Instructions for setting up and running the demo of scanlsf least-squares spectral fitting program.

Linux FBC Version
Ed Berry 2009-12

Installing the program

Download the programs and demo for windows from: https://sourceforge.net/projects/scanedit/
http://sb20.lbl.gov/berry/scanlsf/

Install the demo by untarring scanlsf.tgz to a location on your hard drive or USB stick .(The size is about 700kB). For example:

cd
tar –xzvf scanlsf.tgz

This generates a directory "scanlsf" with three subdirectories, `bin/`, `speclib/`, and `demo/`. (If you download the entire package with source code, the top directory contains three subdirectories: src/ doc/ and demo/; this "demo" is scanlsf above and includes the 3 directories mentioned

To uninstall simply delete the directory scanlsf and all its contents (Installing and running the program does not affect the registry or list of programs in start menu)

The demo involves fitting a set of spectra of purified cyt bc1 complex obtained after adding different amounts of different reducing agent to reach different stages of partial reduction of the cytochromes. If you are in a hurry, just cd to the scanlsf/demo (or OSW-x.x.x/demo/demo) directory and "./ademofb.csh". Double-clidking the icon in a windows filemanager like gnome or kde should also work) This runs the demonstration. To understand what is going on, read the description below.

There are several versions of the suite depending on your platform and tastes. The most completely functional and highly tested is the original set compiled with MS QuickBasic. These are MS-DOS programs, but they work quite well in Windows (at least up to XP), and some drag-and-drop functionality has been added. However they must run in full-screen mode, and the mouse is not functional. Furthermore since they use the Microsoft proprietary runtime libraries, they are not open-source and cannot be licensed under the Gnu GPL license.
Ports are in progress to FreeBasic (www.freebasic.net), allowing to run in a normal window under Windows or Linux and be licensed entirely under the Gnu GPL license (www.opensource.org/licenses/gpl-2.0.php). Although not extensively tested, the spectral fitting routine is now completely functional. It is also being implemented in MS Visual Basic, allowing it to run in Windows and utilize GUIs and dropdown menus in the user interface. Thus this tutorial is provided in four versions to accommodate the slightly different user interfaces:

Windows:

DemoQB.pdf – Quick-Basic compiled programs
DemoWFBC.pdf – free-basic compiled programs
DemoVB.pdf – Visual Basic program (all in one)

Linux:

DemoLFBC.pdf – free-basic compiled for Linux

D. FreeBasic compiled programs for linux.

Getting familiar with the program and running the demo

1. Examine the standard basis spectra.
First examine the standard spectra used for fitting the experimental spectra. These are in the `scanlsf/speclib/bfbcallo.mat` file.

Open a shell and cd to that directory (cd /wherever/scanlsf/speclib).
Execute the scanedit program with the full path of the .mat file as argument:

../bin/scaneditfb $PWD/bfbcallo.mat

(if you get an error about missing lib, see below)

A graphics window opens up and displays the spectra. The blue spectrum which is positive everywhere is the oxidized bc_1 complex. The sharper spectra that go negative and hence may get clipped in the default display are difference spectra of cytochromes present in the bc_1 complex (as well as cytochrome aa_3 which is not present except as a contaminant). The assumption is that any spectrum of the bc1 complex in any redox state can be fit by a linear combination of these spectra. If the complex is fully reduced, only the first spectrum will be required. If any cytochromes are partially oxidized, an appropriate amount of each difference spectrum will be subtracted from the reduced spectrum to fit the experimental one.

To get a better view, expand the scale and allow negative values. Below the display is a numerical menu, items are selected by entering the digit 1-8. Digit 9 always displays the next screen of menu, cycling back to the first after the last. Hitting the space bar returns immediately to the first menu. Hence commands are series of digits, like 94 (set vert display scale). Type 9 then 4 (no <enter>). Enter 1000 for full scale range (units are mAU). Then enter 0 to put zero at midscale. "6PA" redraws the spectra at the new scale. (Currently that has to be uppercase PA, like some other things that are case sensitive, so its a good idea to set the caps lock on when you start the program.)
Now you have seen the standard spectra. "998" (quit) and go on to part 2, or if you want to play with the editor some more, don't quit but continue at "1. continued" further below.

2. Examine the experimental spectra.
Navigate up one level from the `scanlsf/speclib/` directory and down into the demo directory. Most of the files here are a series of spectra named beef2-n where n is 11-38.
../bin/scaneditfb $PWD/beef2-11
Drag `beef2-11` onto `scanedit` in the bin directory. The program will recognize from the -11 that this is part of a series, and it will keep loading successive spectra until it fails or fills all 30 traces. In this case there are only 28 spectra, so all are loaded. If desired adjust full scale absorbance and center value as in (1) above. As you can see, the three peaks of the different cytochromes appear independently due to the different redox potentials. Then enter 998 (quit) and go on to part 3.

3. Least squares fitting spectra.
3a. Run the ademofit.csh script from the command line.

cd to the /wherever/scanlsf/demo directory
./ademofit.csh
A graphics window will flash momentarily then disappear, and the first stage will go by in the shell window, listing coefficients required to fit each spectrum. Then the graphics window will reappear and the fit to the first spectrum will be plotted: blue is the experimental spectrum, green the best fit, and red the residuals. The screen autoscales so the experimental spectrum nearly fills the view, thus if the experimental spectrum is all far from zero the red residuals may be offscreen and invisible.

When in graphics screen, the program cannot accept input from the original stdin, so the graphics screen becomes interactive now. (If it doesn't then click in the graphics screen to give it focus). After plotting each fit the program pauses to let you examine. To continue hit enter to go to the next spectrum. To proceed automatically through the rest of the spectra with a 2-second delay for examination, enter X. The reason for examining each fit is to spot cases where the fit is bad and the results unreliable. You may notice that spectrum 23 is fit badly: This scan was made just after adding hydroquinone, and the spectrum was changing significantly during the scan.

When it finishes the results are saved in a .lft (leastsquares fit) file. This is then formatted and printed to a text file (.PRN) which can be opened in notepad or Word. Each row corresponds to one spectrum (but they will be numbered 1-28 instead of 11-38). Each column gives the concentration of each spectral component. The standard spectral components from the .mat file are listed below the table for a reminder.

Since version 0.1 the results are also saved in a .csv (comma-separated values) file which can be opened with MS Excel or compatible spreadsheet . Hence the .lft and .prn file are obsolete and may disapear in the future, together with the "dataedit" program whose sole purpose is to format the .lft file.

3b. Run the fitbbc.csh script from the command line (not yet set up).
cd to the /wherever/scanlsf/demo directory
and run the script (arguments are basename of the experimental spectra, first, and last numbers to process, separated by spaces or commas):

```
../bin/fitbbc.csh beef2 11 38
```

From here on everything goes as with the above demo. but here you could have fit different spectra, or by copying fitbbc.bat to fitcplab.bat and editing it to use chlorophyll standard spectra you could be fitting to different standards. In practice this is probably the most convenient way to run the program for routine analysis

3c. Run scanlsf from the command line.
This program is superficially like scanedit.exe but with different options and capabilities.

cd to the directory with the data ("demo" in this case) and invoke the program (you can put bin in your path to avoid typing its full path):
../bin/scanlsfb
select menu option 1 (fit spectra),
M (starting from matrix not LS inverse; this should be taken out since never used)
../speclib/bfbcallo.mat<enter> (path and name of standard spectra)
<enter> (unless you want to ignore part of the spectral range because offscale or something)
<enter> to use the default name and location for lsinv matrix (temp.lsi in current dir)
beef2,11,38 (same param as for script but now must be separated by commas)
<enter> for default- obsolete option
temp.lft (or any valid filename, for an unimportant temporary file)

Now it takes off and does pass 1, plotting the experimental spectra and calculating coefficients. At the end it asks if you want to calculate residuals. Always answer "Y".
Now it wants name for .lft file of results. say beef2.lft
(you might be fitting with the experimental spectra in this directory with several different choices for fitting spectra, so name result for fitting spectra)
Hit enter one more time to not save the residuals.

Now it starts pass 2, plotting spectrum, best fit, and residual. At the end of each it waits for input before going on, to give you time to examine the fit. If you enter "X", it waits 2 seconds between plots. If you don't enter X you can enter F on any one to make the "decomposition figure" showing the required amount of each standard spectrum and their sum compared to the experimental spectrum.

1. continued (more stuff in scanedit)-
Make a new spectrum which is the sum of the oxidized bc1 and the three reduced-minus-oxidized difference spectra, which should be equal to the spectrum of the fully reduced complex. Simple arithmetic operations are under menu item 6 (manipulate spectra).

When you hit 6 it will tell you the number of the next empty trace. Remember this so you can put the new spectrum in it. You will also see a list of options. Select 1 (add or subtract two spectra). Then you get a syntax hint: "n1=n2+n3, n1=n2-n3" n1 means the number of the trace to put it in (which can be one of the original traces if you want), n2 and n3 are the traces being added or subtracted, and +/- tells which. Type "8=1+2" and hit enter. Before plotting the result, it asks for a comment for the new spectrum. Say "bc1 with c1 reduced" if you want, or just hit enter to leave the comment blank. So the whole process was:
61<enter>8=1+2<enter>bc1 with c1 reduced<enter>.
now add the other two difference spectra one at a time:
61<enter>9=8+3<enter>bc1 with c1 and bH reduced<enter>.

61<enter>10=9+4<enter>bc1 fully reduced<enter>.

Lets save the last one for future use:
hit 7 (file spectrum), it asks you which trace, 10<enter>
it shows you the comment for 10 and asks for a filename,
Filename can be any 11 alphanumeric characters; if longer than 8 then the others will go in the extension. Don't put a dot; dash is OK. Say bovbc1red.

Now lets make a postscript figure from the standard spectra. This is 92 (plot on paper). First question, which traces? You can enter a range (separated by dash) or single trace. Say 1-4 to get the oxidized and three difference spectra from the original file, and <enter>. That's "1-4<enter>"
now add in trace 10, the fully reduced: "10<enter>"
hit <enter> one more time to indicate you're through.
On the next question enter 1 to cycle through the colors starting with color 1 (blue). Then hit <enter> 4 or 5 times until you see a lot of activity as it writes the traces to the file (another version of this routine lets you preview the figure onscreen, but that's not in here yet). Hit <enter> one more time at the question about the arrow, and it closes the file and tries to copy the file to LPT1:. This has not been set up for linux yet so probably makes a file LPT1, but by now it has already created the plot in "temp.ps" which you can send to a color printer (lpr temp.ps) or open in Illustrator or ghostscript. or convert to pdf with ps2pdf.

Contents (Additional files are for other platforms, and more will be generated when you run the demo. These are the files required for the FB-compiled Linux demo)**:**

BIN directory:
`scaneditfb` - spectrum viewer, editor, simple arithmetic operations on one or two spectra. (Not used in the demo, but for viewing the sample and standard spectra)
`scanlsfb` - many functions, including the Sternberg-et-al. least squares fitting algorithm.

SPECLIB directory:
`bfbcallo.mat` - standard basis spectra for fitting bc1 complex in different redox states

DEMO directory:
`ademofb.sh` - script to fit experimental spectra using bfbcallo.mat
`beef2-11` etc. experimental spectra to be fit

Reference:
Sternberg, J., Stillo, H. & Schwendeman, R. (1960). Spectrophotometric Analysis of Multicomponent Systems Using the Least Squares Method in Matrix Form. Analytical Chemistry 32, 84-90.

dynamic shared libraries-
The linux version is dynamically linked, so needs certain libraries which you may not have especially if using a 64-bit OS. However they can easily and reversibly be installed from online repositories using apt-get (ubuntu or debian?) ur yum (fedora or red-hat.

For ubuntu:
apt-get install gcc
apt-get install g++ (or gcc-c++ on Fedora)
apt-get install binutils-dev (or binutils-devel on Fedora, and binutils-static on Fedora 14 and later)
apt-get install perl
apt-get install autoconf
apt-get install libncurses-dev (or ncurses-devel on Fedora)
apt-get install libx11-dev (or libX11-devel on Fedora)
apt-get install libxext-dev (or libXext-devel on Fedora)
apt-get install libxpm-dev (or libXpm-devel on Fedora)
apt-get install libxrandr-dev (or libXrandr-devel on Fedora)
apt-get install libxrender-dev (or libXrender-devel on Fedora)
apt-get install libgpm-dev (or gpm-devel on Fedora)
apt-get install freeglut3-dev (or freeglut-devel on Fedora, or another one that pulls in OpenGL headers)

If using a 64 bit OS, you ma need to append .i386 or .i686 after each package-name to get the 32-bit libraries and dependancies.

Contact the programmer at:

eaberry at users.sourceforge.net,
eaberry1 at gmail.com