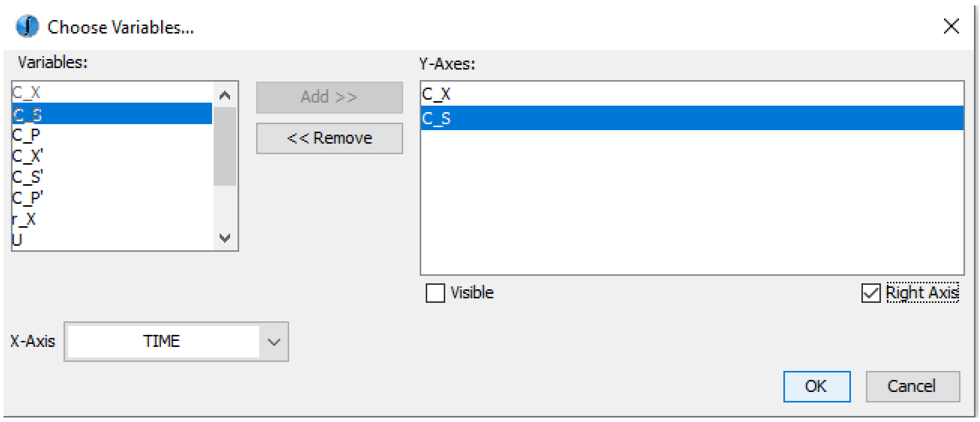


Figure 4. If a new graph is chosen under **Graph/New Window** then the data must be selected under **Graph/Choose Variables.**

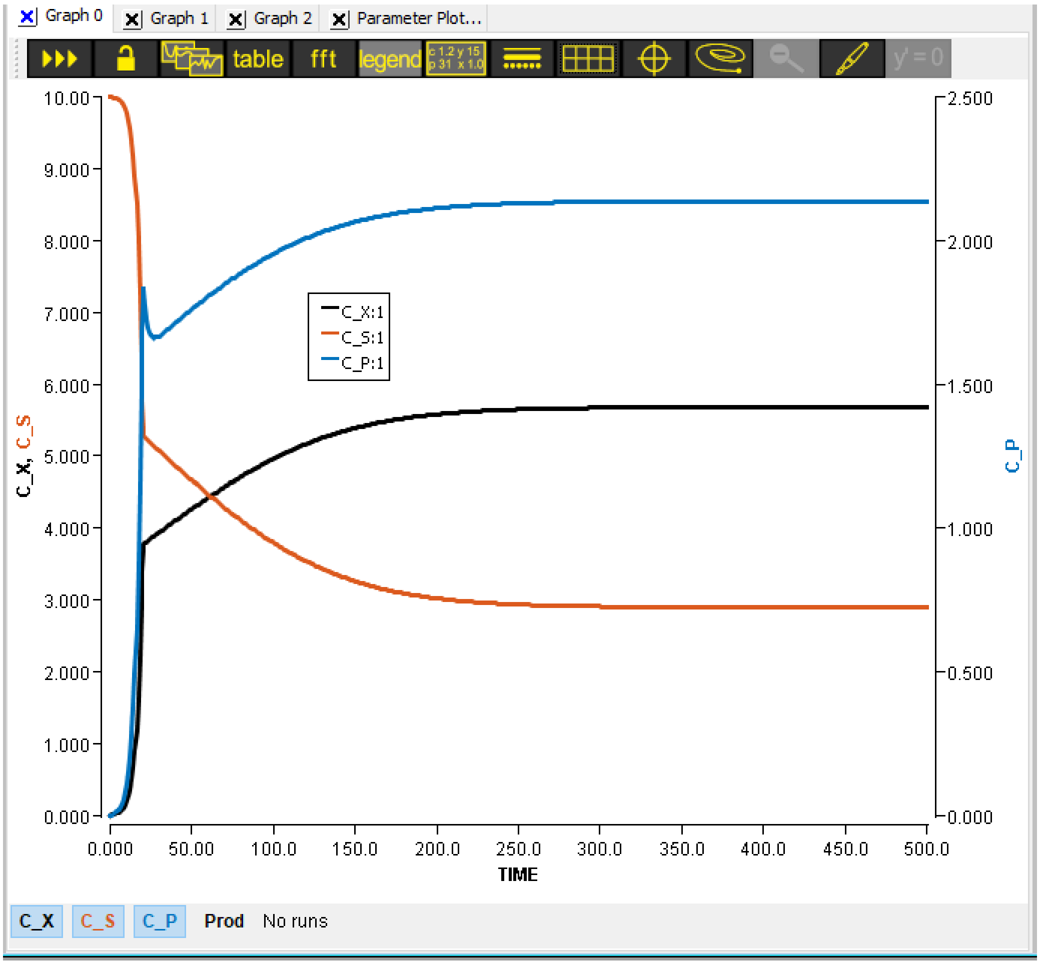


Figure 5. A graph window for variables on the left and right-side Y axis with Legend button selected.

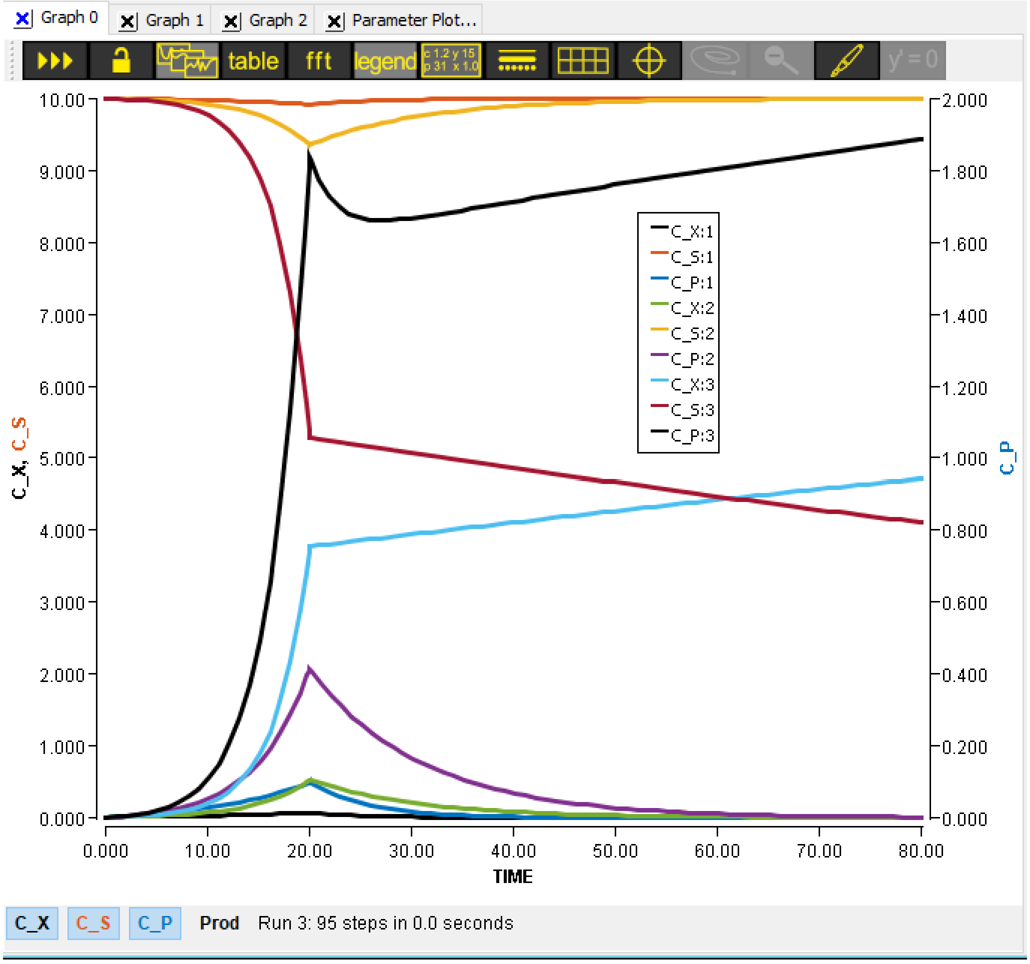


Figure 6. An **overlay graph** for three values of D1 as selected in the Parameter Window.

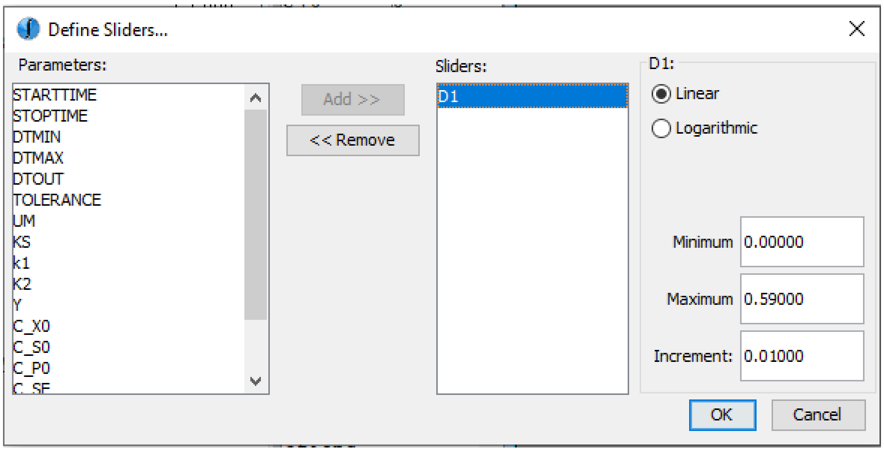


Figure 7. Part of the window to define the sliders.

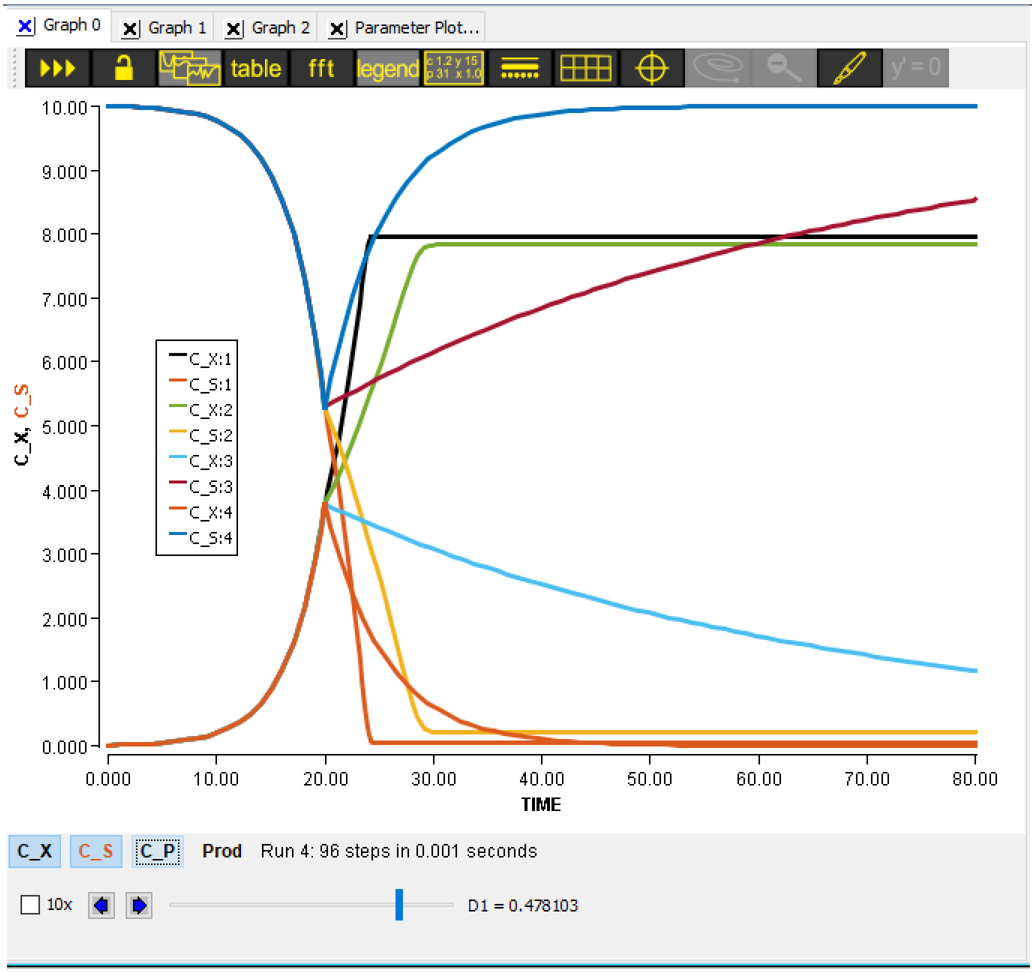


Figure 8. A graph of four slider runs, showing the Graph Parameters activated.

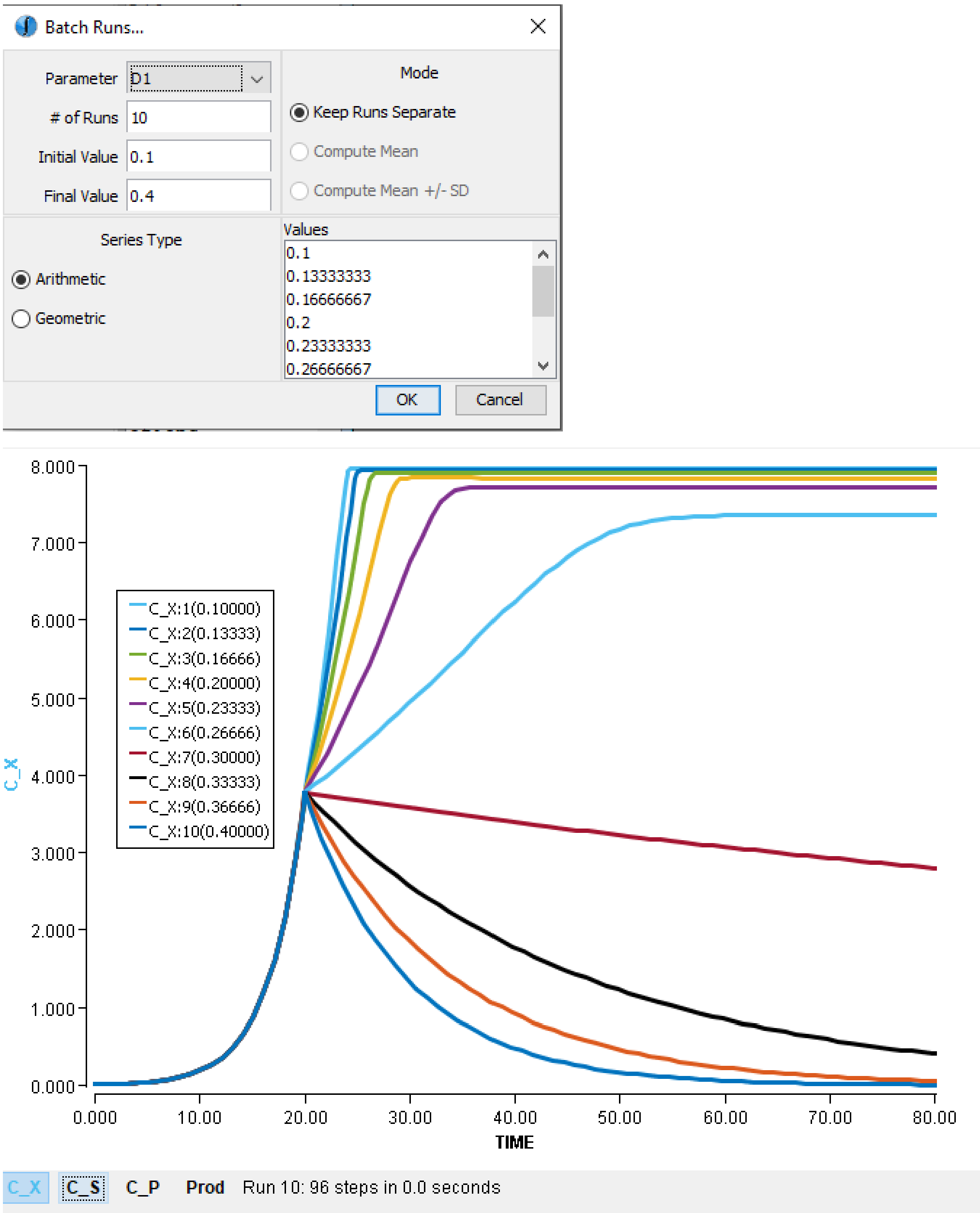


Figure 9. The Batch Runs window for 10 values of D1.

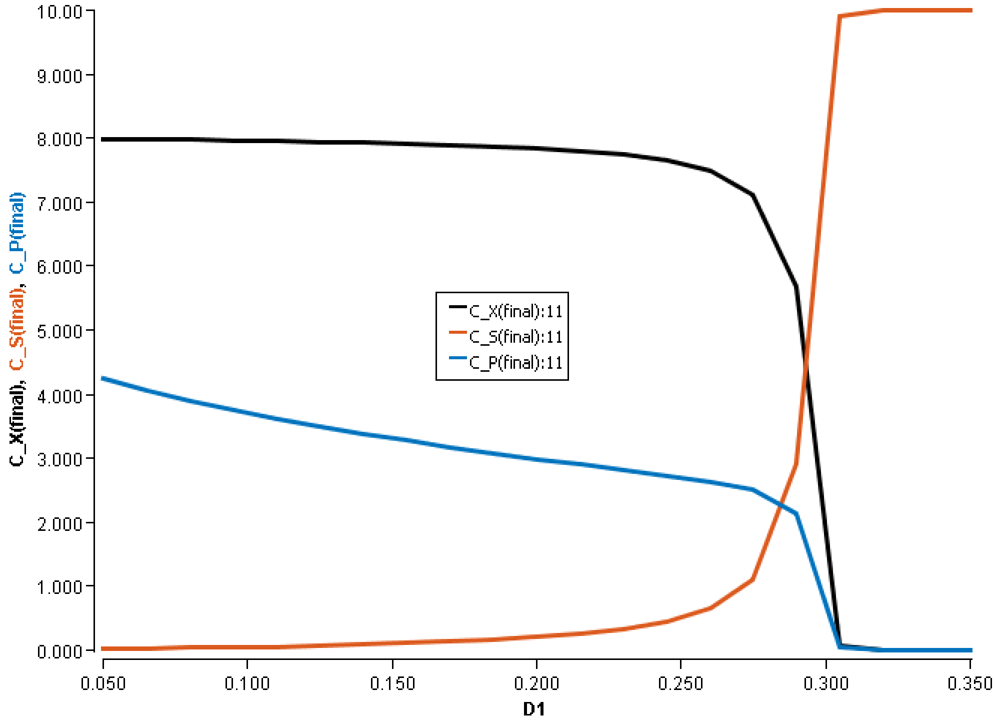


Figure 10. A **Parametric Plot** was chosen for 40 runs changing values of D1 to give the final, steady-state values. Selecting a thin graph line and choosing Data Points allows the individual values to be seen.

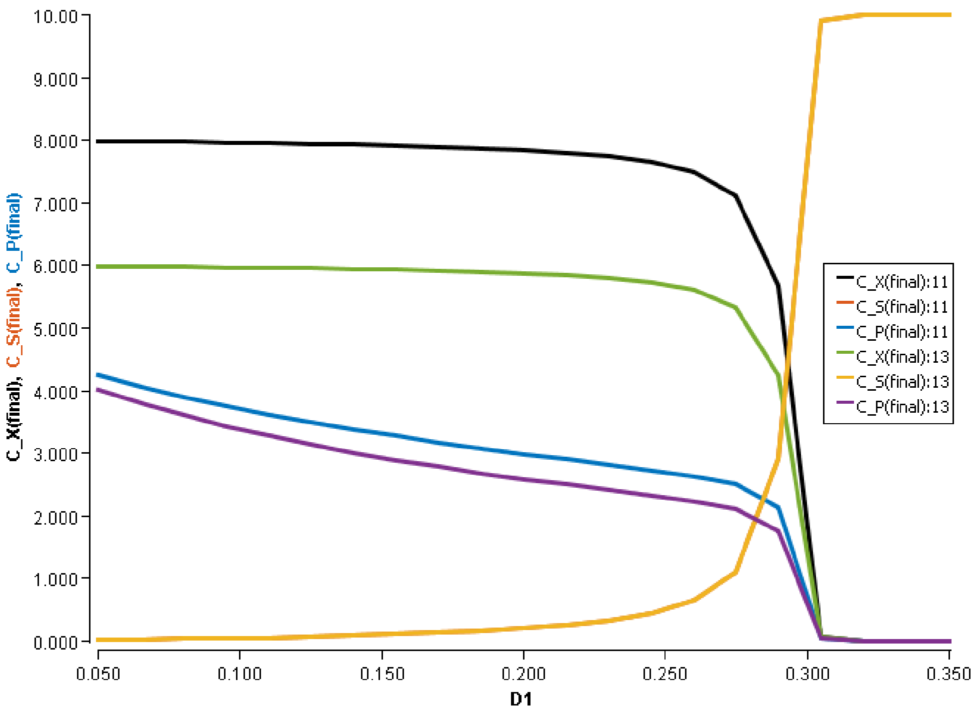


Figure 11. Two Parameter Plots overlaid showing the effect of reducing Y from 0.8 to 0.6.

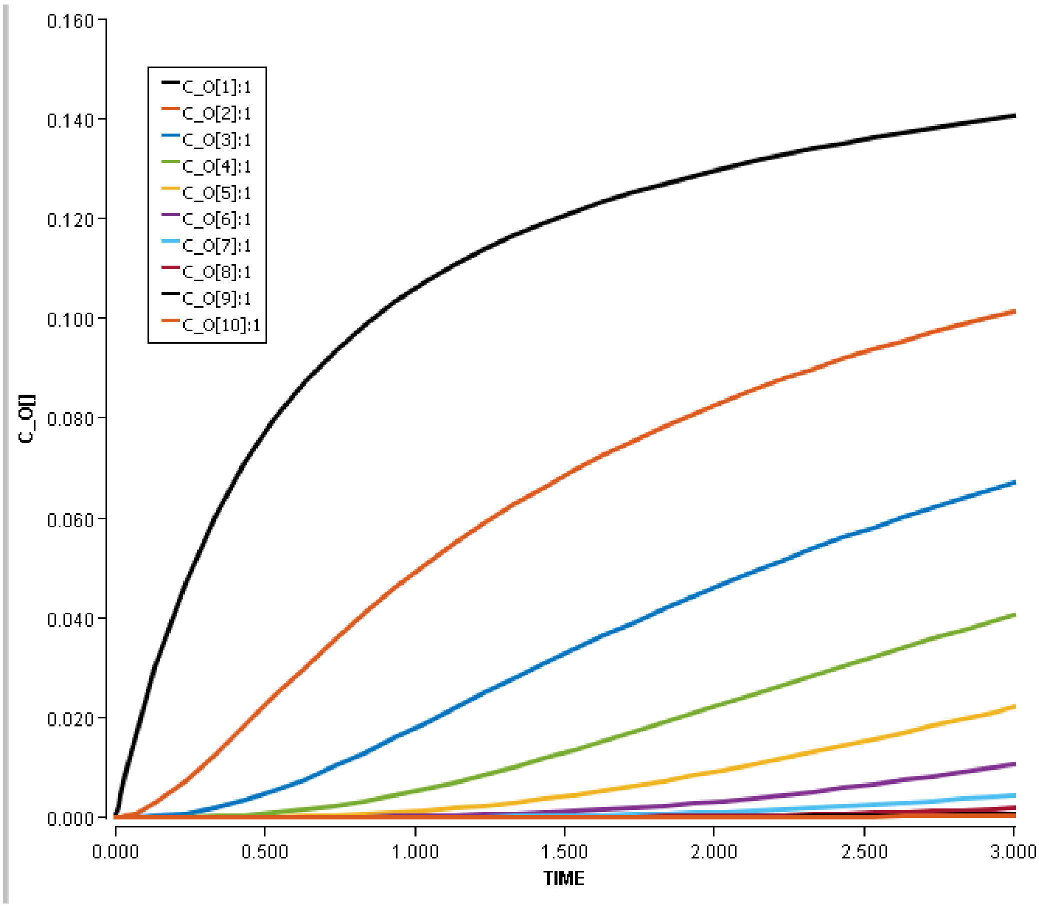


Figure 12. A program written in an array form allows plotting all the values versus time by choosing the variable vector, here S[ ] versus TIME for the program CELLDIFF.

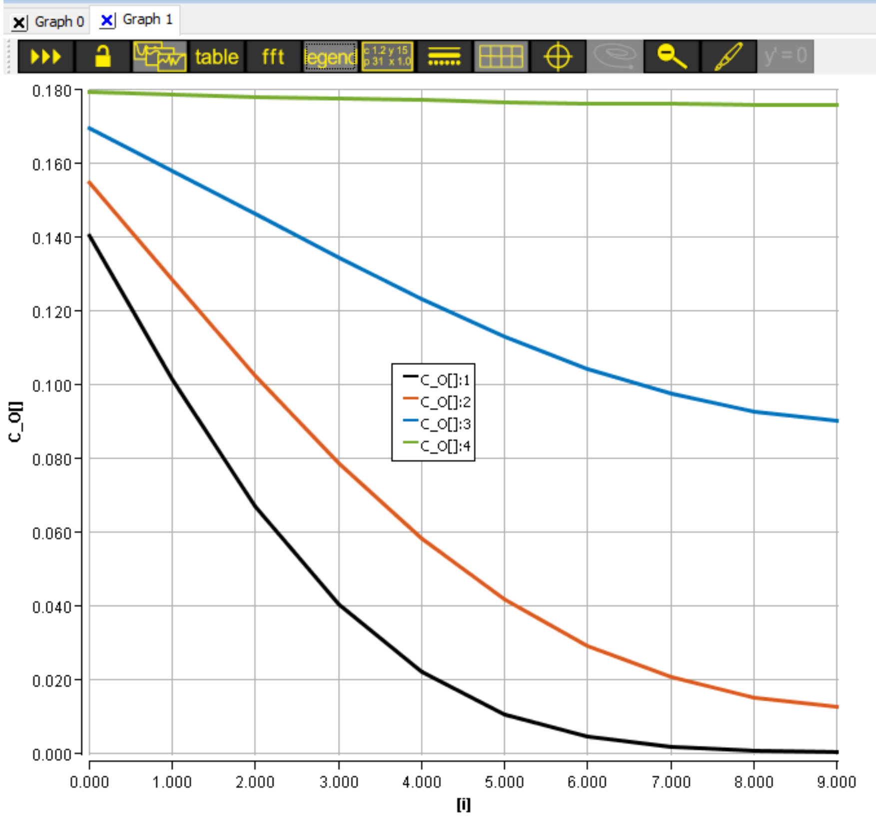
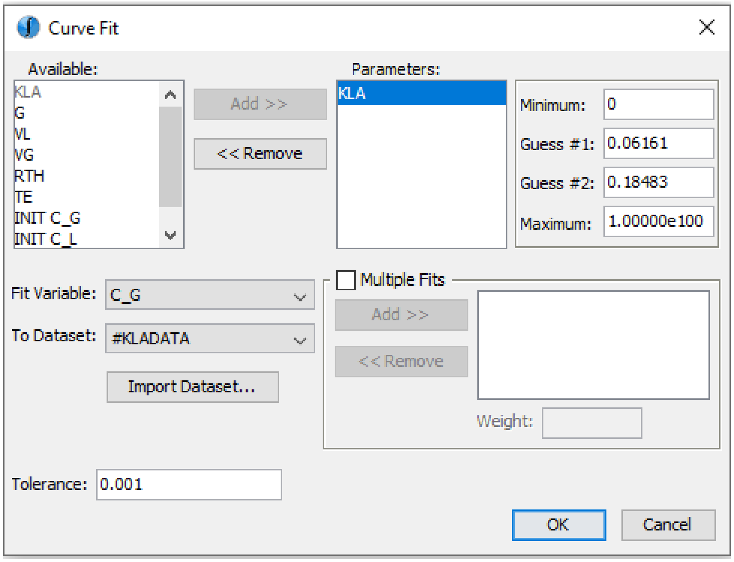


Figure 13. From the same program as Fig. 12, radial profiles of three runs are plotted in an overlay plot. The [i] values can be selected in the Choose Variables. Here the Radius has been changed from 0.4 (1) to 0.1 (4) to demonstrate the large influence of diffusion length.



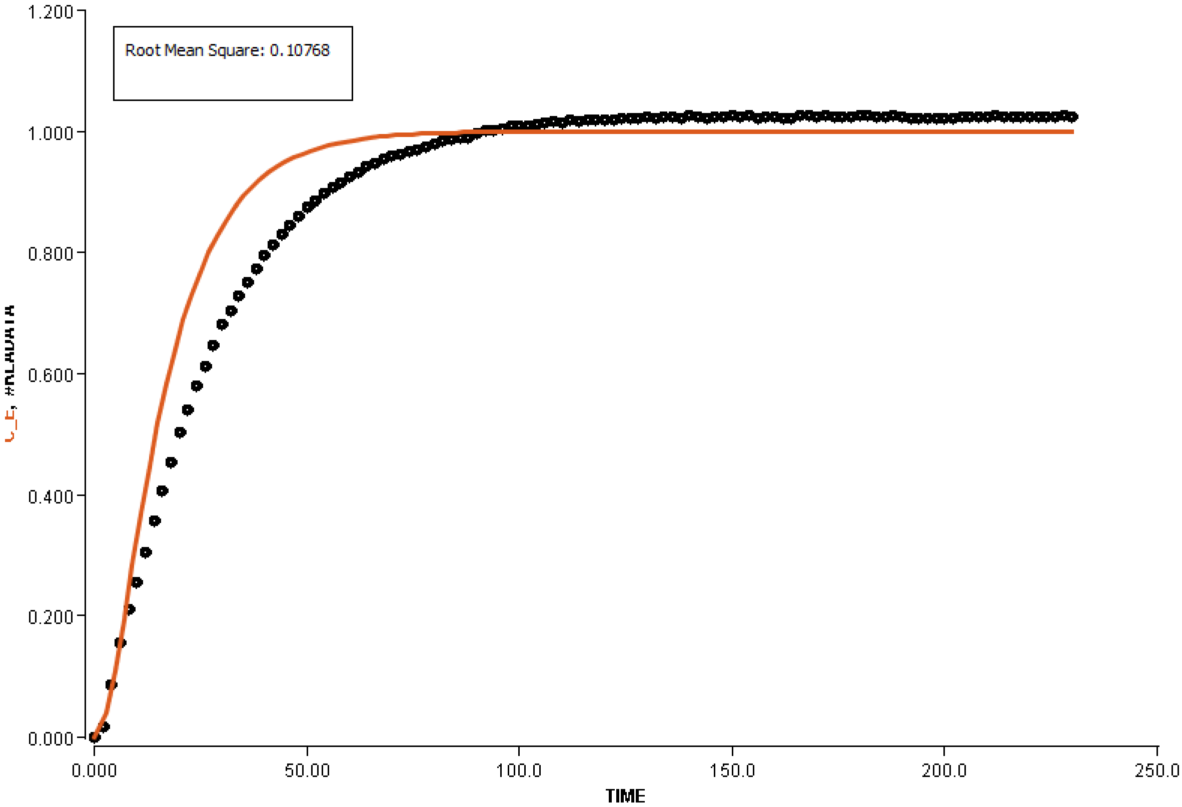


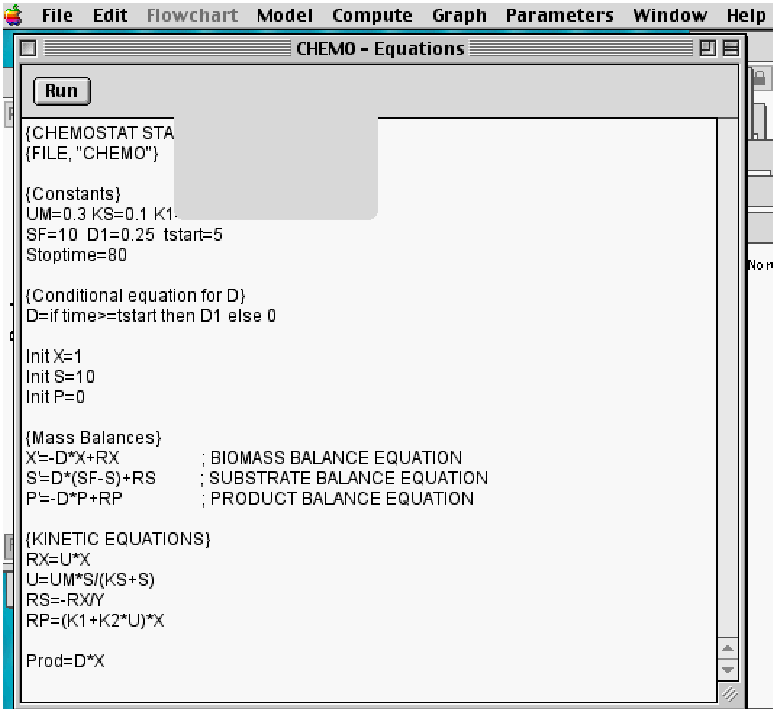
Figure 14. Here the program file KLAFIT is run and fitted to data in the text file KLADATA. The data consists of two columns: time and CE at equal intervals as seen by the Data Points on the plot. Note that the fit variable is CE and the parameter varied to minimize the difference in least squares is KLA.

3. Screenshot Guide to Berkeley Madonna, Version 8

Working with downloads from Wiley Websites.

It is recommended to download a single example and then open it from the already started Berkeley Madonna Version 8. The Berkeley Madonna Version 8 *.zip file package requires unpacking first after downloading. If you have both Madonna versions 8 and 10 installed on your Windows computer, clicking on the simulation files icon will always open it in Version 10. If you want to use Version 8, e.g., for making optimizations, open the simulation file from within Berkeley Madonna Version 8.

This guide is intended as a supplementary introduction to Berkeley Madonna, Version 8.3.18



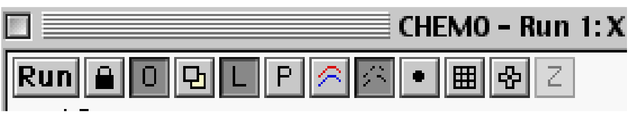


Figure 15. The example CHEMO has been opened and the Menu (From left: File, Edit, Flowchart active only for flowchart programs, Model, Compute, Graph, Parameters, Window and Help) and Graph Buttons (From left: Run, Lock, Overlay, Table, Legend, Parameters, Colors, Dashed Lines, Data Points, Grid, Value Output and Zoom

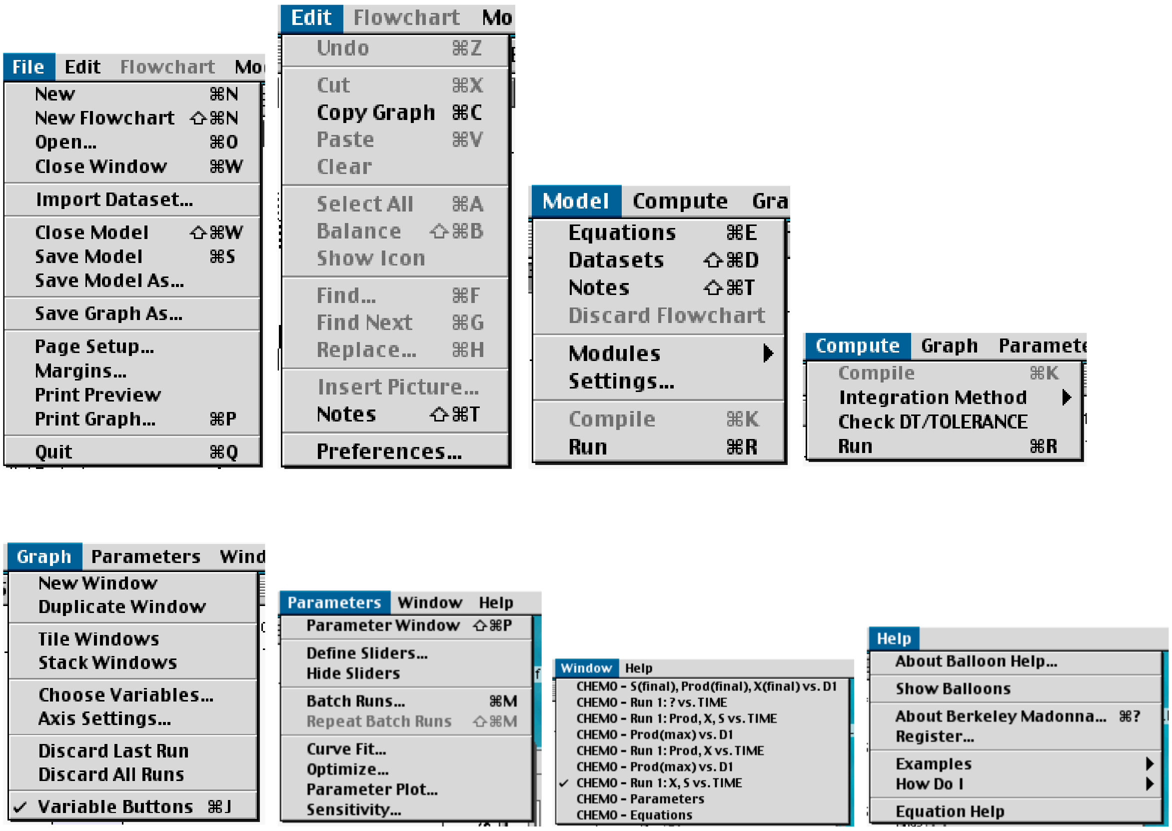


Figure 16. The Berkeley Madonna menus are shown above.

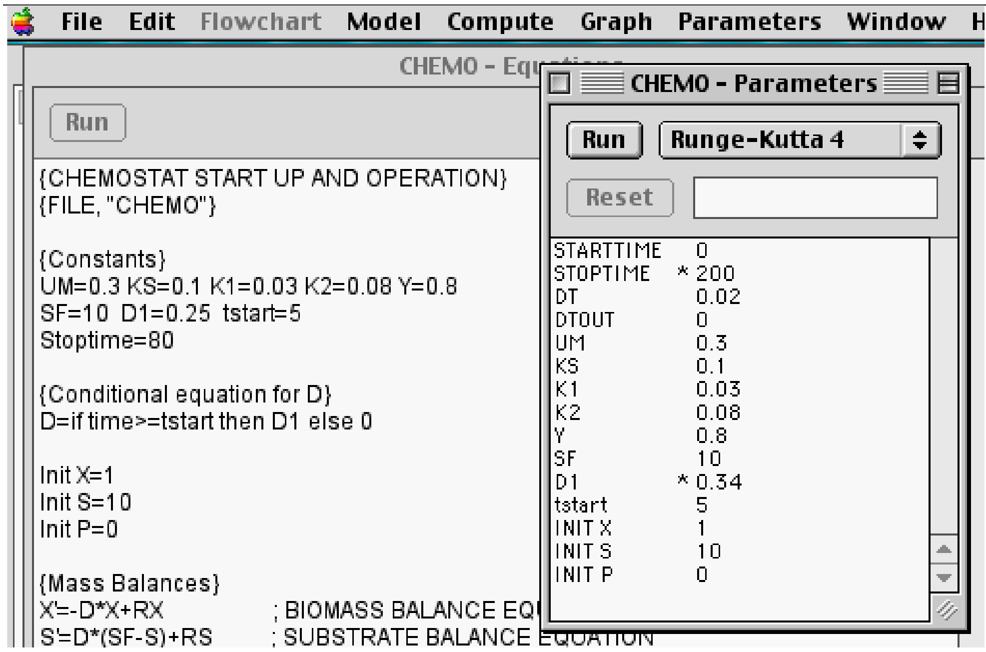


Figure 17. The **Model/Equations** was chosen. Seen here is also the **Parameter Window**.

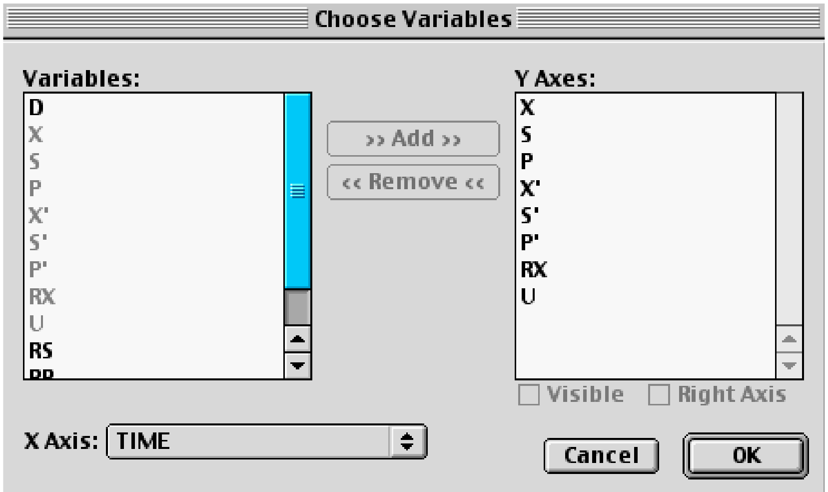


Figure 18. If a new graph is chosen under **Graph/New Window** then the data must be selected under **Graph/Choose Variables.**

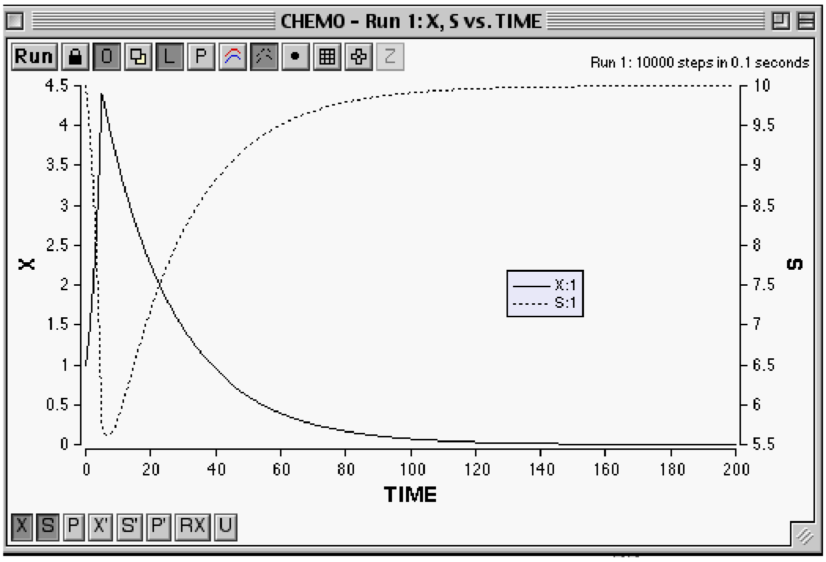


Figure 19. A graph window for variables on the left and right-side Y axis with Legend button selected.

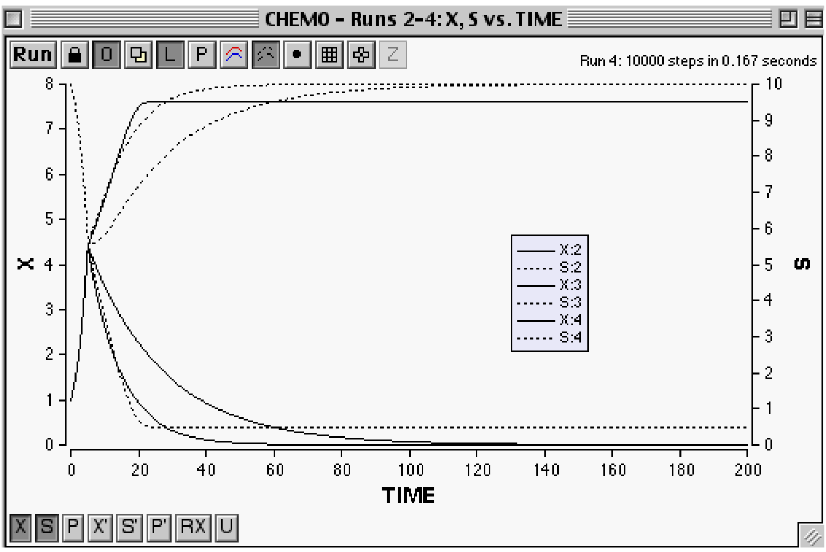


Figure 20. An **overlay graph** for three values of D1 as selected in the Parameter Window.

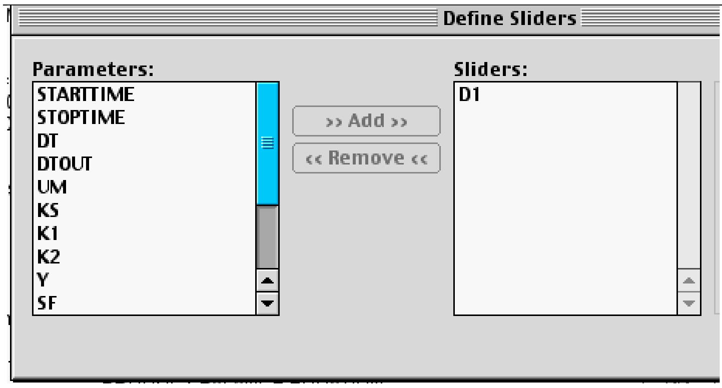


Figure 21. Part of the window to define the sliders.

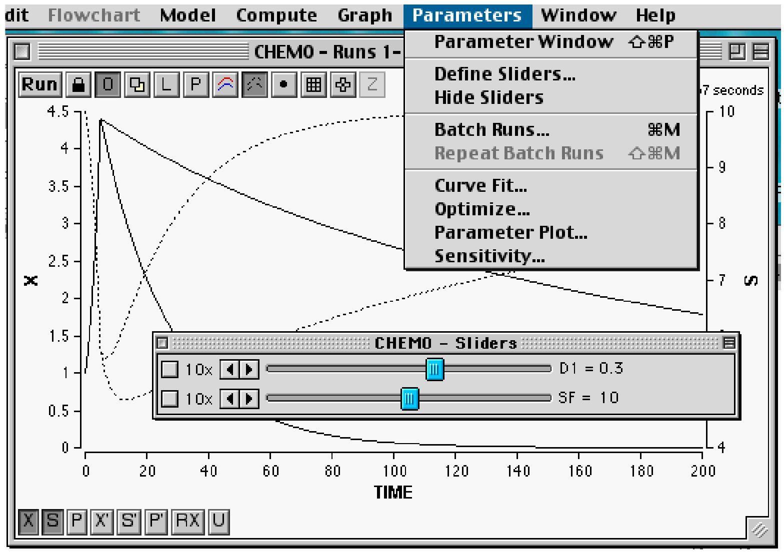


Figure 22. A graph of two slider runs, showing the Parameters Menu pulled down.

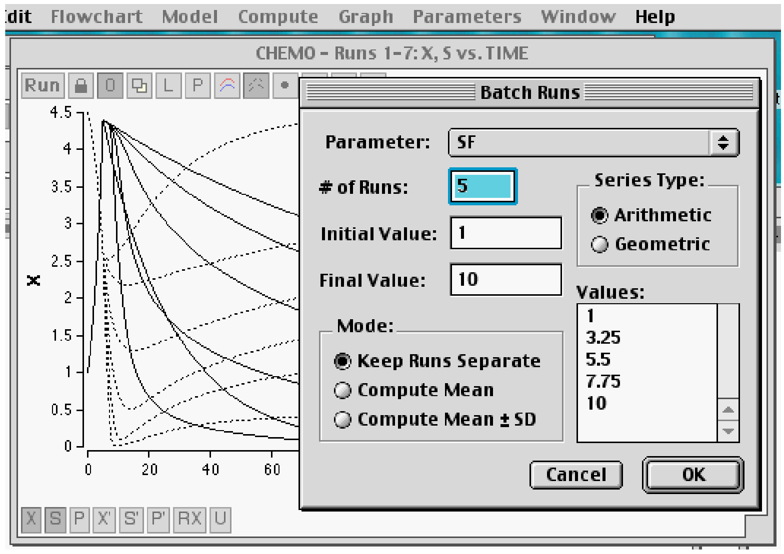


Figure 23. The Batch Runs window for 5 values of SF.

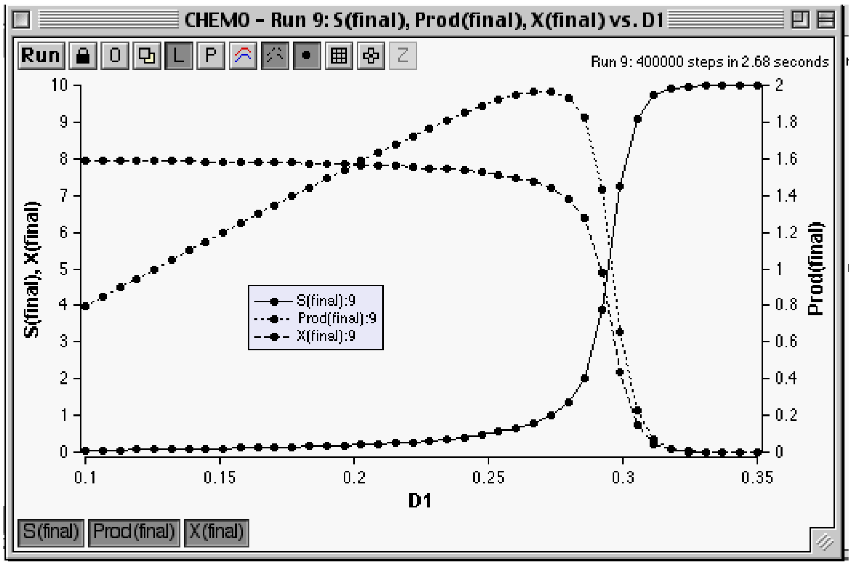


Figure 24. A **Parametric Plot** was chosen for 40 runs changing values of D1 to give the final, steady-state values.

The **Data Button** was pressed to give the points for each run.

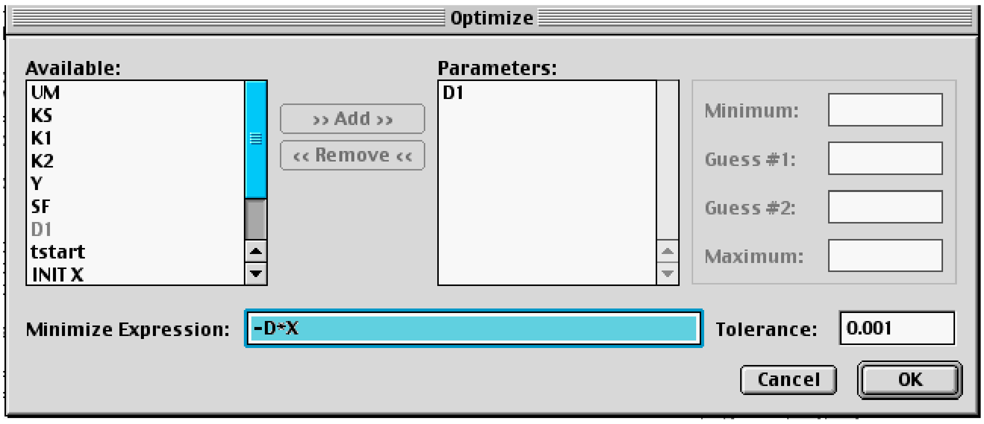


Figure 25. The Optimize Window, with the value of D1 being selected to minimize the expression -D*X. The value found was 0.27, corresponding to the Parameter Plot results.

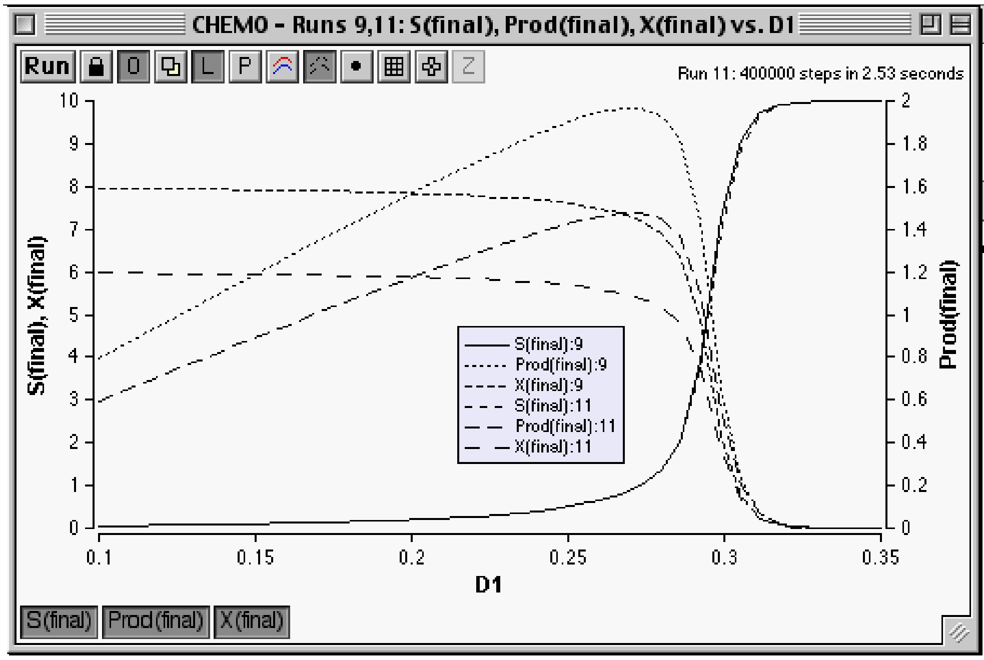


Figure 26. Two Parameter Plots overlaid showing the effect of reducing Y from 0.8 to 0.6.

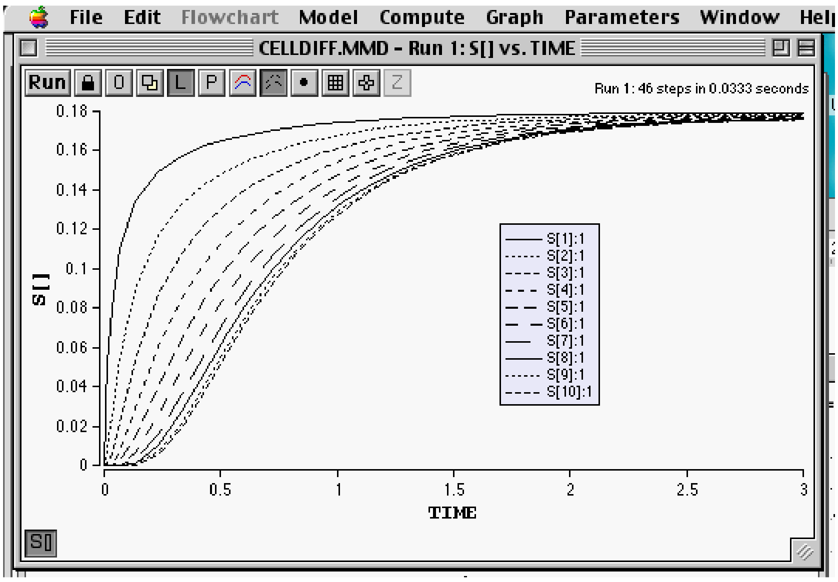


Figure 27. A program written in an array form allows plotting all the values versus time by choosing the variable vector, here S[ ] versus TIME for the program CELLDIFF.

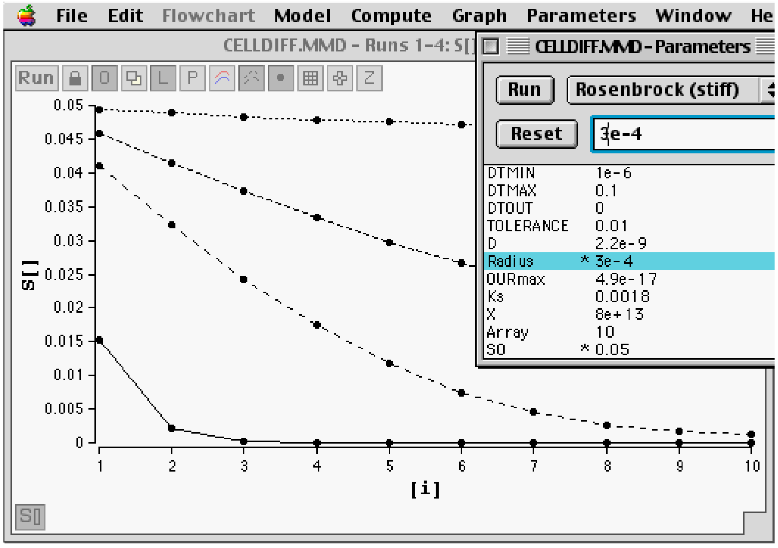


Figure 28. From the same program as Fig. 27, radial profiles of three runs are plotted in an overlay plot. The [i] values can be selected in the Choose Variables. Here the Radius has been changed to demonstrate the large influence of diffusion length.

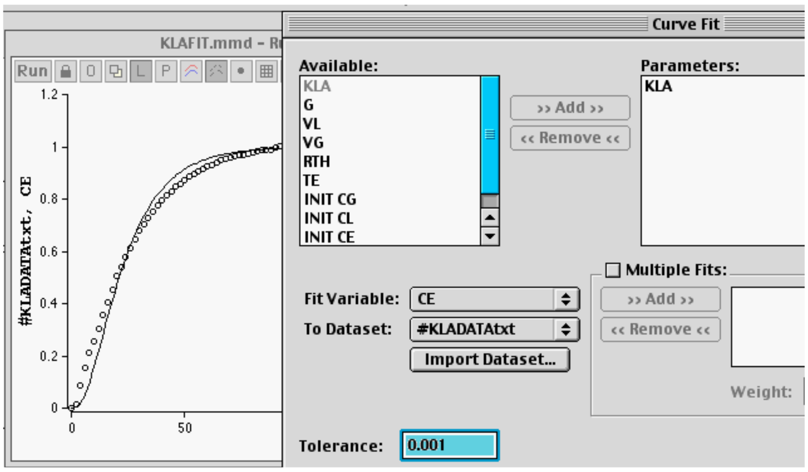


Figure 29. Here the program file KLAFIT is run and fitted to data in the text file KLADATA. The data consists of two columns: time and CE at equal intervals as seen by the open circles on the plot. Note that the fit variable is CE and the parameter varied to minimize the difference in least squares is KLA.