ALSCRIPT
A Tool to Format Multiple Sequence Alignments

Geoffrey J. Barton
University of Oxford
Laboratory of Molecular Biophysics
The Rex Richards Building
South Parks Road
Oxford OX1 3QU

Tel: (0865) 275368
Fax: (0865) 510454
Electronic Mail: GEOFF@UK.AC.OX.BIOP

Contents

1 Introduction

Alignments of protein and nucleic acid sequences are a central aid to understanding biological systems. Although several effective tools now exist for the rapid automatic alignment of large numbers of sequences (e.g. see: (Barton & Sternberg, 1987; Feng & Doolittle, 1987; Vingron & Argos, 1989; Higgins & Sharp, 1989), the preparation of publication quality figures of such alignments with annotations is often extremely difficult.

Artwork using pen/ink/dry lettering can allow labelling and boxing, however, this involves a great deal of labour and a fair degree of skill to produce visually pleasing results (e.g. the large 128 sequence globin alignment in (Barton & Sternberg, 1987)). With the advent of powerful word-processing and graphics packages for personal computers, many laboratories 'tidy up' their alignments by judicious mouse-work. Although good results may be obtained by this route, the process in common with conventional artwork is rather inflexible. A particular problem is that having chosen a number of characters per line and character pointsize, it is difficult subsequently to modify the figure. It is also laborious to add new sequences to an existing figure. In addition, one is normally limited to the use of one of a few fixed-width fonts (e.g. Courier) rather than the full range of fonts available to the word-processor. The table drawing facilities of various packages can overcome the problems of proportional fonts. For example, the Unix 'troff' macros 'tbl' were used to prepare the alignments in (Barton & Sternberg, 1990), though subsequently re-typed by the *J. Mol. Biol.*! LATEX (Lamport, 1986) tables were used for the 88 sequence annexin alignment in (Barton *et al.*, 1991) and the 67 sequence SH2 domain alignment in (Russell *et al.*, 1992); both figures were extremely time consuming to prepare, but rather easier to update than a conventionally word-processed alignment.

The ALSCRIPT program described in this article was developed specifically to allow the easy formatting and graphical display of large multiple sequence alignments. Although written originally for the author's use the interface is relatively friendly, and should be easy to learn by anyone familiar with plotting graphs.

2 Description of ALSCRIPT

ALSCRIPT takes a multiple sequence alignment in the simple AMPS (Barton, 1990; Barton & Sternberg, 1987) block-file format and a set of formatting commands, and produces a PostScript file that may be printed on a PostScript laser printer or viewed using a PostScript previewer (e.g. Sun Microsystem's PageView program). GCG "MSF" format files and CLUSTAL format files (PIR) are also supported. ALSCRIPT is strictly a formatting, display and annotation tool. It is not a program for multiple sequence alignment, or editing. The previous programs that offer the closest functionality to ALSCRIPT are PRETTYPLOT and PRETTYBOX as supplied with the GCG package (Devereux *et al.*, 1984), and the latest colour version of SOMAP (Parry-Smith & Attwood, 1991). Whilst these programs do not provide the same degree of flexibility in display as offered by ALSCRIPT, they do allow the calculation of consensus sequences and automatic shading/boxing according to defined rules. The aim of ALSCRIPT is to allow the user total control over the display of the sequence alignment; such control is essential if non-sequence information is to be used to highlight features of the sequence. For example, the location of active-site residues, the positions of secondary structures (α– helices and β– strands), and domain or intron/exon boundaries. The flexibility of ALSCRIPT also permits it to be used as a "front-end" display tool for programs that offer sophisticated alignment analysis features. For example, the AMAS (Analysis of Multiply Aligned Sequences; Livingstone & Barton, in preparation) system which highlights structurally important regions of a multiple alignment, generates ALSCRIPT commands as one output option. Similarly, the consensus algorithms encoded in the GCG program PRETTY could be used to generate ALSCRIPT shading, boxing or font-changing commands for graphical output.

Given a block-file and text pointsize, ALSCRIPT calculates how many residues can be fitted across the page, and how many sequences will fit down the page, it then prints the alignment at the chosen pointsize on as many pages as are needed. Running ALSCRIPT with a smaller or larger pointsize will automatically re-scale the alignment to fit on fewer or more pages as appropriate. The actual page dimensions may be re-set to any value, so if an A3 PostScript printer or typesetting machine is available, alignments can readily be scaled to make best use of the extra space.

Each output page has three regions. The left hand edge contains identifying text for each sequence, the main part of the page holds the alignment, and the top part, the position numbers and optional tick marks. ALSCRIPT commands make use of a character coordinate system for font changes, and other formatting commands. Thus, any residue in the alignment may be referred to by its sequence position number (x-axis) and sequence number (y-axis), ranges of residue positions, or sequences may also be defined in the character coordinate system.

The basic ALSCRIPT commands allow the following functionality:

Fonts: Any PostScript font at any size may be defined and used on individual residues, regions or identifier codes.

Boxing: Simple rectangular boxes may be drawn around any part of the alignment. Particular residue types may be selected and automatically "surrounded" by lines. For example, if the characters "G" and "P" are selected, then lines will not be drawn between G and P characters, but only where G and P border with other characters.

Shading: Grey shading of any level from black to white may be applied to any region of the alignment, either as a rectangular region, or as residue specific shading. e.g. to "shade all Cys residues between positions 6 and 30"

Text: Specific text strings may be added to the alignment at any position and in any font or font size.

Lines: Horizontal or vertical lines may be drawn to the left, right, top or bottom of any residue position or group of positions.

Defaults: All defaults e.g maximum number/length of sequences, character spacing etc. may be modified using ALSCRIPT commands without the need to recompile the program.

Figure 1a[1] Illustrates a small section of a multiple sequence alignment in AMPS block-file format. The sequence identifier codes stored above the alignment have been deleted for brevity. In this example, the block-file also contains character-

[1] Figure1a.ps

based histograms representing the prediction of helix, strand and turn. Figure 1c[2] illustrates the result of running ALSCRIPT on this file using the commands shown in Figure 1b[3]. It is not suggested that this combination of options gives the clearest representation of the data, rather, the options have been chosen to illustrate many of the capabilities of the program.

Although written with the aim of producing figures for publication, ALSCRIPT is a useful research tool for interpreting multiple sequence alignments. For example, the boxing, shading and font changing facilities can be applied to highlight amino acids of a particular type and thus draw attention to clusters of positive or negative charge, hydrophobicity, and so on. Furthermore, computer programs for the automatic analysis of alignments can be made to produce ALSCRIPT formatting commands and a block file, thus simplifying the task of generating graphical representations of such analyses.

3 System Requirements and Availability

ALSCRIPT is written in C and should compile and run on most computers with a C compiler. An IBM compatible disk (1.44 MB) including the source code and a compiled version of the program for 386 DOS computers is available from the author.

4 Acknowledgements

The author thanks the Royal Society for support, and Professor L. N. Johnson for providing a stimulating working environment.

References

Barton, G. J. (1990). *Methods Enzymol.* **183**, 403–428.

Barton, G. J., Newman, R. H., Freemont, P. F. & Crumpton, M. J. (1991). *Eur. J. Biochem.* **198**, 749–760.

[2] Figure1c.ps

[3] Figure1b.ps

Barton, G. J. & Sternberg, M. J. E. (1987). *J. Mol. Biol.* **198**, 327–337.

Barton, G. J. & Sternberg, M. J. E. (1990). *J. Mol. Biol.* **212**, 389–402.

Devereux, J., Haeberli, P. & Smithies, O. (1984). *Nucl. Acid. Res.* **12**, 387–395.

Feng, D. F. & Doolittle, R. F. (1987). *J. Mol. Evol.* **25**, 351–360.

Higgins, D. G. & Sharp, P. M. (1989). *Comput. Appl. Biosci.* **5**, 151–153.

Lamport, L. (1986). LaTeX *a document preparation system*. Addison-Wesley.

Parry-Smith, D. J. & Attwood, T. K. (1991). *Comput. Appl. Biosci.* **7**, 233–235.

Russell, R. B., Breed, J. & Barton, G. J. (1992). *FEBS Lett.* **304**, 15–20.

Vingron, M. & Argos, P. (1989). *Comput. Appl. Biosci.* **5**.

5 Legend to Figure 1

Illustration of some of the formatting capabilities of ALSCRIPT.

(a) A small AMPS block file. Each aligned sequence is represented by a column. This block file includes character-based histograms that show the result of a combined secondary structure prediction method (Russell *et al.*, 1992). (b) An ALSCRIPT command file. Comment lines start with a # character, sections A-C set up general layout definitions: the ADD_SEQ command allows extra space to be inserted in the alignment at any location, 39 3 means add space for three sequences after sequence 39. PORTRAIT specifies that the output will be arranged with the longest side of the paper vertical. POINTSIZE sets the default pointsize for plotting characters. The DEFINE_FONT commands set up the fonts that are to be used and their pointsize; font 0 is used as the default. The pointsize may be set to the default value, an explicit value, or relative (REL) to the default size. Sections D–E show the boxing, shading and font changing commands in action. The IDENT_FONT ALL 6 specifies font 6 (Times-Roman) for all identifiers. SURROUND_CHARS LIV ALL specifies that all L I and V characters will be separated from other characters by a line (e.g. at sequence positions 19-21). FONT_CHARS KR ALL 4 switches the font of K and R characters to font 4 (Times-Bold 10 point). When regions are selected the coordinates are given in character units with the origin at the top left hand corner of the plotted alignment. For example, BOX_REGION 14

1 16 38 means box from residue 14 to 16 of sequences 1 to 38. These coordinates apply irrespective of how many pages are needed to plot the alignment at the chosen pointsize. The SHADE_CHARS command permits specific characters to be shaded with a grey value; The INVERSE_CHARS command allows white lettering to be superimposed on the shaded background, as shown for D and E. LINE TOP 2 40 8 draws a horizontal line at the top of the characters from residue 2 to 8 of sequence 40, similar commands allow lines at the left, right and below the defined residues; SUB_ID changes the identifier code for a sequence and SUB_CHARS allows a character to be substituted with another. In the example shown here, SUB_CHARS is used to remove the H, E and T characters that made up the prediction histograms and replace them with SPACE ‘ ’characters. The similar commands used to format the Sheet, Turn and Summary predictions are ommited for brevity.