CDtool – A new program for CD spectral analysis

5.1 Introduction: CD spectral data processing and analysis

Rapid data collection is one of the advantages of CD spectral analysis of proteins. It is the subsequent data processing and analysis steps that can be surprisingly time consuming. Currently, several different software packages are required for processing and analysing the data. Large volumes of data are typically collected resulting in data management issues. After data collection the spectra are processed, stored and then analysed.

5.1.2 Existing Processing Tools

Several packages exist for processing CD spectra. Processing software currently in use in our group is super-3 (). This command line fortran-77 program runs off a VAX machine. This program incorporates all of the basic functions for processing a CD spectrum until it is scaled to $\Delta\varepsilon$ units. The main smoothing algorithm for the program is an implementation of the Savitsky-Golay smoothing algorithm. Add-on packages also exist for super-3 whereby error bars can be calculated and plotted.

Several of the CD machine manufacturers provide the machines with their own software. New Aviv machines are provided with the IgorCD program. This is a set of macros for processing CD spectra and thermal melts. It has all of the common functions such as subtracting averaging and smoothing. The program is able to read in data from a Microsoft excel file format and some older Aviv formats. Jasco machines have a software package called spectra manager. This

runs under the Windows OS and reads the output from several different Jasco instruments. Within this package several methods are available for smoothing, unit conversion, de-convolution and baseline correction functions.

5.1.3 Existing analysis tools

Several packages exist for analysing scaled CD spectra. In particular the CDPro software is a widely used. This consists of three popular analysis programs SELCON3, CDSSTR and CONTIN as well as the clustering program CLUSTER (). CDPro is able to accept the same input data file format for all four programs. The main alternative to this is the dichroweb server hosted at Birkbeck college (). This is basically a front end to the 3 main packages included in CDPro along with varslc () and k2d methods (). The site is able to accept a large variety of data formats and contains a considerable amount of online documentation and is well supported.

5.1.4 Data storage and retrieval

Relational databases have become increasingly important to many areas of biology for storing data. Several different varieties of database management systems (DBMS) exist for running relational databases. DBMS provide several desirable features that are distinct from an ordinary flat file system. Firstly, concurrency gives the ability of one ore more user to access the database at the same time.

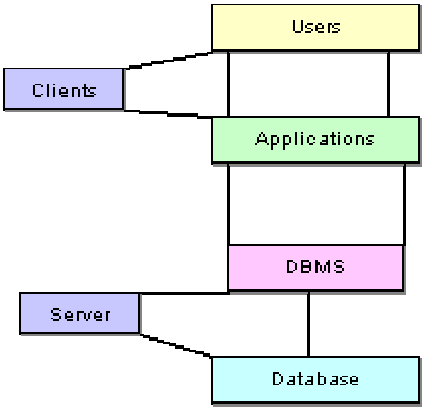


Figure 5.1: Diagram highlighting distinction between the client and server sides of a database architecture.

Compactness is provided as several different packages can access the same database. A relational database provides a dedicated structured query language (SQL), which is fast and flexible. SQL allows the user to describe the data they want to see from a database, and manipulate that data.

A relational database is principally made up of a set of tables and a number of relations between them. For modelling a database there are several important modelling concepts, including entities, instances of entities, attributes and relationships. An entity is a class that can be uniquely identified, for example a protein is an entity, whilst an instance of this is the protein model with PDB code 1c3w. This protein model then has a large number of attributes associated with it. Some of these attributes may uniquely identify the protein model, and are referred to as candidate keys. A candidate key can be selected as a primary key for a table. The primary key uniquely identifies each row of the table from each other and cannot occur more than once in a table, and may never take a NULL value. If a table has no such attribute suitable for the primary key one may be invented, and is referred to as a surrogate key. A foreign key is used to establish relations between different entities.

Normalisation is a method by which redundancy is removed, ensuring that data elements are stored only once. The main advantage of removing redundancy is improved data integrity. Normalisation consists of several steps involving splitting larger tables to smaller tables, so that all the attributes of an entity depend more and more on the key alone. Although transaction databases are good at

maintaining data integrity, they can only handle limited queries. For more complicated and novel queries a separate database is needed known as the query database. This contains greater redundancy than the deposition but as a result is more flexible in retrieving data.

Certain graphical notations exist for describing a database architecture. UML (Unified Modeling Language) is one standard used. Various software is available for implementing UML diagrams.

5.1.1 Aims

With the needs of our project in mind it was decided to write an application that would unify all aspects of CD data processing and analysis in an efficient way. Critically it should be distinct from existing software packages for processing CD data. As described above, packages exist for initial processing and smoothing. Also packages exist for secondary structure analysis and clustering. However, no package currently exists that contains all of these steps.

In view of the large amount of data generated from our project it became apparent that a relational database would be an advantage for data storage and retrieval. Our program should be able to access the database to store and retrieve data. This access should be available remotely from any computer. The prediction tools used should include the common matrix methods currently employed for secondary structure prediction. Where possible jobs should be automated to reduce user input to a minimum.

From theory it is known that the source of a CD spectra is not solely based on the secondary structures assigned by DSSP. In particular, structures such as a PP-II helix can often mask the spectra of other secondary structures (). Including a means by which secondary-structures most relevant to CD could be visualised was thought to be desirable.

The resulting program named *CDtool*, is a multi-platform GUI (Graphical User Interface) application for processing and analysing CD data. It contains all the necessary features from initial processing to storage of data and final structural prediction. The remainder of this chapter details the design steps taken, and describes the workings of the finished program.

5.2 Program design

5.2.1 Choosing a programming language

It was decided to use object-oriented (OO) programming. This style of programming allows encapsulation and separation of interface from implementation. Java and C++ are both examples of modern OO languages. In the end the Qt package was decided upon (). Qt has compilers for Windows, Linux, Mac and Unix systems, and the same code can be used for all platforms. Qt display, unlike Java, has a native look and feel to the platform it is running on. Qt has a large range of classes that includes many of those that are useful for this project. A series of SQL classes exist for database integration into a Qt application. Qt has a program called designer for drawing and creating the GUI. This program generates an XML file that can then be compiled directly into the code. Qt has a simple mechanism of signals and slots, for connecting up different objects within the program.

5.2.2 Program appearance

This has several advantages that were useful for our purposes. Separating different parts of the program into different tabs allows the program to be compartmentalized into different sections, with different tabs relating to different aspects CD. It also allows a large number of windows to be displayed on the same screen at once. This allows the user to refer to different aspects of the CD data at the same time.

5.2.3 Software libraries

5.2.3.1 Qwt library

The Qwt library () contains GUI components and utility classes for 2D-plotting widgets. QwtPlot is the plotting widget class and contains the majority of functions for plotting. Additionally the library provides means by which labels can be added to the plot. Functions for automatically rescaling are provided. A number of signals are provided from the classes to specify which object or part of the plot is being selected.

5.2.3.2 QwtPlot3D library

This is an OpenGL 3D plotting widget library for plotting 3-Dimensional data. It provides a comprehensive list of display modes, including wireframe, filled polygons, hidden line and floor projections of the 3D data. Scaling, rotating, shifting, zooming of data sets are all provided with the SurfacePlot class. Various pixmap (all Qt supported formats) and vector outputs (PostScript, EPS and PDF) are provided .

5.2.3.3 gMatVec library

gMatVec is a small C++ matrix/vector template library that contains routines for SVD. Data is stored in Mat and Vec classes for matrices and vectors respectively. The SVD class carries out SVD analysis from which U, W and V matrices can be obtained.

5.2.3.3 C Clustering library

The C lustering library is a collection of numerical routines that implement the most commonly used clustering algorithms. The library is written in ANSI C and so can be easily linked into C++ programs.

5.2.4 The class structure

Any successful OO program must have a well thought out class structure. A UML diagram of the main classes from *CDtool* illustrates the usefulness of an OO approach (figure)

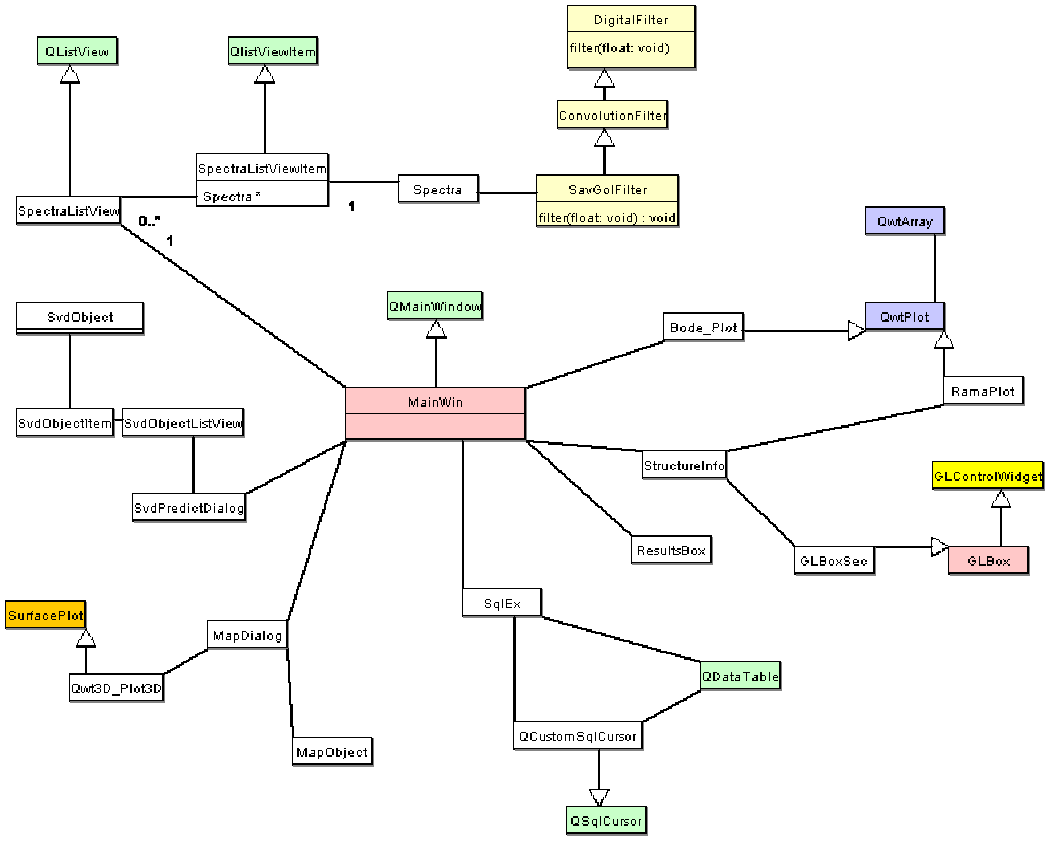


Table 5.2: UML diagram of the main classes for CDtool. Arrows signify that one class inherits from another. A line without an arrow signifies that one class contains the other. The numbers on the connecting lines detail the numerical relationship between classes. A (*) symbol indicates a 1 or more relationship. Boxes are coloured according to the library they are from.

5.2.4.1 Main window class

The MainWin is the top-level class that is first created on executing the program. It contains all of the tabbed interfaces that make up the program. It has a large number of signal to slot definitions connecting the top level toolbar to the corresponding tabs.

5.2.4.2 Spectra class

It was decided early on to create a Spectra class. This class encapsulates all of the essential features of a CD spectrum. This includes a range of wavelengths with the corresponding CD and HT signals. The Spectra class can smooth itself and the resulting data stored in a corresponding vector. It also stores all of the information about how it was collected. The vectors used in this class are from the standard template library (STL).

5.2.4.3 Savgol class

The Savgol class implements the Savitsky-Golay digital filter. The code for the algorithm was obtained from open source code at (http://www.taygeta.com/skip.html). The class runs the algorithm with a specified smoothing window dimensions and for a given polynomial order. In all cases for our data an even window and a 3rd order polynomial are used. The filter function carries out the smoothing via the pure virtual function in the DigitalFilter base class, allowing for the use of other digital filters in the future.

5.2.4.4 SpectraListViewItem

A class was necessary to allow different spectra objects to be selected through the GUI. The SpectraListView class was derived from the Qt Class ListViewItem. A list view item is capable of displaying itself in a QListView. The inherited class has a pointer to a spectra object as a protected member. In this way it allows the selection of an item in the list view to be coupled to selecting a spectra object.

5.2.4.5 SqlEx

The SqlEx Class contains functions for interacting with the database. In particular, functions for retrieving and scaling data are implemented. It has a data table that allows tables in the database to have entries inserted, deleted or edited. The SqlEx class keeps track of the changes and runs subsequent SQL queries to maintain the database integrity.

5.2.4.6 SvdObject and SvdObjectItem

The SvdObject class was designed to contain all of the information for a reference dataset. The spectra and structure matrices are contained in the class using gMatVec matrices. It can also print itself out to a file in a format that can be read in by another SvdObject. The SvdObject is accessed in a similar manner to the SpectraListViewObject, by using a derived class form the QlistViewItem class.

5.2.4.7 MapDialog

The MapDialog class has been developed fro rendering 3D plots as might be obtained from a time or temperature scan. The MapDialog class contains two important classes for it's functioning. The SurfacePlot class is able to represent 3D surfaces, allowing zooming and rotating of the image.

5.2.4.8 QCustomSqlCursor

QSqlCursor is a class that handles all the SQL commands in Qt. However this class is not able to take the standard SQL format how a user would type it out. To allow such custom SQL commands, a subclass, QSqlCustomCursor. Currently the virtual functions for sorting, selecting, updating, inserting haven't been re-implemented. However this should be important in future work to make the database more useable.

5.2.4.9 PdbModel and Residue classes

The PdbModel class currently stores the co-ordinate data of a PDB file in a QMap of Residue classes. The Residue objects are indexed by a unique number assigned for each of the Cα's of the PDB file. The Residue class contains information for each amino acid including various secondary structure assignments.

5.2.4.10 GLBoxSec

The GLBoxSec class has been inherited from the GLBox class from the Bbone library (). The GLBox class it is inherited from stores the PDB data using the BTL library (). However, the inherited class stores a pointer to a PdbModel. Also an alternative repaint function is implemented that allows only certain residues to be highlighted. In this way the GLBoxSec has been modified to be more useful for displaying secondary structures, and then relating these to the CD spectra.

5.2.4.11 BodePlot and RamaPlot

The RamaPlot and BodePlot classes are inherited from the QwtPlot class. The Bode class carries out most of the functioning for the BodePlot as does the StructureInfo class for the RamaPlot class. However the RamaPlot contains a pointer to a PdbModel object through which it is able to obtain the relevant data.

5.2 Database design

It was decided to use a relational database for our project and the popular MySQL DBMS was chosen for a number of reasons. MySQL is free and open source, and has relatively low system requirements compared to the DBMS Oracle. The perl DBI package () is also able to interface with MySQL facilitating any perl scripting to input data.

5.2.1 Tables

A UML diagram of the main tables of the database are presented (figure).

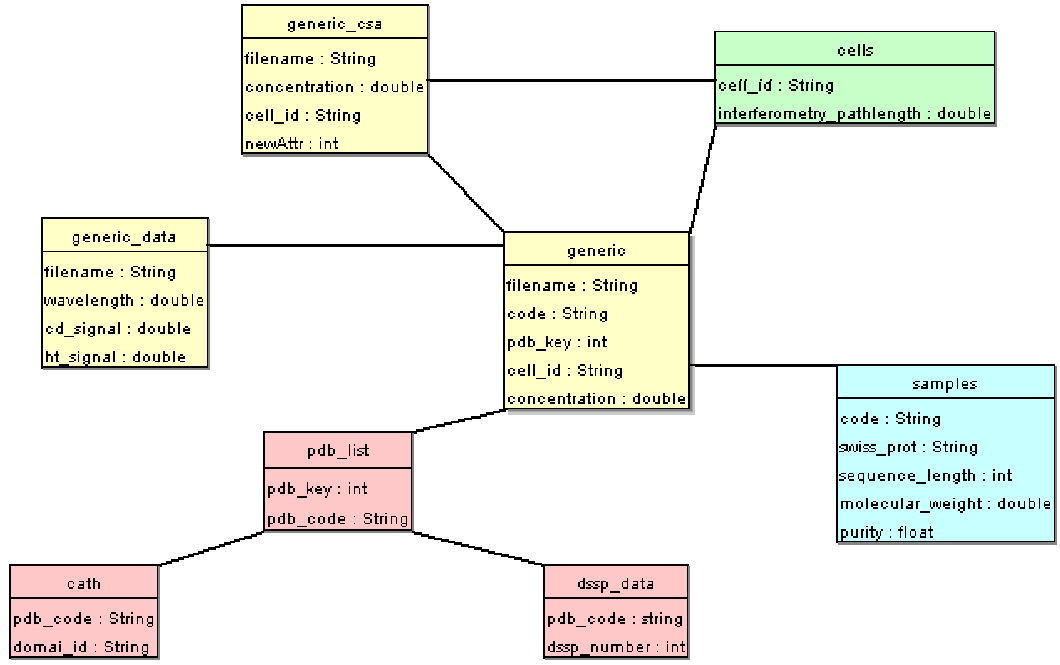


Figure 5.3: UML diagram of the main tables created for the database to store our data. Straight lines indicate tables that are connected by foreign keys. Important attributes are listed in the lower box. Attributes with a star next to them were used as the primary key.

5.2.1.1 The generic table

The 'generic' table contains the main details associated with each spectrum. The primary key for this table is the filename originally assigned to the spectrum. The experiment column is the name of the experiment, and contains the name of the protein and any other distinguishing details. The column titled description contains details about the experiment, the reasoning behind it and other general information. The notes column describes any relevant features noticed after the spectrum had been taken. The cut_off column records the low wavelength cut off of the data after which the data is no longer valid.

Column	**Data type**	**Description**
Filename	VARCHAR *	The filename of the spectra

Experiment	VARCHAR	The experiment name
Description	VARCHAR	Rational of why the spectra was collected
Temperature	INT	Temperature at which the experiment was taken.
Dwell time	FLOAT	The dwell time used for the experiment
Units	ENUM	The type of units the spectra is in
Machine	VARCHAR	The name of the machine the data was collected on
Cell_ID	VARCHAR	The ID of the cell the spectra was collected in
Code	VARCHAR	The code of the samples table
Pdb_ID	INT	Key linking the spectra to a PDB file
Zeroed_low	FLOAT	Low wavelength boundary for zeroing
Zeroed_high	FLOAT	High wavelength boundary for zeroing
Cut_off	FLOAT	The wavelength below which the data is of no use
Notes	VARCHAR	Notes on any unusual features noticed on data collection or processing

*Table 5.1: Column name, data type and descriptions for the generic table from the database. The data types are given based on the MySQL formalism. The symbol * denotes the attribute used as the primary key.*

5.2.1.2 The generic_data table

The wavelength data for each entry in the generic table is stored in the 'generic_data' table. The primary key for each entry is a composite key of the filename and the wavelength. Hence a spectra's data can be retrieved from its filename.

Column	**Data type**	**Description**

Filename	VARCHAR (CK)	The filename of the spectra
Wavelength	FLOAT (CK)	The wavelength of the data
CD_signal	FLOAT	The CD-signal of the data
HT_signal	FLOAT	The HT-signal of the data
CD_smoothed	FLOAT	The smoothed CD-signal data

Table 3.4: Column name, data type and descriptions for the generic_data table. The data types are given based on the MySQL formalism . CK labels which columns were used to make the composite key.

5.2.1.3 Samples table

A 'samples' table was created to record details of the protein sample used for a CD spectrum. The primary key for the samples table is a surrogate key given by the 'code' column. The 'generic' table contains the 'sample' table code as a foreign key. In this way a spectrum can be traced back to a given sample. This has a one to many relationship with the 'generic' table since a sample can have many different CD spectra, but a CD spectra will only have one sample. A sample entry has details specific to a given protein sample such as the purity given by the supplier. The table also has information about the molecular weight as measured by mass-spectrometry.

Column	**Data type**	**Description**
Code	VARCHAR (PK)	The unique code of the sample
Product_name	VARCHAR	The proteins name, as it was provided from the supplier
Source	VARCHAR	The source organism
Sequence_length	INT	The number of amino acids in the protein
Sequence_mw	FLOAT	The Molecular weight of the sequence (Da)

Purity_supplier	FLOAT	The purity as specified by the supplier
Purity_measure	FLOAT	The purity we have measured by SDS-electrophoresis
MW	FLOAT	The molecular weight measured by mass-spectrometry in Daltons

Table 5.3: Column name, data type and descriptions for the samples table from. The data types are given based on the MySQL formalism. PK labels which columns is used as the primary key.

5.2.1.3 Structure_data table

This table contains an entry for every C-alpha atom of all the PDB files in the database. Several columns for different secondary structure assignments were created. A unique value for each residue of a PDB file is assigned using the dssp_number attribute. The rows of the table have a composite key of the PDB code and the dssp_number attributes. The form of the table is straightforward and has different columns for each secondary structure. By storing the raw data from the secondary structure assignments we are more flexible in the final assignment method we use.

5.2.1.4 generic_csa table

The 'generic_csa' table contains information about the CSA calibration files. Information on the concentration, cell-ID, units and dwell time are collated here. The filename column is the primary key and is the foreign key to the 'generic' table, thus any spectra in the generic table can be calibrated to millidegrees. The spectra data for this table is contained in the generic_csa_data table.

5.2.1.5 Cells table

The cells table uses a surrogate key 'cell_id' as its primary key. The table contains pathlength information about the cell used. Specifically we have the pathlength specified by the manufacturer and that measured by interferometry.

The cell-ID is contained in the ‘generic’ and ‘generic_csa’ table as the foreign key to this table.

5.3 Using the program

5.3.1 Installation and licensing

CDtool version 1.0 is available for download from the cdtools.cryst.bbk.ac.uk website. Currently only a Windows version of the program is supported however Mac and Linux versions are planned in the near future. Installation is performed by downloading the setup wizard and going through the step by step instructions. This installation wizard was created using Inno-Setup version 3.0.6 ().

5.3.2 Protocol for usage

A typical protocol for using *CDtool* is described below. Three or more scans for the sample and baseline is advised for data collection. After loading the data the CD-smoothed signal is checked for the successive scans. If it appears that the signal is unstable over time, then the spectrum should be rescanned. The samples are averaged as are the baselines. The CD spectra of the baselines should match up around 263 and 270nm. The samples can then be baseline subtracted and zeroed between 263-270nm. If the samples and baselines do not match up then there may have been an error in data collection and the samples or baselines will need to be collected again. However if there is a good explanation to the lack of overlap, such as a signal due to aromatics, then the baselines should be subtracted but not zeroed. If a PDB file is present then structural features of the protein can be displayed in the structure tab. The low wavelength cut-off of the data can be determined by plotting the HT signal. Header information can then be added about various experimental data including the low wavelength cut-off. The spectra should then be calibrated to millidegrees using a calibration compound such as CSA recently collected on the machine. If the pathlength, concentration and

sequence are known the spectra can now be scaled to Δε units. At this point the spectra is ready to be analysed by one of the analysis methods in *CDtool*.

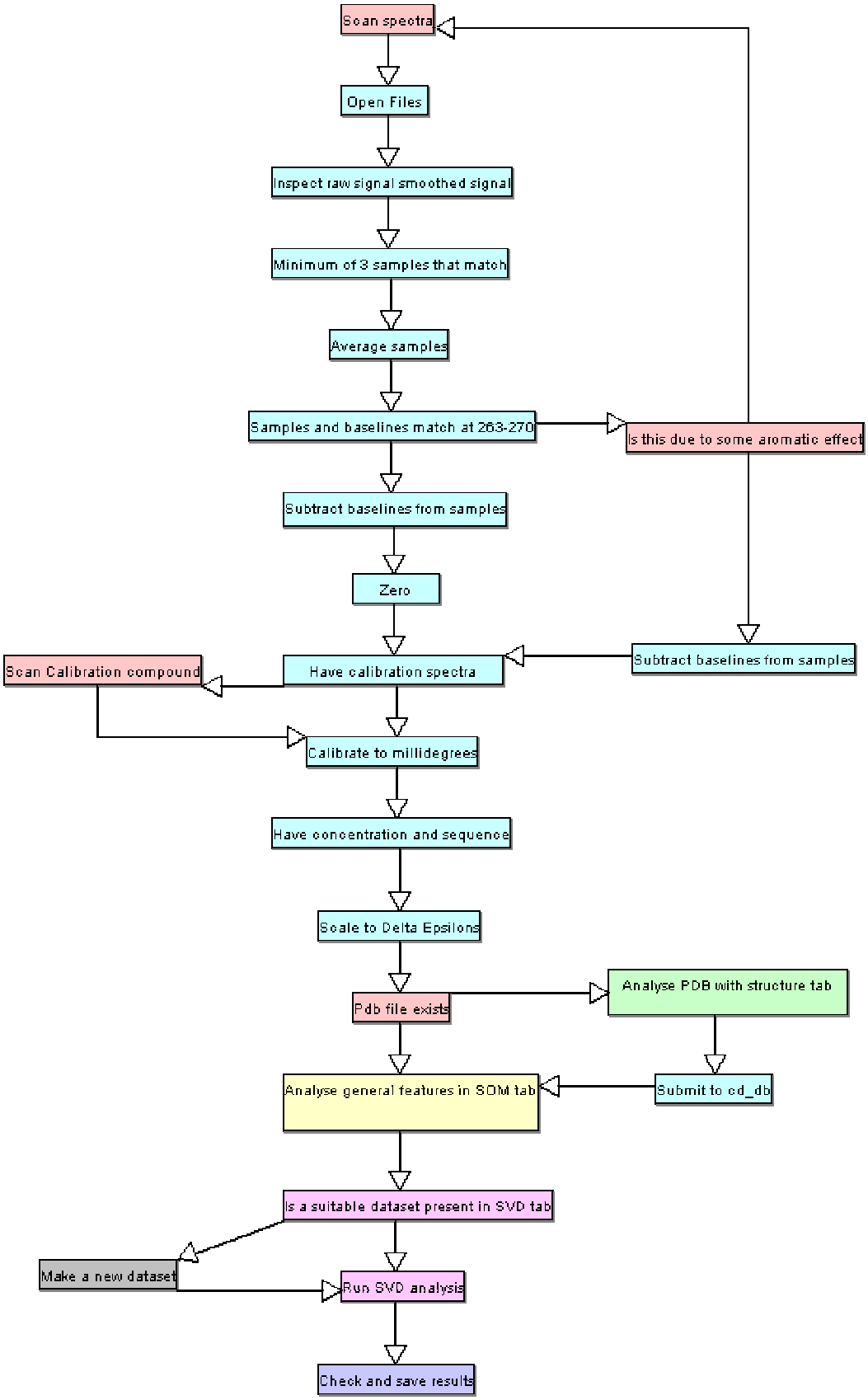

Scan spectra
Open Files
Inspect raw signal smoothed signal
Minimum of 3 samples that match
Average samples
Samples and baselines match at 263-270
Is this due to some aromatic effect
Subtract baselines from samples
Zero
Scan Calibration compound
Have calibration spectra
Subtract baselines from samples
Calibrate to millidegrees
Have concentration and sequence
Scale to Delta Epsilons
Pdb file exists
Analyse PDB with structure tab
Analyse general features in SOM tab
Submit to cd_db
Is a suitable dataset present in SVD tab
Make a new dataset
Run SVD analysis
Check and save results

Figure 5.4: A UML diagram showing the protocol for using CDtool. Different colours for the boxes indicate processes been carried out in different tabs of CDtool.

5.4 Plot tab

The plot tab is the first tab that is displayed when the program is executed and is concerned with visualising and processing CD spectra (figure). The left side contains a list view detailing the currently open files. Selecting an item in the list view makes it active for functions to act on. The right side has a plot area where the CD spectra are visualised. This list view has 5 columns. The first column indexes the number of the spectra opened. This is increased by 1 whenever a new spectrum is created. The operation column indicates the last operation that was carried out. Possible values are “ave”, “sub”, “sca”, “cal” and “zer” corresponding to average, subtracted, scaled, calibrated and zeroed respectively. The samples column details the list of sample files used to make the spectra, whilst the baselines column lists the baseline files used. The data can be ordered by a specific column by clicking the top of a column. Common functions such as averaging subtracting and zeroing are accessed via buttons at the bottom left of the tab.

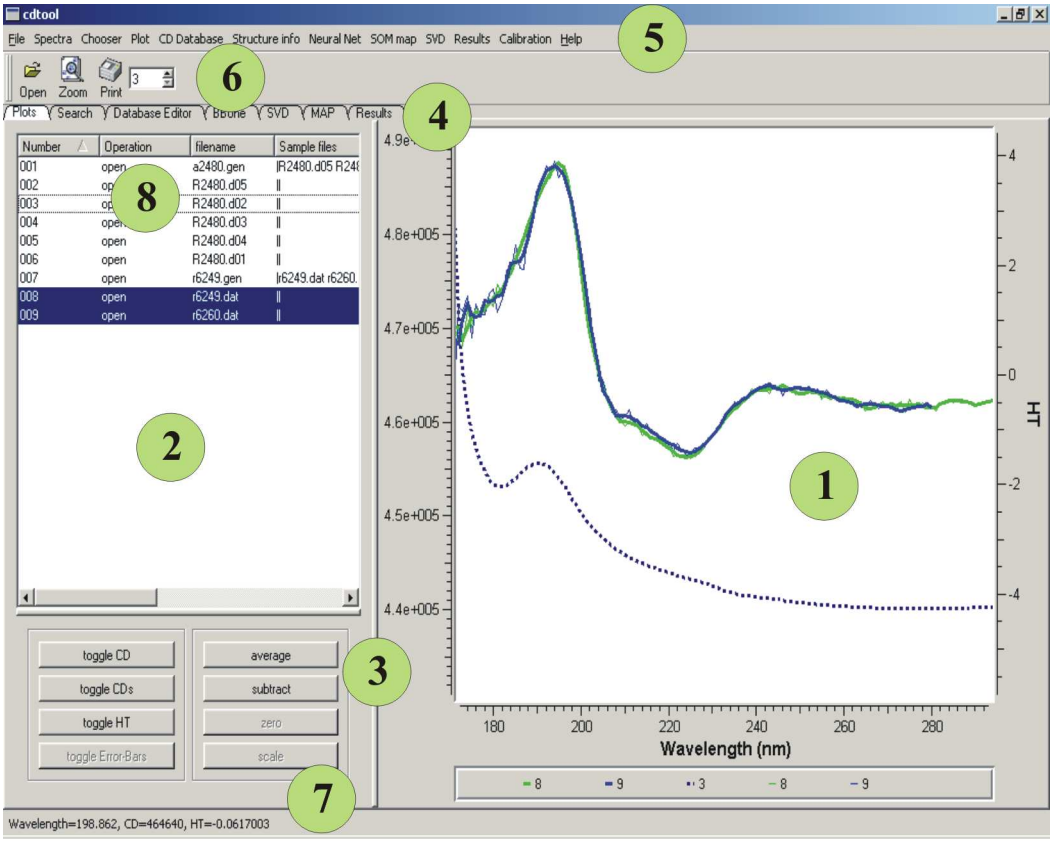


Figure 5.5: Screenshot of the plot tab. (1) The plot region displays spectra data. (2) The list view details the currently open files. (3) Commonly used functions have easily accessible buttons. (4) The tab bar headers, allow for different tabs to be displayed. (5) The top level menu. (6) The toolbar. (7) The status bar displays various information about the plot area.

5.4.1 Data input

A file. or multiple files can be opened by selecting File → Open from the top level menu bar. *CDtool* is able to read in file formats produced from several different machines. Currently supported formats are Aviv, Jasco, SRS CD-12 and ISA UV-1. The filenames and other details of currently opened files are then plotted in the list view.

5.4.2 Data visualisation

After a file has been opened its data is automatically smoothed and plotted in the plot area. Selecting one or more list view items and pressing the 'toggle CDS' button switches between displaying and hiding the CD-smoothed data. The 'toggle-HT' and 'toggle-CD' buttons do the same for the HT-signal and the CD-signal respectively. To keep track of which spectra are currently plotted, the user can right click on the list view. Selecting 'highlight plotted CDs' will highlight all of the currently displayed CD smoothed data. Selecting 'highlight plotted CD' or 'highlight plotted HT' will do the same for the CD and HT data respectively. The plot area can be cleared by choosing Plot → 'Remove all plots' from the menu bar. Clicking on a curves legend or right clicking the mouse on a curve selects the item corresponding to this curve, and de-selects all the other curves.

5.4.3 Editing the plot

The plot area can be customised in a variety of ways. Right-clicking the background of the plot produces a pop-up menu bar from which general features of the plot can be altered. Selecting the 'plot titles' option produces a dialog box (figure 5.6), which allows various details of the plot to be altered including the axis and main title text. Selecting show/hide tick box shows or hides an axis.

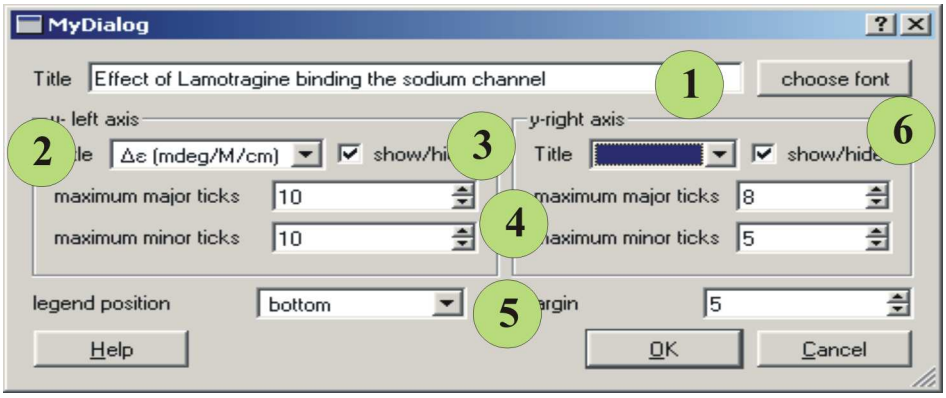


Figure 5.6: Screenshot of the titles editor dialog from CDtool. (1) The title can be edited here. (2) There are various options for the axis labels. (3) An axis can be

displayed or hidden. (4) The number of tickmarks on the axis can be altered. (5) The legend position can be specified. (6) The font used can be altered.

Selecting the 'curve styles' option from the background menu produces a new dialog (figure 5.7). Here the colour, style and width of a line can be chosen. The legend text for a given curve can also be altered.

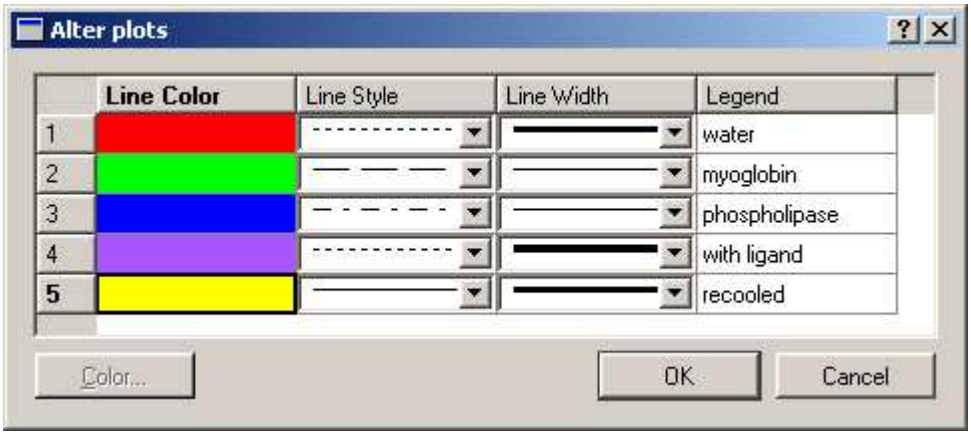


Figure 5.7: Screen shot of the curve style dialog. Colours, line-styles and legends for curves can be selected here.

5.4.4 Plot Output

The plot area information can be saved in a number of ways. A bitmap image of the plot area can be saved in one of a number of file types (PNG, BMP, JPG) by selecting Plot → 'Save image' from the menu bar.

Alternatively the plot area can be outputted to a printer or postscript file by selecting Plot → 'Print plot' from the menu bar. This gives the usual printer options to which the plot area can be printed. Selecting 'print to file' box will cause the plot area to be saved as a postscript file.

5.4.5 Analysing details of a plot

Pressing the left mouse button in the plot area produces a set of green crosshairs. The readout of the position of the cross hairs on the three axis is shown in the status bar at the bottom of the main window. Selecting the Plot → 'Zoom' option from the menu changes the mode to zoom mode. Now by clicking the mouse button on the plot area and dragging produces a rectangle. When the mouse button is released the area of the plot is enlarged. Selecting un-zoom from the plot menu bar will reset the plot to the full scale.

5.4.6 Adding experimental notes to the plot

Experimental information can be added to a plot before printing out. This is a useful quick tool for adding notes to a plot for future reference. To do this select a single item in the list view and then choose the Plot → 'Add header' option from the menu bar. Then clicking on the plot with the left mouse button will add the header information about the file to the plot area. To remove the header information select the Plot → 'Remove header' option .

5.4.7 Averaging and subtracting

Selecting multiple spectra-items and clicking the average button produces a new averaged spectrum that is displayed in the list view. The sample files used in averaging are added to the sample files column of the list view. Subtraction of two spectra is carried out by first selecting one spectrum, then a second and clicking the subtract button. This subtracts the second selected item from the first.

In the process of averaging and subtracting, if one spectrum has a larger interval than the others it is the larger interval that is used for the new spectrum. If one of the spectrum has an important difference to any of the others then a warning box is displayed, indicating the differences. For example if one spectrum was collected with a different dwell time, a warning box will appear explaining the differences. An option exists for subtracting multiple files by the same baseline. This can be done by selecting the files to be baseline subtracted in the

list view and then selecting Spectra → 'Subtract multiple files' from the menu bar. The resulting dialog box allows the user to specify the unique spectrum ID to use as the baseline.

5.4.8 Zeroing

The zeroing button produces a dialog box allowing zeroing between two wavelength boundaries. This box defaults to 263 and 270 since this region is most frequently found to be CD-silent for proteins. To zero at a single point, the values in both of the boxes can be set to the same number. By default the raw CD data is used for calculating the baseline shift.

5.4.9 Calibration to millidegrees

After zeroing, the spectra can be calibrated using a series of calibration standards. The simplest way to do this is to choose Spectra → 'Calibrate to Millidegrees Using CSA' from the menu bar. This produces a dialog box where the CSA calibration parameters can be inputted (figure), a one or two point scaling method can be specified here. Clicking 'OK' will then calibrate any of the selected spectra in the list view.

Figure 5.8: Screen shot of the CSA calibration dialog. All of the parameters for CSA calibration can be inputted here.

A more flexible calibration system is provided by selecting Calibration → 'Calculate new polynomial' from the menu bar. Here the user is able to specify a number of theoretical and measured calibration values. Scaling using several points has been shown to increase the match between CD spectra from different machines ().

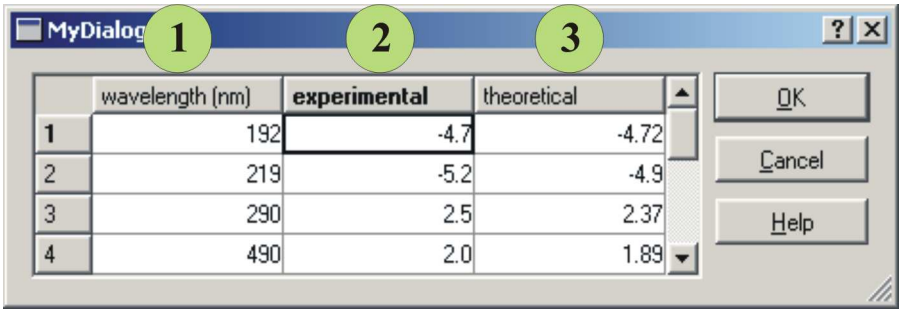


Figure 5.9: Screenshot of the calibration dialog. The first column labelled (1) contains the wavelength value. Columns (2) and (3) contain the experimental and theoretical respectively, for a known calibration standard.

The default wavelengths and theoretical values in the dialog box are those from previous work (). However the user is free to input their own wavelengths with corresponding experimental and theoretical values. After clicking 'OK' the new calibration curve is created and can be used for scaling by choosing Calibration → 'Scale by polynomial' from the menu bar. The curve can be visualised or hidden by selecting the display or hide calibration curve options from the calibration menu bar.

Compound	**Wavelength (nm)**	**Intensity delta epsilons**
CSA	192	-4.72
CSA	290	+2.37
Pantolactone	219	-4.9
Cobalt (III) tris-ethylenediamine	490	+1.89

Table 5.4: The table shows the theoretical values in Δε units for different calibration compounds used as default options in CDtool. Values were obtained from Miles et al.()

The 2nd order polynomial is generated on these points using the following matrix equations. First of all, the information about the system is generated by equation (5.1). The y values are defined as the ratio of the theoretical over the

experimental values whilst the x values are the corresponding wavelengths. The coefficients of the polynomial are then generated by (5.3). Where the rows of ***C*** contain the cooeficients of the polynomial in its rows.

$$\mathbf{A} = \begin{vmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ x_3^2 & x_3 & 1 \end{vmatrix} \times \begin{vmatrix} x_1^2 & x_2^2 & x_3^2 \\ x_1 & x_2 & x_3 \\ 1 & 1 & 1 \end{vmatrix} \tag{5.1}$$

$$\mathbf{D} = \begin{vmatrix} x_1^2 & x_2^2 & x_3^2 \\ x_1 & x_2 & x_3 \\ 1 & 1 & 1 \end{vmatrix} \times \begin{vmatrix} y_1 \\ y_2 \\ y_3 \end{vmatrix} \tag{5.2}$$

$$\mathbf{C} = \mathbf{A}^{+} \times \mathbf{D} \tag{5.3}$$

5.4.10 Conversion to Δε units

Selecting Spectra → 'Scale to DE' from the menu bar produces a new dialog box (figure). After specifying the appropriate values and clicking 'OK' the spectra will be scaled to Δε units according to equation ().The concentration here is in mg /ml.

Figure 5.10: Screenshot of the Δε dialog. The user can input the Concentration in mg/ml, pathlength in cm and MRW.

5.4.11 Recording and viewing experimental information

Selecting the Spectra → 'Edit headers' from the menu bar produces a wizard for inputting experimental information about a spectrum (figure 5.11). Several important pieces of information can be added here. Importantly the cut off value should be specified here to indicate below which wavelength the data is no longer valid.

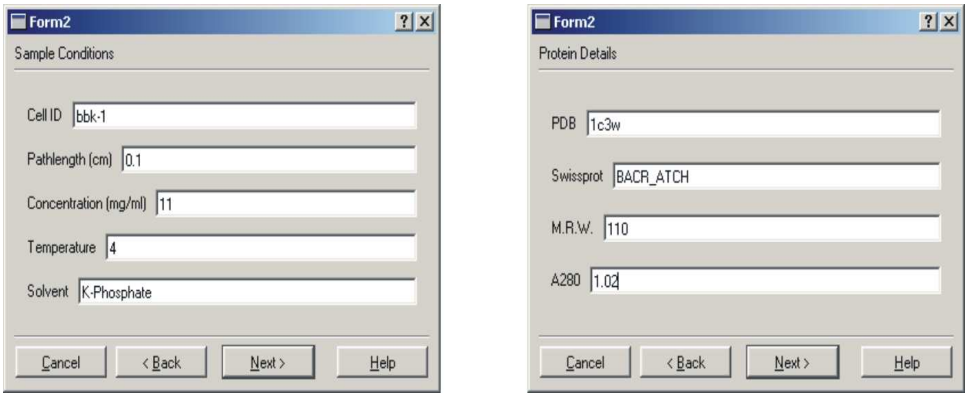


Figure 5.11: Screenshots of the different pages in the header info wizard. Various experimental details can be altered using the wizard.

Right-clicking on a spectrum in the list view and selecting the 'Show header' option displays the spectra data in a dialog box (figure 5.12). Typing text into the search box and pressing return searches the document and moves to matching text.

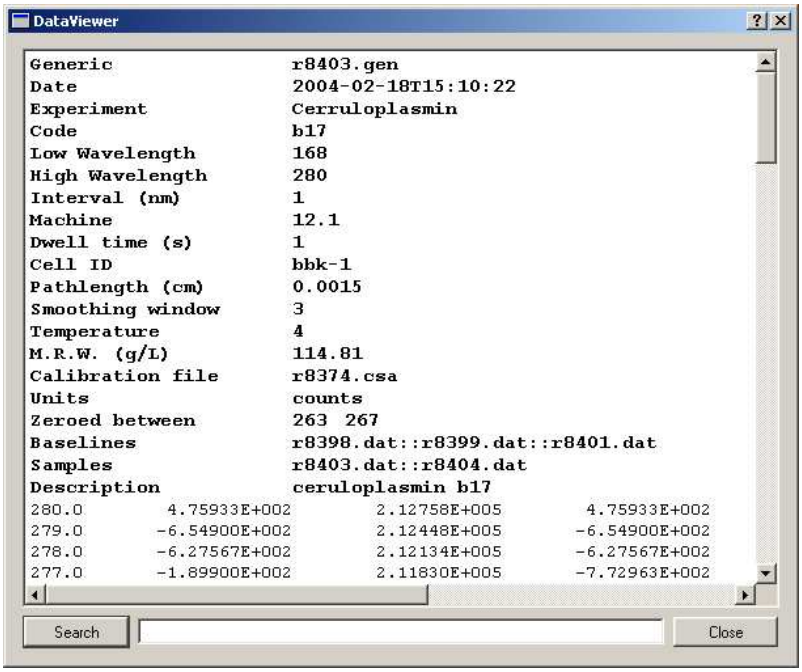


Figure 5.12: Screenshot of the header dialog. (1) The header part of the file detailing experimental information. (2) The data part of the file containing the spectra part of the data. Columns 1,2,3,4,5,6 and 7 contain the wavelength, CD signal, HT signal and CD smoothed signal, pseudo-absorbance, standard deviation of the sample scans and standard deviation of the baseline scans respectively. (3) The search box allows searching for matching text in the dialog.

Entry	Description
Generic	The filename of the spectra
Date	The date and time the spectra was collected
Experiment	The name of the experiment
Code	The experimental code uniquely identifying the protein sample
Low Wavelength (nm)	The low wavelength the data was collected to
High Wavelength (nm)	The high wavelength data was collected to
Cut off (nm)	The low wavelength cut-off of the data
Interval (nm)	The interval of the data
Machine	The type of machine the data was collected on
Dwell time (s)	The dwell time (averaging time) of data collection in seconds
Cell ID	The unique code given to the cell the sample was collected in
Pathlength (cm)	The pathlength of the cell used in cm
Smoothing window	The smoothing window used in generating the CD smoothed column (column 4)
Temperature	The temperature the sample was collected at in ºC
M.R.W.	The mean residue weight of the sample
Calibration file	The calibration file used for scaling to millidegrees
Units	The units of the spectra data
Zeroed between	The region the data was zeroed between
Baselines	The baseline files used to make the spectra
Samples	The sample files used to make the spectra
Description	A more detailed description of the experimental details

Table 5.5: Table detailing the different entries provided in the header of a spectrum file.

5.4.12Data smoothing

Data smoothing is carried out automatically for any newly created spectra. The default smoothing windows applied to the data have been chosen from experience to be appropriate for a given interval. However, it is possible to change the smoothing window, by choosing the spectra→'change smoothing window' option.

5.4.13 Displaying error bars

Error bars for a processed spectrum can be visualised by right-clicking the mouse on a curve in the plot and selecting the 'show-error bars' option. The interval and individual points at which error bars will be plotted can be selected. Options are given for including the standard deviation of the baselines as well as the samples, and how many standard deviations should be plotted.

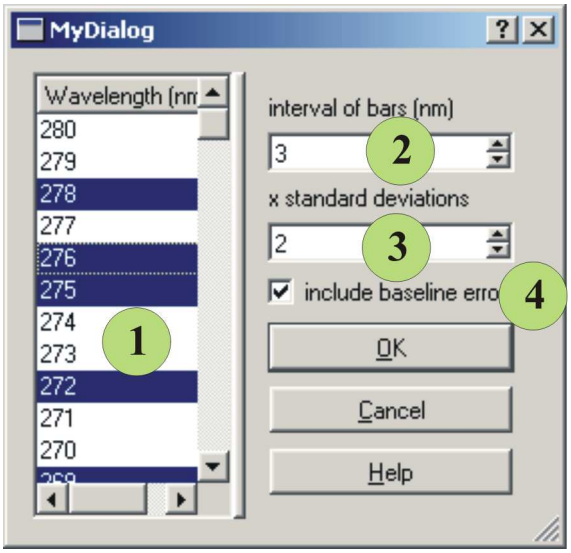


Figure 5.13: Screenshot of the error-bar dialog from CDtool. (1) The list view allows selection of positions of the error bars. (2) The spin box provides a means for selecting the interval at which error bars should be plotted. (3) The spin box allows the user to specify how many standard deviations should be plotted. (4) An option of including the baseline error is provided.

The pooled standard deviation of the sample and baselines are calculated from equation (5.4).

$$stdev_{TOTAL} = \sqrt{(stdev_{baselines})^2 + (stdev_{samples})^2} \quad (5.4)$$

5.4.14 Saving results

After processing, the spectra can be stored by saving as a text file for later use by selecting one item from the list view and choosing save from the file menu bar. If one of the filenames already exists a warning box will be shown to confirm if the old file should be overwritten. Alternatively several spectra can be saved in one go by selecting save multiple spectra and choosing the directory for the spectra to be saved to.

5.5 Database tab

Information in the database can be accessed via the database tab (). This includes several features such as retrieving spectra, and making new datasets for SVD.

5.5.1 Connecting to the database

Before any of the database functions can be accessed the user must first connect to the database. In order to do this, a username and password are required. These can be obtained by email (*cdtools*@mail.cryst.bbk.ac.uk). However until the Birkbeck CD database has been published this feature will not be available. An alternative method for installing your own local database is described in section 1.3.

To connect, select the database → 'connect to database' option from the database menu, and a dialog box will appear (figure 5.14). Input the correct driver, hostname and port number. These will automatically be set for connecting to the Birkbeck database but will need to be set if a local database is installed. Input the

user name and password and click the connect button. The left hand list view of the database tab is then automatically populated with the relevant tables.

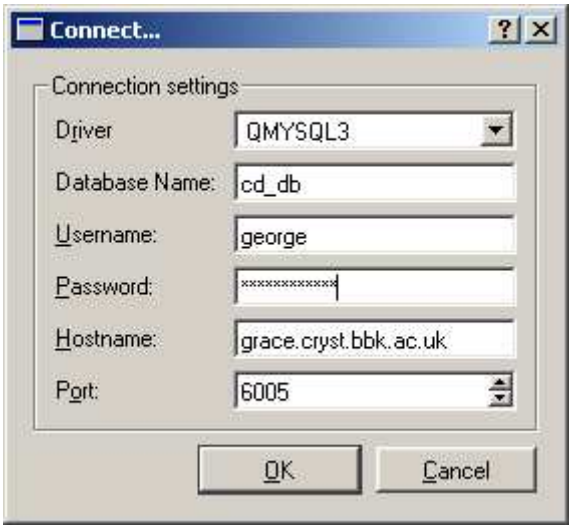


Figure 5.14: Screenshot of the database connection dialog from CDtool.

5.5.2 Navigating the database

Double clicking one of the list view items displays the tables contents in the main data table. For users with sufficient privileges, elements of the table can be deleted, inserted or edited by right-clicking a row of the table. The text edit allows users to supply MySQL queries the results of which are shown in the main data table. Selecting one of the data-tables column headers sorts the data alpha-numerically by the values in that column.

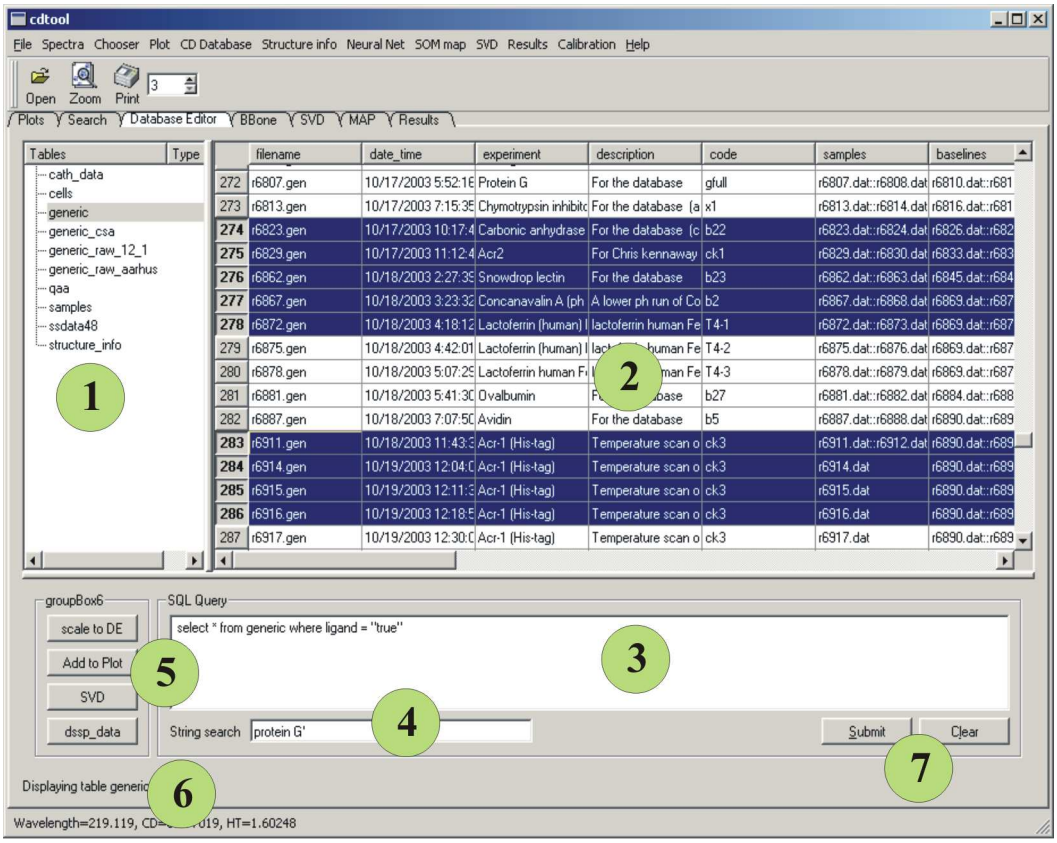


Figure 5.15: Screenshot of the database tab. The relevant tables are displayed in a list view (1) Results of database query are displayed in the central widget. (2) Tables from SQL queries are displayed in the main data table. (3) MySQL queries can be submitted directly. (4) Text in the data table can be searched. (5) Common database operations have buttons. (6) Queries can be submitted via a submit button.

5.5.3 Retrieving Spectra

To retrieve spectra from the database, the user can select one or multiple rows from the 'generic' table and press the 'Add Spectra' button. This will retrieve the spectra from the database and display it in the main plot window. The data is retrieved to the plot tab, as it is stored in the database with no extra scaling. Alternatively the 'Add and Scale' button can be pressed to produce a dialog box (figure 5.16). Selecting the 'mdeg' box will add the data to the plot in calibrated by the calibration compounds to millidegrees. Selecting a single calibration wavelength will scale the spectrum to millidegrees by a single point scaling.

Selecting two calibration points will add by vector scaling. Selecting three or more calibration points will add the data to the plot by a 2nd order polynomial scaling, as detailed above. If the 'DE' box is checked then the spectra will be added in Δε units. Again selecting different numbers of calibration boxes produces different scaling methods. If for some reason one or more of the values for scaling is not present then a dialog box will be displayed alerting the user to why the spectra couldn't be scaled.

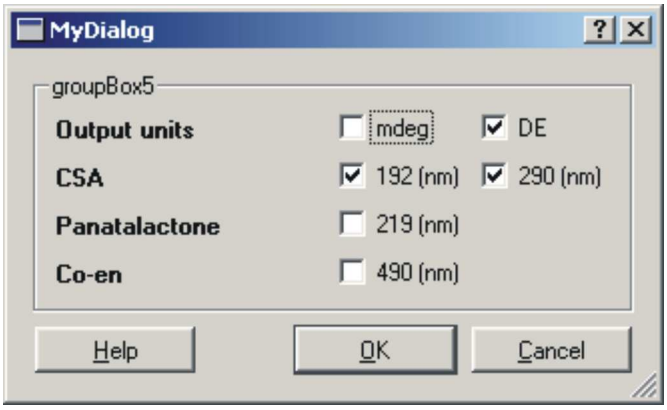


Figure 5.16: Screenshot of the database tab scaling dialog box. Check boxes are provided to indicate which calibration compounds are to be used in the scaling.

5.5.4 Making a use-defined dataset

It is possible to make a dataset for incorporation in other prediction tools by a two step process. The first step involves making a table of secondary structure values. Clicking the 'new structure table' button produces a wizard (). Selecting a secondary structure type from the combo box in the first column (secondary structure column) will add that structure to the dataset. The second column assigns the secondary structure from the first column to a certain group. Clicking the 'Next' button produces the page where each group is assigned a name. Clicking the 'Next' button again brings up the final page. This page allows for determining which way overlaps should be dealt with. Clicking the hierarchy check box causes overlaps to be dealt with hierarchically. This would mean for example, if a residue was assigned to β-sheet and PP-II helix, and β-sheet had a higher group number, the residue would be assigned as 100% sheet and 0% PP-II

helix. However, if the hierarchical check box is not clicked then such a residue would be assigned as 50% sheet and 50% PP-II helix (ie. the overlap is normalized so the total assignment for that residue is 100%).

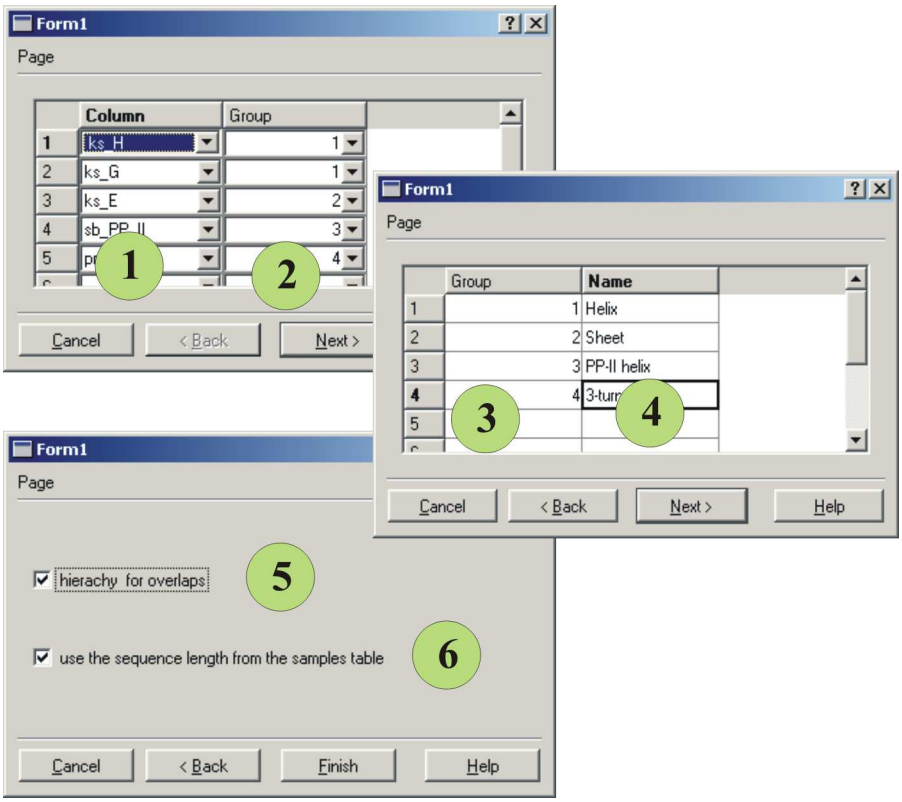


Figure 5.17: Screen shot of the wizard for making a new dataset. The first page of the wizard is concerned with determining which secondary structures to be inc1uded in the dataset and how they are grouped together. (1) Allows for different secondary structures in the database to be selected. (2) Describes which group the secondary structure is assigned to. (3) Different groups can be named here. (4) The mode of overlap can be chosen. (6) The mode by which the total number of residues is determined can be chosen.

The total number of residues is then used to convert the number of residues found for each group to a percentage. This number by default is the total number of C-alpha carbons with defined electron density. However if the 'include undefined residues' option is clicked then residues without electron density are included in the total number. Finally an extra class is automatically defined called 'other' that includes the remaining percentage of residues not classified into the other groups.

After this wizard has been executed a temporary table is created called ‘secondary_structure’, with the columns containing values for fractions of secondary structures between 0 and 1. This table is saved for as long as the user is logged into the database, or a new data-table is created.

The next stage in making a dataset involves selecting the rows from the ‘secondary_structure’ table to include. Pressing the ‘make dataset’ button produces a new dialog box. The wavelength range desired can be specified here. Executing the dialog results in a dataset that is immediately imported to the SVD tab.

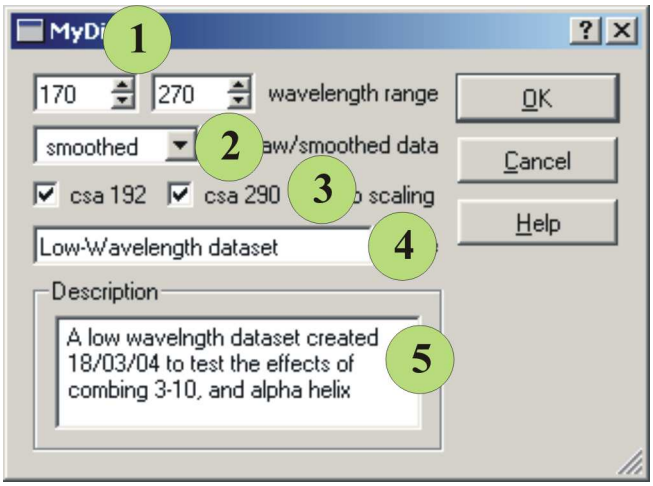


Figure 3.5: Screenshot of the second dialog for making a new dataset. (1) The wavelength range to be included can be specified. (2) The smoothed or the raw data can be used. (3) The method of scaling can be selected. (4) The name of the dataset can be added here. (5) A description of the dataset for future reference can be given.

5.6 The SVD tab

This tab contains tools for various SVD methods used with CD. The currently opened datasets are displayed in the list view.

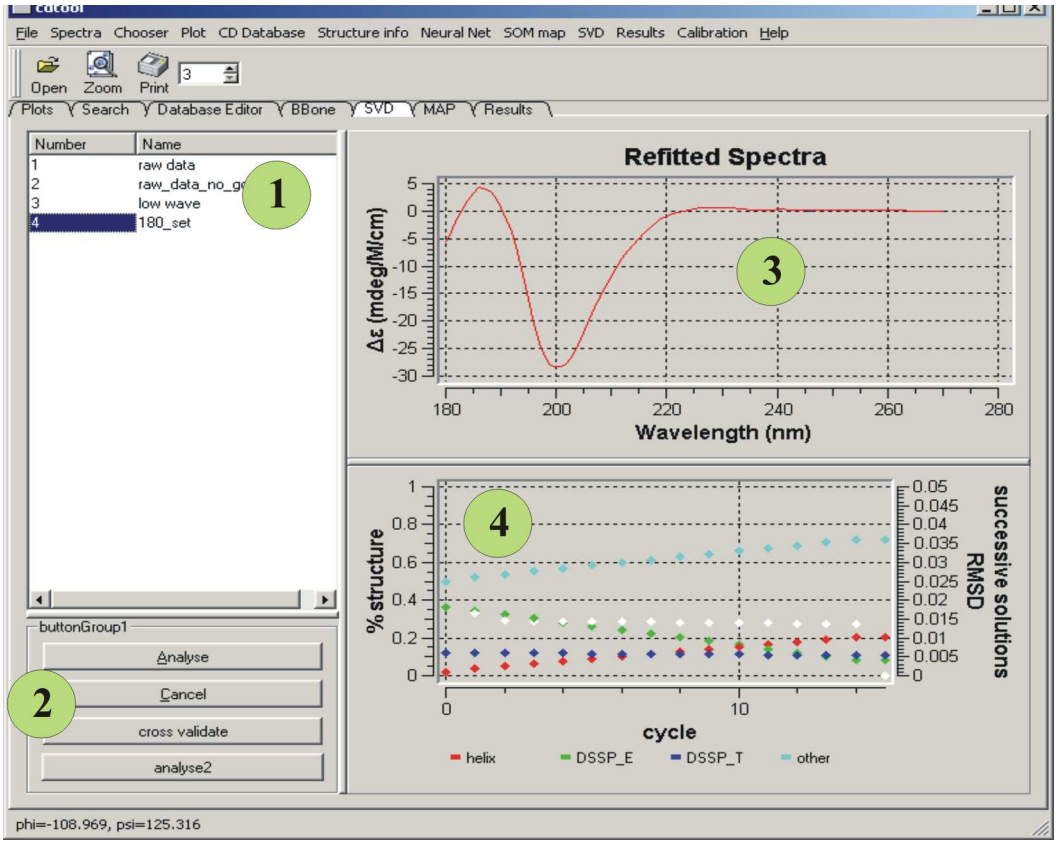


Figure 5.19: Screenshot of the SVD tab. (1) The list view details the currently open datasets. (2) Commonly used function buttons connected with SVD. (3) Plot showing the query spectra and the best refitted spectra of an SVD solution. (4) A plot showing the successive solutions of the SVD algorithm.

5.6.1 Reference dataset format

Right-clicking on adataset item displays a menu which gives various options connected with the dataset. Choosing the 'Show data' option at this point allows the user to display the information in that dataset (figure 5.20).

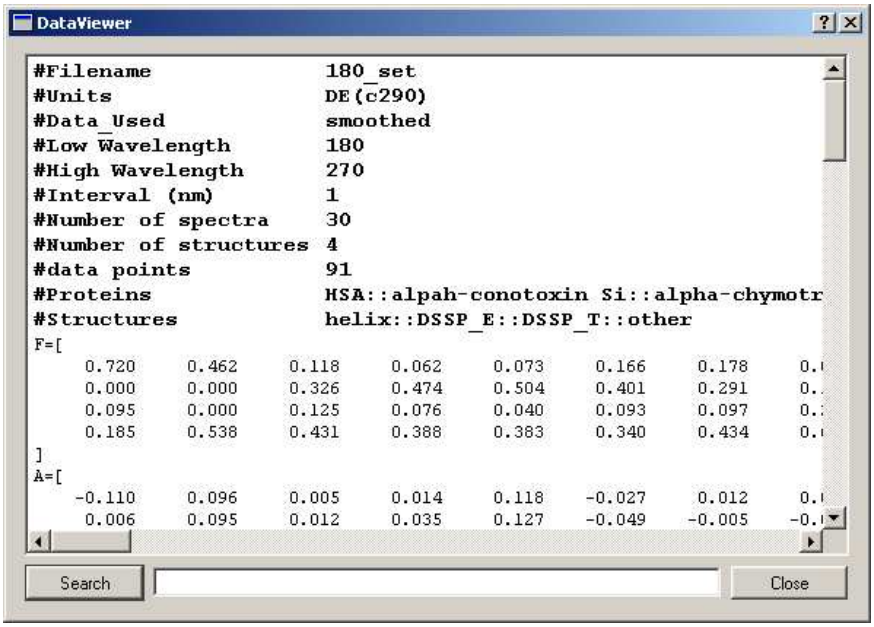


Figure 5.6: Screenshot of SVD data dialog containing information about a dataset. (1) The header information. (2) The F matrix containing secondary structure values for different spectra. (3) The A matrix contains CD spectra data.

The format consists of a data-matrix and a structure-matrix which are denoted as **A** and **F** respectively with various header details also available (Table). Different rows of the **A** matrix correspond to different wavelengths, whilst different columns correspond to different proteins. The **F** matrix has different secondary structure values in its rows whilst the columns correspond to different proteins. The header part of the file contains information about how the dataset was made. The proteins row lists all of the proteins used in creating the dataset. The structures row denotes which secondary structures that are classified in the dataset. The wavelength range denotes the wavelengths included. The contents can be saved to a text file by right clicking a dataset and selecting the 'save dataset' option. The file can then be stored for later use. The file format thus saved can be directly incorporated into the *octave* package ().

Entry	**Description**
Filename	The name previously given to the dataset

Units	The units of the data
Data_used	The data used which can be raw or smoothed data.
Low Wavelength (nm)	The low wavelength of the data
High Wavelength (nm)	The high wavelength of the data
Interval (nm)	The interval of the data
Number of spectra	The number of spectra included in the dataset
Number of structures	The number of structures classified
Data_points	The number of data points in the data
Proteins	The list of the names of proteins in the dataset
Structures	The structures classified by the dataset
Description	A description of the idea behind the dataset

Table 5.6: Table detailing the header information in the SVD dataset.

5.6.2 Calculating secondary structure curves and basis spectra

Options are provided for calculating various component curves of a dataset. Right-clicking a dataset and selecting ‘calculate component curves’ produces a new dialog box for this purpose (figure 5.21). The number of eigenvectors and spectra to be used can be specified in this dialog. Selecting the ‘show basis spectra’ check-box will calculate and plot the basis spectra according to equation (). Selecting the ‘show secondary structure curves’ check-box will calculate and plot the secondary structure curves.

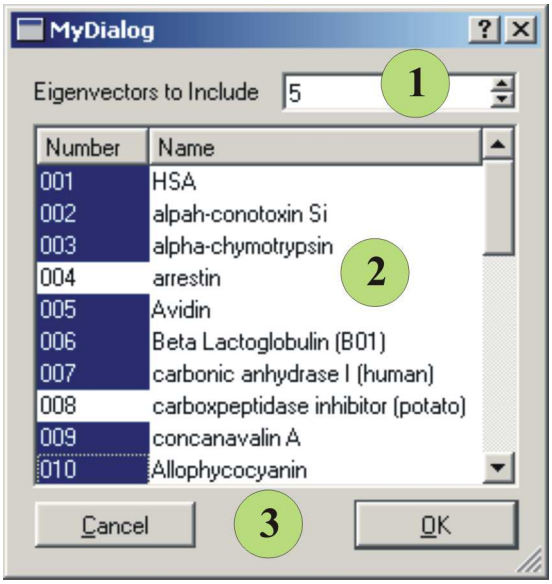


Figure 5.21: Screenshot of component curves dialog. (1) The number of eigenvectors to include. (2) The List of proteins to include in the analysis can be selected. (3) The type of component curves to be plotted can be specified.

The method of calculating the pure secondary structure curves the same method as decribed by Johnson et al. (). The **A** matrix of the dataset corresponds to **A** in equation (5.5). The number of spectra selected and eigenvectors used gives the value *x* and *e* in equation (5.6) respectively. The effect of lowering the value of *e* corresponds to reducing the amount of noise. For most reasonably sized datasets the *e* value can be set to about 5. The same SVD decomposition is then applied to the **F** matrix (5.7). The component curves can be calculated by (5.8). This gives the matrix **X** that contains the spectra for the 'pure' secondary structures in its rows. The component curves are added to the main plot tab for inspection.

$$\mathbf{UWV}' = svd(\mathbf{A}) \tag{5.5}$$

$$\mathbf{A}_{(n,e)} = \mathbf{U}_{(n,e)}\mathbf{W}_{(e,e)}\mathbf{V}'_{(e,x)} \tag{5.6}$$

$$\mathbf{UWV}' = svd(\mathbf{F}) \tag{5.7}$$

$$\mathbf{X} = \mathbf{A}_{(n,x)} \tag{5.8}$$

5.6.3 SVD analysis functions

The main function of the SVD tab is to analyse spectra for secondary structure content by SVD methods. Selecting a dataset to use and clicking the 'Analyse' button carries out SVD analysis on any spectra selected in the plot tab list view. It is essential to ensure that the low wavelength cut off of the data has been added to the header file. The only method currently implemented is a self consistent method. The method is similar to the Selcon2 method. The main difference is that more than 1 solution is found for each number of basis spectra / size of basis set combination. Although this method gives reasonable accuracy the method has not been published, and it is advisable to use the various methods in the CDPro package for comparison.

5.7 Results Tab

The results tab displays results from the various prediction techniques employed (). The results of the main table can be saved to a tab-delimited text file. The lower text edit contains a log of the analysis methods used to produce the table.

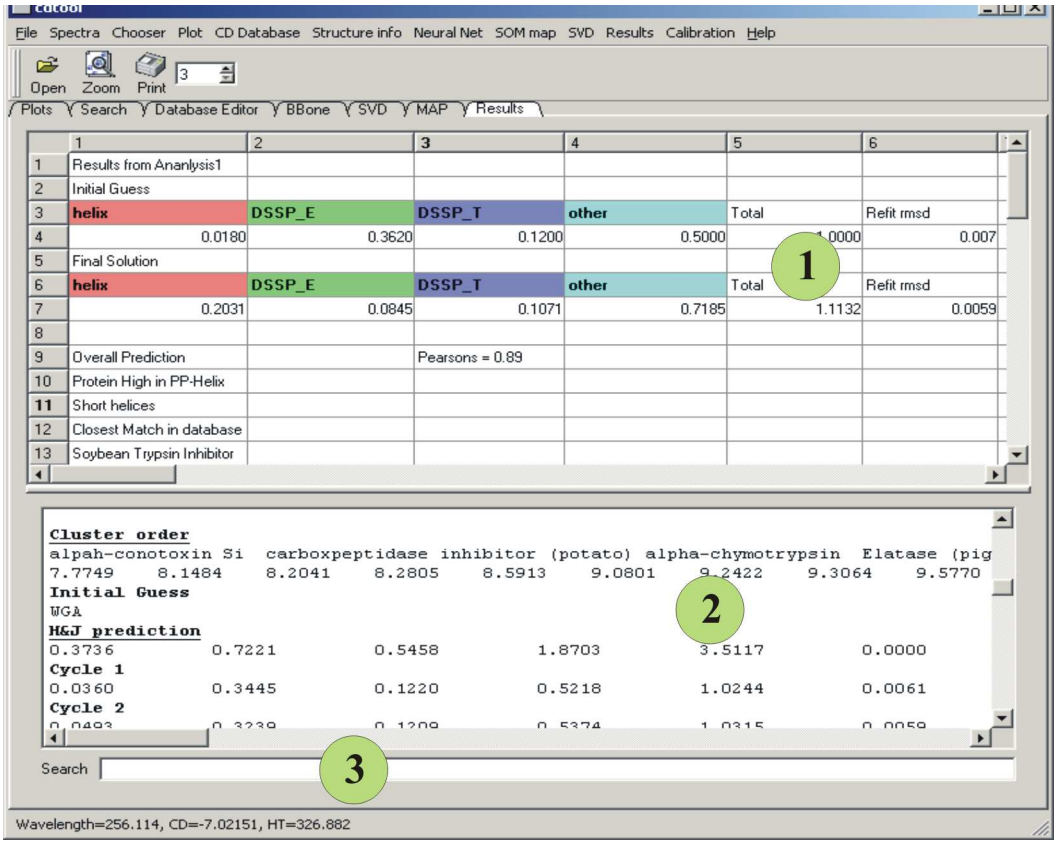


Figure 5.22: Screenshot of the results tab. (1) The table details results from secondary structure analysis. (2) The lower text area details a log of the secondary structure analyses carried out.

5.8 3D-Plot tab

The 3D-plot tab allows for displaying 3-D plots generated by temperature and time series scans. Theplot can be zoomed or rotated by the usual mouse actions.

Figure 5.23: Screenshot of the 3D-plot tab.

5.8.1 Plotting Data

Data can be inputted into the 3D plotting area in a number of ways. From the plot tab selecting a number of spectra and choosing 3D plot → 'Add to 3D-plot' from the menu bar. Alternatively a previously created 3D plot that has been saved

can be re-opened directly by choosing 3D plot → 'Open 3D-plot' from the menu bar. The data format the plot is saved in is the same as the dataset format from the SVD tab.

5.8.2 Lighting options

The lighting options of the plot can be altered by pressing the 'lighting' button (figure 5.24). Several of the parameters of the surface and the light source can then be altered to give the preferred appearance of the plot.

Figure 5.24: Screenshot of the dialog for choosing the lighting preferences for the 3D-plot.

5.8.3 Outputting the plot

The 3D-plot can be outputted to various formats including PNG, BMP, and PDF formats. This is done by selecting the format from the 3D-plot tab tool bar and clicking the 'save' tool bar icon.

5.9 Clustering tab

The clustering tab provides a means by which spectra can be clustered by hierarchical clustering methods. The resulting dendrograms can be visualized and then cut into various numbers of clusters (figure 5.25).

Figure 5.25: Screenshot of the clustering tab.

5.9.1 Importing data

Two methods exist for importing data for clustering. Selecting a number of spectra from the plot tab and choosing Clustering → 'Import spectra'. The other method is to import a dataset by choosing Clustering → 'Open dataset' from the

menu bar. The file format of the datasets for this is the same as the dataset format for the SVD dataset tab.

5.9.1 Distance measures

Several different methods exist for generating the distance matrix for clustering the data and can be chosen via a pulldown menu. Additionally to the usual distance measures certain combinations of distance measures are provided. For example a mixture of the pearsons and spearman correlation coefficients is provided as a distance measure, since this has previously been shown to be useful in data analysis. Options are also provided for chososing the distance measure between the clusters.

5.9.2 Classifying the clusters

An option exists to subdivide the dendrogram into a given number of clusters. The number of clusters to be formed can be specified by changing the number next to the cut tree button. The resulting clusters in the dendrogram will be colour coded.

5.9.3 Editing the dendrogram

Various aspects of the dendrogram can be changed by the user.

5.9.3 Outputting the dendrogram

The dendrogram can be printed to a postscript file or to a prionter by selecting the Clustering → 'Print tree' option from the menu bar.

5.10 Structure tab

The structure tab provides a means of viewing structural features of a PDB file most relevant to CD. Currently PDB files can only be loaded from the database. However future versions will have the facility to load PDB files directly and assign secondary structure. If the program is connected to the database, the data table is populated with a list of available PDB files. Other columns of the table contain secondary structure summaries for the file. A structure can be loaded into the viewer by selecting a PDB file from the data table and clicking the display button. The structure is displayed in the Bbone viewer and the Ramachandran plot populated with residues/points being coloured initially by the DSSP assignment (). The structure can be zoomed, rotated or translated by the usual mouse buttons.

Left clicking the mouse on the Ramachandran plot displays the phi and psi angle at that point in the status bar of the main window. Right clicking the mouse on a marker in the Ramachandran plot labels the marker with its amino acid type, residue number and chain identifier. Right clicking away from a marker produces a menu bar. Selecting the ‘set zoom on’ option from changes to zoom mode. Now left clicking and dragging over an area in the plot zooms into that area.

Residues and their secondary structure classifications can be seen in the list view. Each column of the list view can be sorted alpha-numerically. Selecting residues from the list view and clicking the highlight button highlights only those selected residues. Alternatively selecting the colour residues produces a colour dialog from which the colour of the selected residues can be altered. In this way any of the 20 or so secondary structure assignments can be grouped together and highlighted / coloured in the Ramachandran plot and structure viewer.

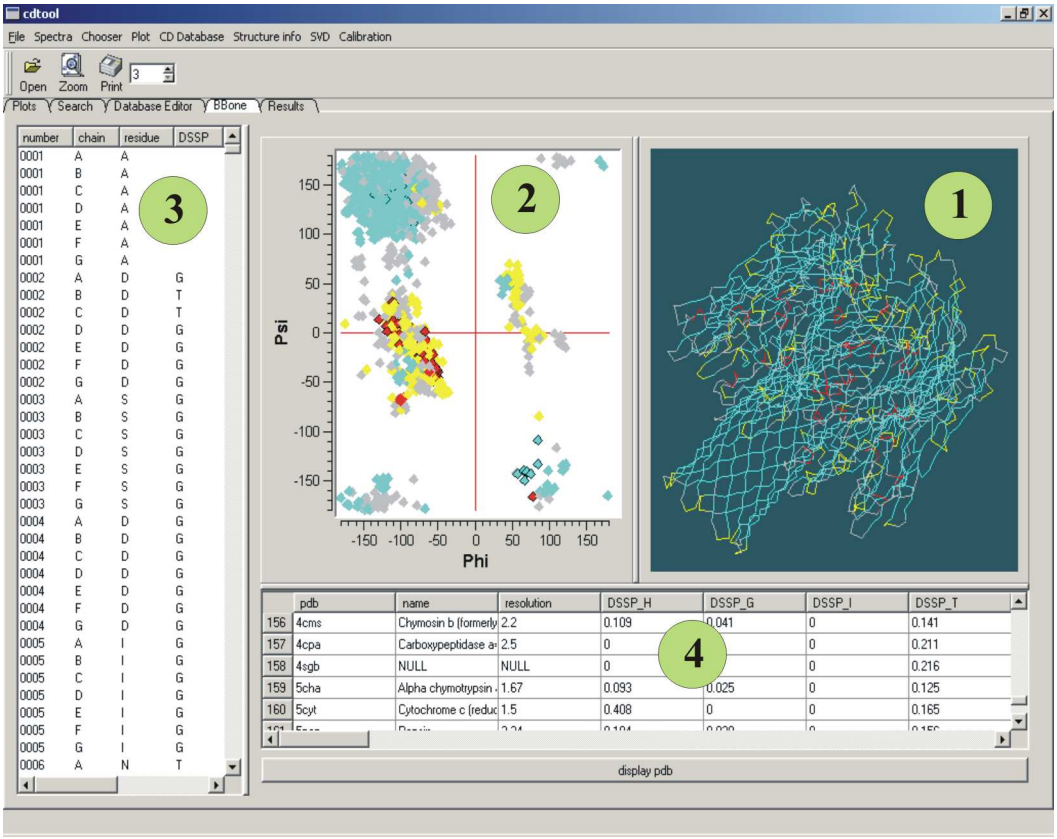


Table 5.26: Screenshot of the Structure tab. (1) The Bbone viewer displays a C-alpha trace coloured by a secondary structure assignment. (2) Ramachandran plot displays the phi and psi angles for a PDB file. (3) A list view enables sorting and selection by various categories. (4). A table details a structural summary for a given PDB file.

5.11 Conclusions

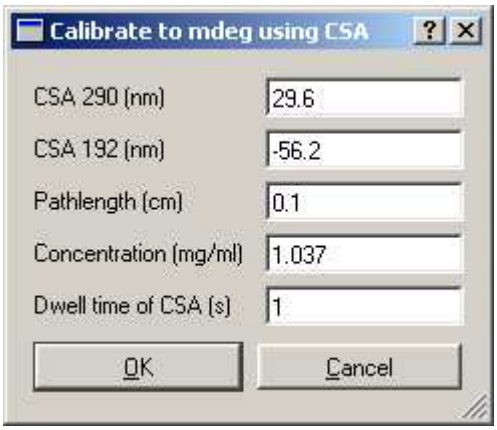
5.11 Tutorials

5.12.1 Processing a spectra

The following tutorial provides the quickest way to become familiar with *CDtool.*

1. Open the files in the tutorial folder provided with the *CDtool* download.
2. Select all of the files starting R2480, and click the average button.
3. Do the same for the R2479 files.

4. Un-select all of the files, then first click the a2480.gen file and then click the a2479.gen file. After this press the subtract button.
5. Now select the a2480.gen file with "sub" in the operations column. This is the baseline subtracted file. Press the zero button and type 263 and 270
6. Now we want to calibrate our spectra to millidegrees. Select the zeroed spectra and choose Spectra→'Calibrate with CSA to millidegrees'. Fill in the dialog box with the 290 and 192 nm peak values for the CSA spectra you have taken. Also add the CSA pathlength and concentration.



7. The spectra will now be calibrated, and so can be scaled into Delta Epsilon units. Select the calibrated spectra and choose Spectra→'scale from millidegrees to Delta Epsilon'.

[x] http://www.trolltech.com/products/qt/
[x] http://www.mysql.org/
http://www.gnu.org/software/gama
http://qwt.sourceforge.net/
http://www.stack.nl/~dimitri/doxygen/
http://www.jrsoftware.org/isinfo.php
http://lamar.colostate.edu/~sreeram/CDPro/main.html
http://www.cryst.bbk.ac.uk/cdweb/html/home.html
http://qwtplot3d.sourceforge.net/
http://www.cryst.bbk.ac.uk/classlib/bioinf/BTL.html
http://dbi.perl.org/
http://www.biochem.ucl.ac.uk/bsm/cath/
http://www.wavemetrics.com/
http://www.jascoinc.com/spectroscopy/software.shtml
http://www.photophysics.com/index.htm
http://argouml.tigris.org/
http://www.octave.org/