ABOUT SSE

INSTALLATION

SSE USER INTERFACE

	Identify poor quality sequences --- Strip sequence gaps from alignment Create sequence fragments --- Create parsimony file --- Import sequence annotation Remove sequence annotation Annotate description with tag name Design sequence label Restore Original Sequence Names Create Column Labels from Annotation Split sequence into genes or peptides Split sequences using column labels Annotate with open reading frames --- Change Keyboard Mapping Purge keystroke memory Change label storage size
ENVIRONMENT	Reverse View Change Numbering Sound On / Off / Disable --- Change Screen Display Mode Edit Nucleotide / Amino Acid Colours --- Toggle Sequence Display Mode Change Reading Frame Display DNA/RNA Select Genetic Code --- New Column Label Edit Column Label Copy Column Label Delete New Column Label --- Mark Linear/Circular --- Switch files

RESEARCH	*Special Requirements* --- Sequence Distances Shannon entropy / Sequence Motif Scans Similarity Scan Composition Scan Sequence Changes --- Association Index Sequence Grouping Bootscan TreeOrder Scan --- Folding Energy Scan StructureDist Covariance Test --- Mutate Sequences Scramble Sequences
REFERENCES AND CITATIONS	
CONDITIONS OF USE	

ABOUT SSE Version 1.4

SSE developed out of a sequence editor that has been in use for entering and editing nucleotide sequences for several years in my research group. Increasingly, it has been used as a platform for a number of sequence analysis methods developed in the course of studies in viral sequence variation, and RNA secondary structure prediction.

SSE uses several embedded and external interfaces to other programs, such as the DPlot graphics package and MUSCLE and CLUSTAL for sequence alignments. Phylogeny and RNA structure prediction programs however use PHYLIP and RNAFold software that needs to be separately installed.

Registered users will receive notification of version changes when they are downloaded onto the website.

I hope you enjoy using the program, and that it is useful for your research. Do note that this will be last version of SSE distributed as a single integrated package. Version 2, currently under development will be distributed with separate 64-bit executables and compilable C++ code for most of the research functions and can therefore be used on multiple platforms. They will continue to interface directly with a now separate SSE editor but also can be called through external scripts, such as Python. This will remove the restriction of SSE functions to PCs and enable their use on Unix and Mac platforms.

Peter Simmonds

(Peter.Simmonds@ndm.ox.ac.uk)

<BACK TO DOCUMENT INDEX>

CHANGES IN VERSION 1.4

Remarkably, further testing and use of version 1.3 revealed a few further bugs and inconsistencies that have now been resolved. I am again very grateful to Donald Smith, University of Edinburgh for reporting these in the development of this program version.

Additional program features include a better interface with Genbank / ENA so sequence metadata can be directly imported and used to update the annotation fields of sequences within SSE. The annotations can then be used for assigning gene boundaries within alignments, for designing informative sequence labels and extracting coding region sequences. Sequences can now be imported through accession numbers that are identified and extracted from file contents or from the clipboard.

Sequences can be grouped into tag sets based on their divergence from each through user-specified thresholds. Other utilities extend the identification of similar or identical sequences, incomplete sequences, and those achieving a minimum sequence quality (lack of missing or unresolved bases, stop codons with coding sequences *etc.)*.

Research programs have been expanded, including updates to the Mutate Sequences program to enable mononucleotide composition to be altered under specified constraints such as maintaining dinucleotide frequencies and retaining coding. Listing of sequence changes has been greatly

expanded to incorporate information on the context where substitutions occur. The spacing of pre-specified nucleotide sequence motifs can be listed. The program StructureDist now displays a wider range of outputs including contour plots to depict RNA structural conservation / heterogeneity in alignments of sequences.

StructureDist and Folding Energy Scan are now based entirely on the RNAFold package and support for UNAFold has been discontinued. RNAFold is available from:

http://www.tbi.univie.ac.at/~ronny/RNA/index.html

RNAFold does not require a formal license or payment to download.

INSTALLATION

This program can be run on PCs running on any version of Windows up to and including the current version 10. This runs as a native 32 bit executable file, providing the program with access and editing capability of sequences files and databases of effectively unlimited size.

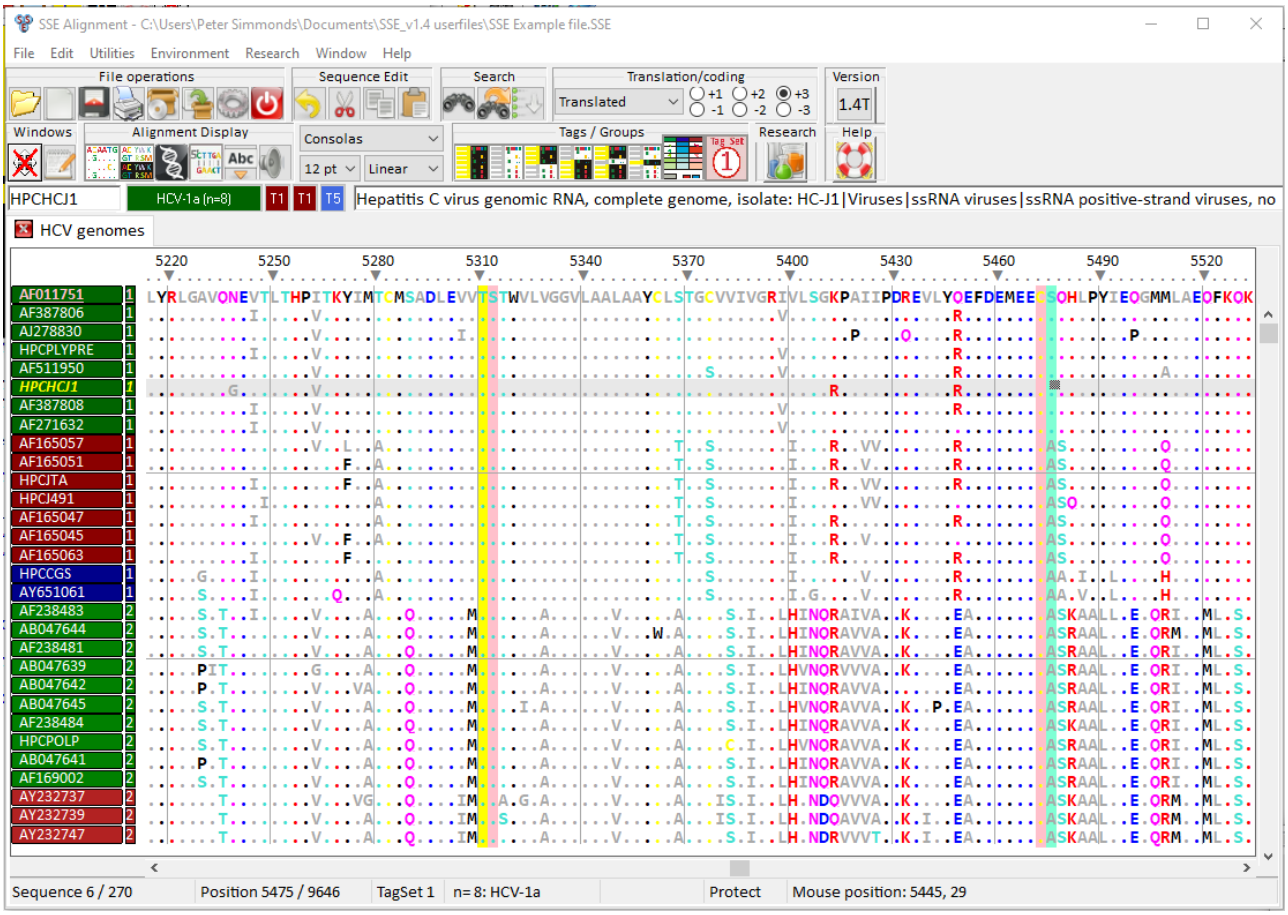
The file provided (SSE_v1.4_SETUP.EXE) is an executable that will install the program, graphics package and associated files and set up an entry on the start menu screen to run the program[1]. Installation requires that the user agree to the licensing conditions stated during the screen installation. Multiple instances of the program can be run at the same time so there is no need for multiple installations of the program.

<BACK TO DOCUMENT INDEX>

[1] I strongly advise you to uninstall any previous versions of SSE before installing version 1.4.

SSE USER INTERFACE

The program runs in a standard windows interface, with menu bars, button bars, status bars, scrollbars, context menus and tabs provided for viewing and editing a sequence alignment. The options on the menu bar (underneath the title bar) provide access to all the programs and utilities within the package. Common functions can be run from the button bar. Hovering over any button displays a tool tip describing its function, while right-clicking displays the appropriate section from the help file.



Many functions can be accessed through context menus that appear on block selection or right clicking on sequences or sequence groups on the editing screen. Right-clicking the tab header similarly allows it to be changed and the description edited.

To provide compatibility with the precursor to SSE, called Simmonics, almost all functions can additionally be run using keyboard shortcuts (listed in the menus and in the Helpfile). This generally involves pressing the <Alt> key (or sometimes <Cntrl>) with a keyboard letter. Where possible these conform to standard Windows shortcuts such as <Cntrl>A to select all sequences and <Cntrl>V to paste data from the clipboard into a sequence.

SSE uses the same filing system and generally the file browser interface as other Windows programs. This provides access to local drives, USB memory ports and both mapped and unmapped networked drives.

The basic environment when running the program comprises a screen display of the sequences, either as a list of sequence names or nucleotide or translated amino acid sequences. The interface

thus enables sequences to be edited manually, and new sequence data entered through the keyboard as required. The appearance of the display is controlled through a large number of selectable options in the "Environment" menu, and these settings are retained when the file is saved in SSE format.

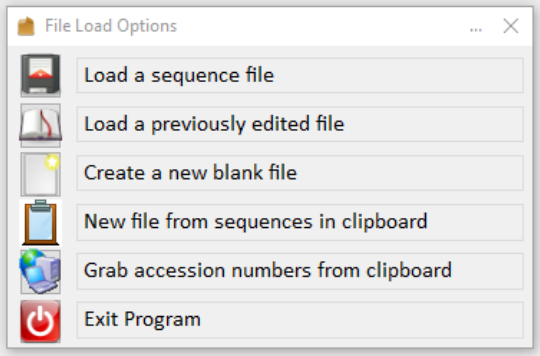
The large number of sequence manipulation and analysis options are accessed through the "Edit" and "Research" menus and operate on sequences selected in various ways by the editor. As will be described below, sequence operations can be carried out on individual sequences, on the whole dataset, or on sequence groups defined by Tags.

Tags are used to define sequences groups and are a key component to the interface. There are a total 27 coloured markers that can be assigned to sets of sequences. These can be separately named (eg. HCV - Genotype 1 as in the example file). The tag codes are displayed to the right of the sequence label, and assigned to individual sequences using context menus, the "Assign Name/Tag" command (<Alt>N) or as a more rapid shortcut by pressing the buttons <Alt>1 to <Alt>9 for green labels, <Alt>F1 to <Alt>F9 for red labels and <Cntrl>F1 to <Cntrl>F9 for blue labels. Sequence assigned to the same Tag group can be collectively selected as groups when carrying out any of the operations on the File, Edit or Research menus. Four independent sets of tags are available for more complex sequence and grouping tasks. Tags remain assigned to sequences if sequences are stored in SSE format.

Sequences or sequence blocks can be selected for editing or research functions using the mouse, through the tag select and context (right-click) menus, using pre-defined selections on the toolbar or by keyboard shortcuts. <BACK TO DOCUMENT INDEX>

STARTING UP

Starting the program presents an opening menu that offers five options. These are “Load a sequence file”, “Load a previously edited file”, “Create a new blank file", “New file from sequences in clipboard”, “Grab accession numbers from clipboard” and “Exit Program”.



LOAD A SEQUENCE FILE

You will now be prompted for a filename. The directory it initially chooses to look for files is in the program user directory: My Documents\SSE_v1.4 Userfiles\, and in this directory it looks for filenames ending in ".SSE" (or ".SIM"). The default filename ".SSE" differentiates files in SSE format from those used for other sequence analysis programs, such as FASTA and GCG.

The SSE file format was specially designed to hold sequence data, and also to provide a particular sequence environment for sequence editing. This includes information on how the sequences are numbered, whether they are translated (and reading frame), whether they should be viewed in reverse complement. It also preserves sequence tags[2] used to group sequences. The format also allows a fuller description of each sequence. This sequence description allows dates, batch numbers and accession numbers etc to be added and accessed during editing. When sequences are imported from GenBank or EMBL, the sequence description is automatically loaded with the annotation provided with the sequence. <BACK TO STARTING-UP OPTIONS>

FILE FORMATS

Sequence data in a file or held on the clipboard will be loaded into the editor that contains an input filter module that automatically recognise nucleotide or amino acid sequences in a variety of standard formats. These include PHYLIP and MEGA sequence alignments, as well as single or multiple sequences in FASTA, FASTQ, PIR, NEXUS (Nucleotide or DNA data types), GenBank and EMBL formats.

The GenBank and EMBL filters are particularly useful as they allow sequences from these databases to be directly imported. The filter will also try to obtain useful sequence information from the entry and incorporate it into the sequence description, so that properly annotated sequence files can be built up. These are appended to the description line, with the “|ORG=” delimiter, while the remainder of the record is imported into the sequence annotation. It similarly uses the parts of the extended sequence name used in FASTA format

[2] Sequence tags are extensively used in SSE to label groups of sequences for various sequence operations. Their use is described in detail in the “Sequence Tagging” section.

for the sequence description. The sequence source imports the annotation provided as a header to the sequence file. Further annotation-based options for importing GenBank and EMBL records are offered, such as importing annotated gene or peptide sequences within each record rather than the whole sequence.

Sequence files in PHYLIP or MEGA format can be interleaved, and this should be detected on loading. Both filters are relatively tolerant of errors in the sequence file, such as uneven lengths of lines, strange nucleotide codes or symbols or different sequence names in different interleave blocks. It is possible to make such an awful file that the load process fails, although this should not be able to crash the program and will probably be reported by the error handler as "Unknown format". <BACK TO STARTING-UP OPTIONS>

LOAD A PREVIOUSLY EDITED FILE

This option allows one from the last 10 previously used sequence files to be loaded from its original location. This is a convenient alternative to specifying files using the standard file browser. SSE keeps an automatic record of the files that have been edited, and updates this on each Load or Save operation (placing the last edited file at the top of the list). Files with the same name stored in different directories or drives are listed separately. Files names that are greyed out in the list are not available for loading (they may have been moved or deleted). <BACK TO STARTING-UP OPTIONS>

CREATE A NEW BLANK FILE

This option is designed to produce a blank sequence file that can be directly typed into. This might be useful for setting up a file to enter sequences manually. Initially, there is only one sequence in the file, of length 300 bases, and no name. Additional sequences can be added or imported during subsequent editing, and each one named and described using the appropriate commands. Typing in sequences beyond 120 bases (with <INSERT> on) allows the sequences to be lengthened.

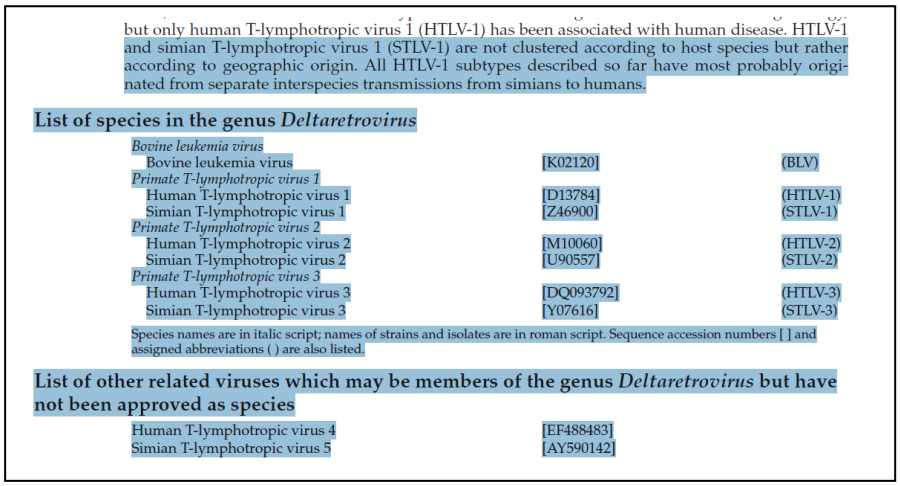
The blank template file produced by CREATE FILE option can also be used when sequences are to be imported from the clipboard. In this case, sequences should be formatted into PHYLIP, MEGA or the other formats recognised by the LOAD FILE filter. Alternatively, as described below, the contents of the clipboard can be directly pasted into the sequence line (<Cntrl>V). <BACK TO STARTING-UP OPTIONS>

NEW FILE FROM SEQUENCES IN CLIPBOARD

This is similar except that a new file is created by importing the contents of the clipboard rather than from a file (assuming it is in a recognizable format). <BACK TO STARTING-UP OPTIONS>

GRAB ACCESSION NUMBERS FROM CLIPBOARD

This input option greatly simplifies the construction of sequence datasets from sequence lists and tables that use GenBank/EMBL/DDBJ or Refseq accession numbers. SSE will identify and extract accession numbers from other extraneous text and formatting. This clipboard copy from a chapter of the ICTV Report:



but only human T-lymphotropic virus 1 (HTLV-1) has been associated with human disease. HTLV-1 and simian T-lymphotropic virus 1 (STLV-1) are not clustered according to host species but rather according to geographic origin. All HTLV-1 subtypes described so far have most probably originated from separate interspecies transmissions from simians to humans.

List of species in the genus *Deltaretrovirus*

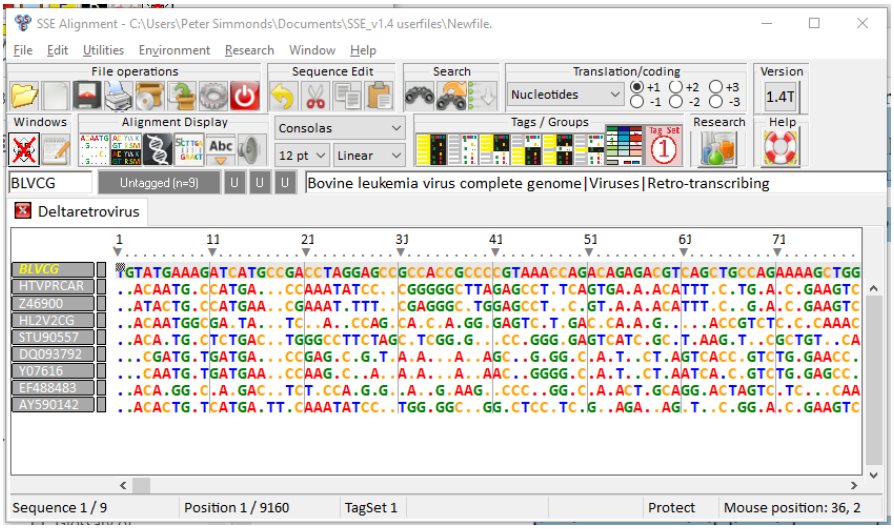
Bovine leukemia virus		
Bovine leukemia virus	[K02120]	(BLV)
Primate T-lymphotropic virus 1		
Human T-lymphotropic virus 1	[D13784]	(HTLV-1)
Simian T-lymphotropic virus 1	[Z46900]	(STLV-1)
Primate T-lymphotropic virus 2		
Human T-lymphotropic virus 2	[M10060]	(HTLV-2)
Simian T-lymphotropic virus 2	[U90557]	(STLV-2)
Primate T-lymphotropic virus 3		
Human T-lymphotropic virus 3	[DQ093792]	(HTLV-3)
Simian T-lymphotropic virus 3	[Y07616]	(STLV-3)

Species names are in italic script; names of strains and isolates are in roman script. Sequence accession numbers [ ] and assigned abbreviations ( ) are also listed.

List of other related viruses which may be members of the genus *Deltaretrovirus* but have not been approved as species

Human T-lymphotropic virus 4	[EF488483]
Simian T-lymphotropic virus 5	[AY590142]

Will directly import the listed sequences along with the GenBank metadata for each sequence, all ready to be aligned and analysed:



<BACK TO STARTING-UP OPTIONS>

EXIT PROGRAM

Changed your mind, and decided not to use the ultimate software package? OK, if you're going to be boring then go ahead! <BACK TO STARTING-UP OPTIONS>

PROGRAM RE-ENTRY

The other possible opening menu is one that appears after a program crash. While SSE is running, it sets up backup files as editing proceeds. If the program is exited correctly, these are cleared away, but they remain on the hard disk after a crash. If these are detected when the program is restarted, it is possible to load the last file edited, which ensures that (possibly valuable) data can be recovered. The options in this box are Recover sequences, which will present a second box listing the sequence files available to be loaded. These will need to be re-saved (using the default "Recovered" suffix to the file name before they are loaded. Quitting at this stage does not delete the backup files. <BACK TO STARTING-UP OPTIONS>

INSTALLING PHYLIP AND RNAFold

The PHYLIP and RNAFold packages provide phylogenetic analysis and RNA structure prediction functions that are used by the SSE editor and Research programs. These include sorting, Association index and sequence grouping calculations, calculation of RNA folding energies and secondary structures.

PHYLIP and RNAFold can be downloaded from:

http://evolution.genetics.washington.edu/phylip/getme-new1.html
http://www.tbi.univie.ac.at/~ronny/RNA/index.html

Because these have to be installed separately from SSE, these accessory programs cause the most problems for users of the SSE package. Essentially, SSE needs to know where PHYLIP and RNAfold programs are installed before they can be used. On initial loading of SSE it will attempt to locate them on the C:\ or D:\ drives. However, there are many configurations and possible locations of program files on computers and this search operation may be unsuccessful. If you encounter error messages about being unable to locate these accessory programs, you will need to manually browse for locations in the "Default File Location" menu item in the File menu.

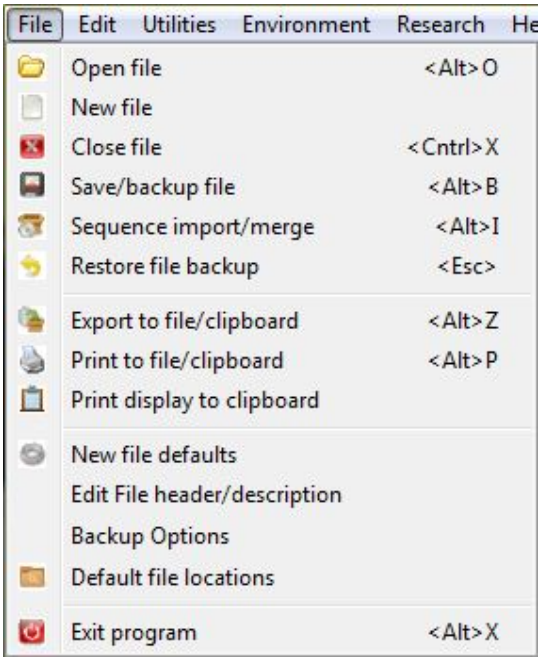
All versions of PHYLIP can be used, although some of the programs in the latest available version (3.69) run much faster than version 3.5 and below. If multiple versions of PHYLIP are installed, the directory will have to be manually entered for the version you want to use.

<BACK TO STARTING-UP OPTIONS>

<BACK TO DOCUMENT INDEX>

FILE OPERATIONS

The following commands are accessed through the File box on the menu line, and include operations that allow loading, saving and exporting sequences to and from SSE file format to external files either for storage or for further analysis by other programs. Files saved by Backup or on exiting the program are saved in SSE format, while a variety of formats can be chosen using the Export command.



Using tags to define sequence groups, Print and Export operations are preceded by a box that defines the scope of the operation (*ie*. individual sequence [on cursor line], sequence group [defined by tag of cursor line], or all sequences). File save on exit and Backup save all sequences but preserve the sequence group names and tags on re-loading.

Command	**Shortcut**	**Scope**[3]

OPEN <Alt>O (A)

This allows one or several files to be loaded into the editor, in addition to the file (or files) loaded on program start-up. Up to 32 files can be loaded at any one time. The users can switch between open files by clicking on the tabs at the top of the alignment display or pressing <CNTRL><TAB> or <CNTRL>W repeatedly to cycle through them. <RETURN TO FILE COMMANDS>

NEW (A)

Creates a new blank file <RETURN TO FILE COMMANDS>

CLOSE FILE <Cntrl>X (A)

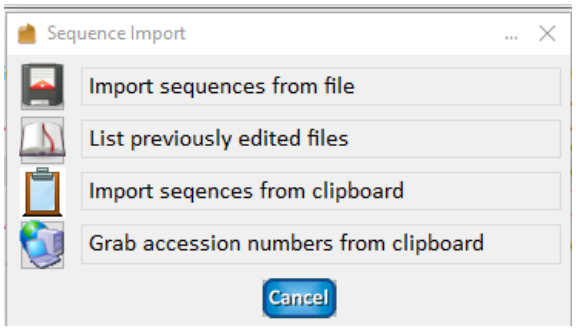
Closes an individual file (and provides the option to exit program if this was the only file loaded in the editor). The name and location assigned to the file to saved uses the standard Windows file browser. <RETURN TO FILE COMMANDS>

SAVE / BACKUP FILE <Alt>B, F10 (A)

Backup provides the means to save the current active file with a choice of filename. All sequences are saved in SSE format. This command is useful to carry out during intensive spells of editing to ensure that sequence information is not lost, although the program will automatically maintain a series of internal backup files that can be re-loaded should anything unfortunate happen to the program(!), or the user do something ill-conceived, regrettable, or frankly wrong to the sequences. <RETURN TO FILE COMMANDS>

[3] This refers to the range of sequences to which the operations can be applied; A = All sequences, T = A tagged group, S = Single sequence

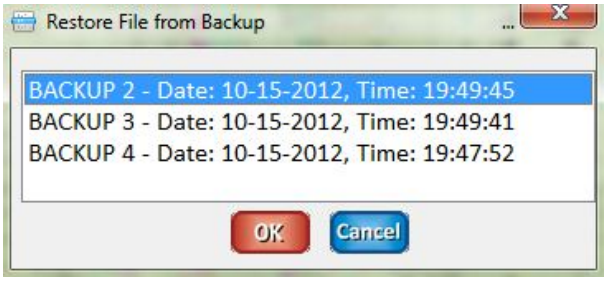
SEQUENCE IMPORT / MERGE <Alt>I, <Shft>F10



Import allows nucleotide or amino acid sequences held in the clipboard or in a different file to be added to those currently being edited. Options are provided to load in one or more new sequence files using the file browser ("Import sequences from file"; press <CNTRL> for multiple file selections) or to select a file from the list of the 10 edited file ("List previously edited files"). Alternatively, sequence data can be directly imported from Clipboard ("Import sequences from clipboard") or accession extracted from Clipboard text ("Grab accession numbers from the clipboard")

Appending sequences provides the means to construct large composite sequence files from different sources (and sequence formats). Thereafter, loading proceeds with boxes showing the status of the importing process (if non-SSE format), followed by a box requesting confirmation that the sequences should be loaded. Imported sequences are added to the end of the file, and may require aligning and sorting before they are useful. <RETURN TO FILE COMMANDS>

RESTORE FILE BACKUP <Esc>, <Cntrl>Z (A)



Undo editing / procedures using previous backups. One of a list of previously stored previous versions of the alignment file can be reloaded from the menu box. The number of backup files and frequency of storing them is specified in "Backup Options". <RETURN TO FILE COMMANDS>

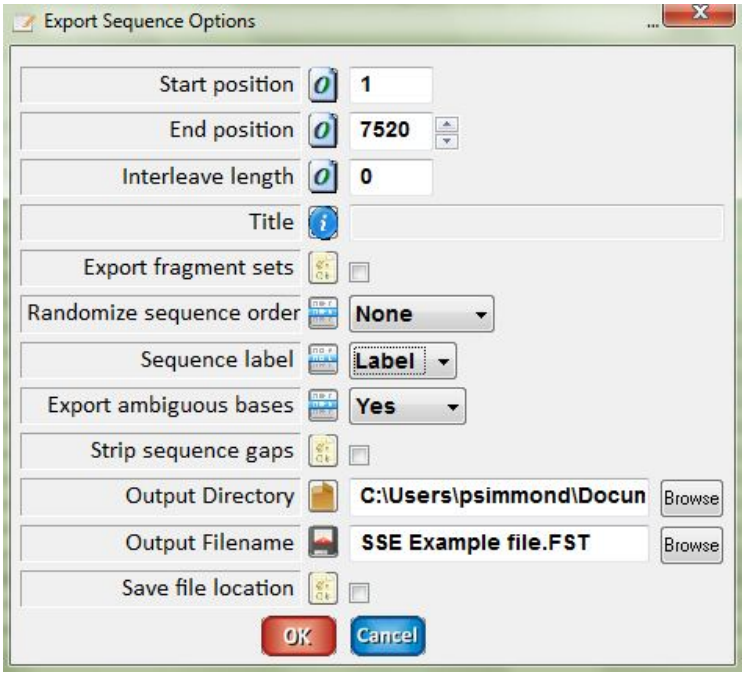
EXPORT TO FILE <Alt>Z S, T, A

This command allows sequences to be exported in a variety of standard formats, thereby providing an interface to a number of sequence analysis packages such as PHYLIP and MEGA. Sequences can be exported either to a file, or onto the clipboard, selectable by the initial option box.

The following cascade of menus controls the export process:

- Sequence selection (single sequence on cursor line, tag group, selected or All).
- Destination of sequences (file, clipboard)
- The file format (SSE, PHYLIP, MEGA, FASTA, NBRF/PIR, CLUSTAL, NEXUS, Single sequence files or Old SSE format (.SIM)
- The sequence type (nucleotide or amino acid)
- A final menu for specifying the sequence range, interleave length (as appropriate), title, the capability of exporting sets of fragments of defined sixe and increment over the sequence range and whether sequences should be randomised in order (including the possibility of uncoupling sequence from their names if required). Other options include exporting tag names instead of sequence names, stripping sequence gaps from the alignment (FASTA, PIR, NEXUS and CLUSTAL formats only) and

finally, if file output was chosen then file location and filename. The file location and filename can be stored (last line in option box) to create a default export location for each file format.

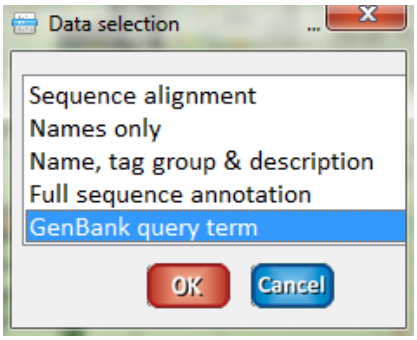


Exporting sequences in SSE format is useful for transferring selected sequences between different SSE files, as this preserves tagging information and descriptions of each sequence.

Exporting sequences in PHYLIP, MEGA, NBRF/PIR or FASTA formats allows sequences to labelled with their sequence group name (or a coded tag description if unlabelled). (a condensed form is used for PHYLIP format files because of its 10 character length limit for sequence names). <RETURN TO FILE COMMANDS>

PRINT SEQUENCES <Alt>P, <Shft>F7 S, T, A

Sequences, sequence labels or annotations in the alignment can be printed to file or to the clipboard in a variety of formats. After sequence selection, SSE provides five option boxes that determine the format of the printed sequences. The file produced uses standard ASCII codes for the text and extended codes for the box-drawing commands if selected.



The first options select whether the sequences themselves, their names (as a comma separated list), sequences and descriptions or a full sequence record should be printed. The final option generates a formatted query string for use for GenBank searches

If sequences are selected, these may outputted as alignments of (N)ucleotides, (A)mino acids or (B)oth. The latter format produces sequence listings where both the nucleotide and amino acid sequences are printed on successive lines, with the nucleotides split into codons. The reading frame for both the (A)mino acid and (B)oth printouts use the reading frame selected in the editing screen. The page width of the sequence printout can be specified, as well as the ability to format the sequences internally into blocks of a user-defined width, and to surround sequence blocks by boxes (line 9).

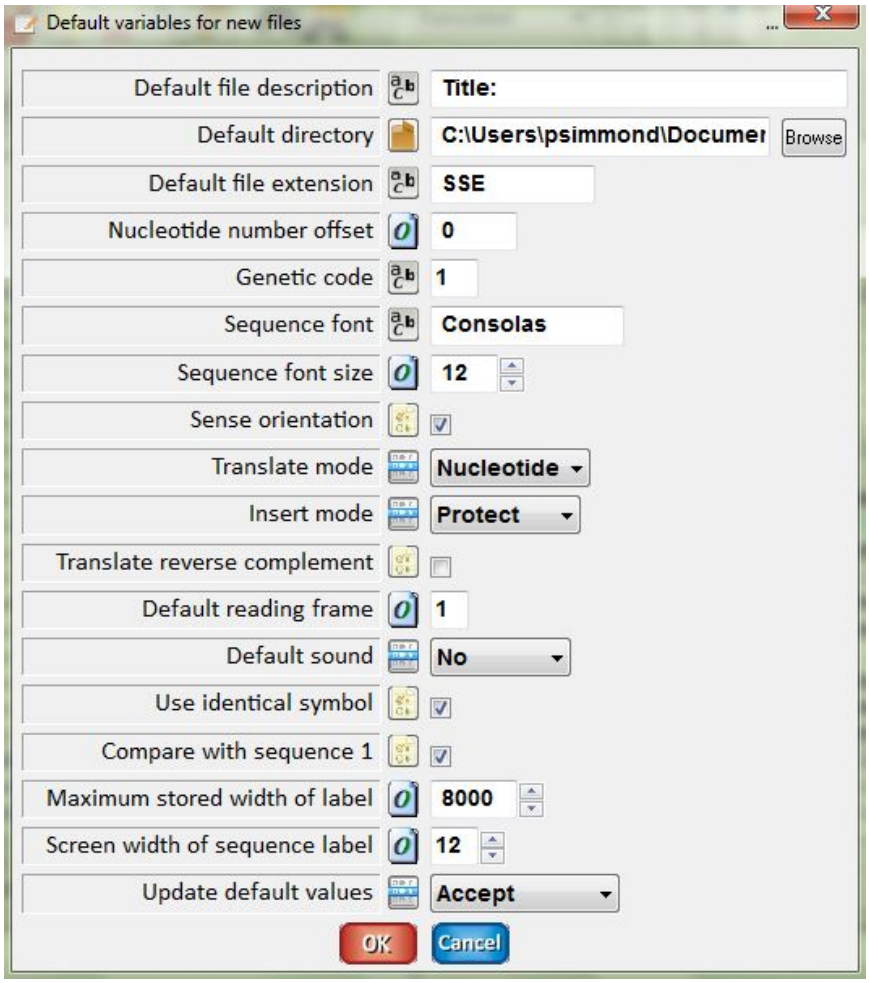
The third box allows the sequence range, and numbering to be specified. The printout can be numbered using the settings of the editing screen as defaults (increment or decrement numbering number of first nucleotide or amino acid). <RETURN TO FILE COMMANDS>

PRINT DISPLAY TO CLIPBOARD

The print the displayed screen with currently displayed formatting, colours and sequence numbering to the clipboard. This can then be pasted as a bitmap into a graphics program or Paint for further editing and saving. For a more flexible text representation of the alignment, use “Print Sequences” . <RETURN TO FILE COMMANDS>

NEW FILE DEFAULTS

This option produces an input box that allows user input on the default file title, location of filenames, file name extension (*eg.* "*.SSE") and editing environment for sequences loaded in non-SSE format (largely self-explanatory so not repeated here). The final line allows these settings to be re-loaded from the current settings, from a series of basic default values, and to be stored for subsequent loading of files. These file settings are overridden if a sequence file in SSE format is loaded, as all options are contained within the SSE formatted file. The settings chosen are stored in the file "NEWFILE." that is set up in the program directory on first running. <RETURN TO FILE COMMANDS>



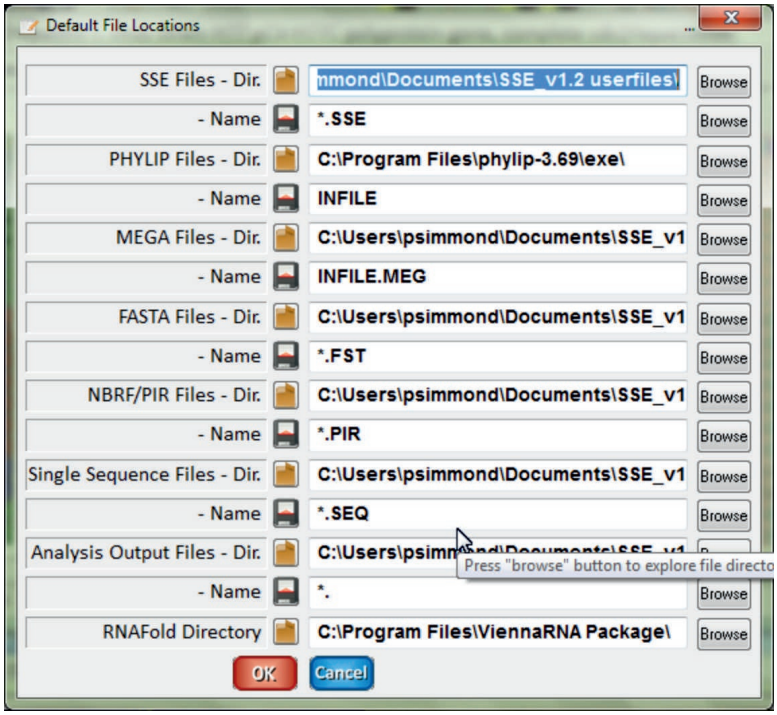
EDIT FILE HEADER/DESCRIPTION <Alt>W

This option allows the entry of a full description of the sequence file and a brief summary that appears on the tab header on the editing screen for identification. On creating a new file, the user will be automatically asked to enter a file header. <RETURN TO FILE COMMANDS>

BACKUP OPTIONS None (A)

Backup options allows the user to specify how many backup files are stored during editing (default 5), and how frequently they are backed up in seconds (range 30-300). These values should not normally need to be changed although disabling or restricting the number of backup copies may be required when editing gigantic files. The options selected are specific to the alignment file, so that backup can be specifically disable for large files loaded into SSE. <RETURN TO FILE COMMANDS>

DEFAULT FILE LOCATIONS None (A)



This allows the default directories for loading and saving sequence files, importing and export sequences in a variety of formats to be directly specified, and avoids having to set these default values when running through individual analysis programs that use them.

It also allows the locations of PHYLIP and RNAFold program files to be entered. These programs can be searched for and located by entering "S" on the appropriate option line, and then pressing OK. This is useful if they have been installed or upgraded to new versions since the installation of SSE. <RETURN TO FILE COMMANDS>

EXIT PROGRAM <Alt>X, F7 (A)

Fed up with editing sequences? It may be time to exit SSE and get a life!! Press <Alt>X or F7 and an option to save each open file as the exit process proceeds. The filename can be changed if required, although the file location and filename default to those of the originally loaded or imported file. Thereafter, SSE makes a feeble gesture to interest you in loading another sequence file to edit, or to return to the editing screen of the current file. Ignore these, choose option 5 and switch the computer off! <RETURN TO FILE COMMANDS>

<BACK TO DOCUMENT INDEX>

THE EDITING SCREEN

The editing screen allows sequence information to be directly typed into a file in a user friendly way (the original purpose of SSE), and for these sequences to be aligned, compared with other sequences, printed, exported to other sequence analysis packages, and stored.

MOVING AROUND SEQUENCES.

The cursor position represents the point where sequences appear upon keyboard input, and is the starting point for many of the manipulations that can be carried out on sequences. As well as using the mouse and arrow keys to move the cursor and the scrollbars to move within a file, several keyboard shortcuts are provided:

<Cntrl><Right arrow>	Move cursor right 10 bases
<Cntrl><Left arrow>	Move cursor left 10 bases
<Cntrl><Up arrow>	Move cursor up 10 lines
<Cntrl><Down arrow>	Move cursor down 10 bases
<Alt><Right arrow>	Move cursor right by one screen
<Alt><Left arrow>	Move cursor left by one screen
<Alt><Up arrow>	Move cursor up one screen
<Alt><Down arrow>	Move cursor down one screen
<Shft><Right arrow>	Drag sequence (and cursor) to the right
<Shft><Left arrow>	Drag sequence (and cursor) to the left
<Shft><Cntrl><R.arrow>	Drag sequence one base right in Amino acid display
<Shft><Cntrl><L.arrow>	Drag sequence one base left in Amino acid display
<Shft><Up arrow>	Drag sequence (and cursor) up one line
<Shft><Down arrow>	Drag sequence (and cursor) down one line
<PgUp>	Move up 10 lines
<PgDn>	Move down 10 bases
<Home>	Move to start of sequence
<End>	Move to end of sequence
<Cntrl><PgUp>	Move up to first line of file
<Cntrl><PgDn>	Move to last line of file
<Alt>G	Produces an input box which allows sequence position and sequence line to be directly entered (can also be selected from Environment menu).
<Del>	Delete base below cursor[4]
<BckSpc>	Delete base behind cursor

<RETURN TO EDITING SCREEN OPTIONS>

[4] Action depends on INSERT state (see below)

INSERT, OVERWRITE AND PROTECT EDITING MODES

The insert key toggles between three rather than two states. Insert and Overwrite states determine whether bases entered from the keyboard are inserted between or replace bases on the cursor line. The third state (Protect) is designed to allow sequence editing and manual alignment without running the risk of overwriting or deleting sequence information. In this mode, only unknown bases ("N") or alignment gaps ("-") can be entered or deleted. The mode is indicated on the status bar and by the cursor shape as follows:

Overwrite	Large rectangular cursor
Insert	Underline cursor
Protect	Grey cursor

<RETURN TO EDITING SCREEN OPTIONS>

AUTO-BACKUP

SSE maintains running backups of the sequence file as editing proceed. This operation is invisible to the user and the sequences produced are cleared when the program is exited correctly. The running backups allow the effect of editing changes to the sequences to be reversed. Previous versions of the sequence file can be recovered after potentially drastic operations such as merging files, sorting or aligning sequences. It also provides the means to recover from interruptions to editing such as a network problem, power failure or program crash.

Backup files are accessed during editing by pressing <Esc>, which will present a list of the backup files available and the dates and times they were stored (pressing <Esc> itself leads to the creation of a backup file, which is the first in the list but cannot be selected). The same menu is presented upon restarting SSE if it was not exited correctly (see above).
<RETURN TO EDITING SCREEN OPTIONS>

INPUT OF NUCLEOTIDE AND AMINO ACID SEQUENCES

The primary input data type is nucleotide sequence (GATC) and amino acids. The following characters can be typed from the keyboard, and will appear at the cursor position:

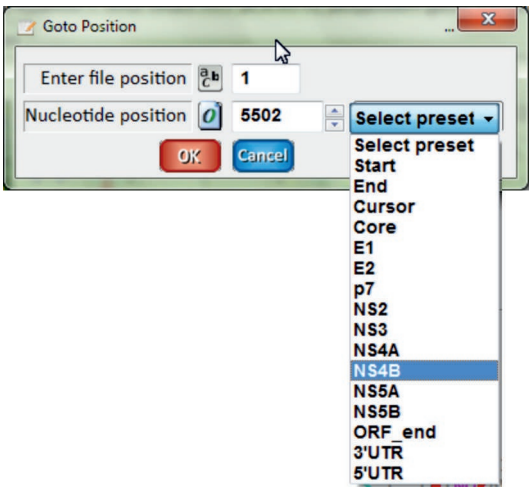
G	Guanosine	G
A	Adenosine	A
T/U	Thymidine/Uridine	T/U
C	Cytidine	C
Y	Pyrimidine	C T/U
R	Purine	A G
S	Strong	G C
W	Weak	A T/U
M	Meta	A C
K	Keto	G T/U
B	Not-A	C G T/U
D	Not-C	A G T/U

H	Not-G	A C T/U
V	Not-T/U	
N	Unknown or Any	A C G T/U
-	Alignment gap	None

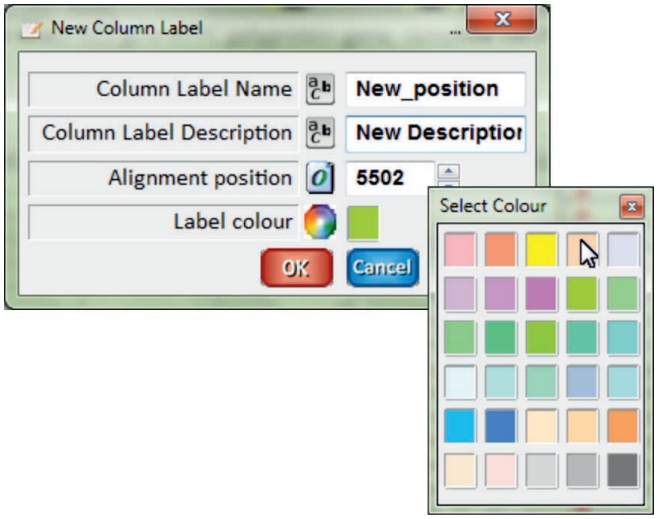
Alternatively, in Amino Acid display mode, amino acids can be entered via the keyboard. Unknown or incomplete residues are entered and displayed as “x”.

Note that "N" or “x” is treated differently from "-", as the former really means that there is base or amino acid there but its identity is unknown, while a hyphen means that a gap has introduced into the sequence normally to preserve alignment with other sequences). Although "N" is entered using the "N" key on the keyboard (or a user-defined key), it is by default represented on the screen as a blank to reduce clutter (this can be changed from the Screen Display menu). Attempts to enter letters other than those listed above from the keyboard are accompanied by an error sound, and are not allowed. <RETURN TO EDITING SCREEN OPTIONS>

COLUMN LABELS



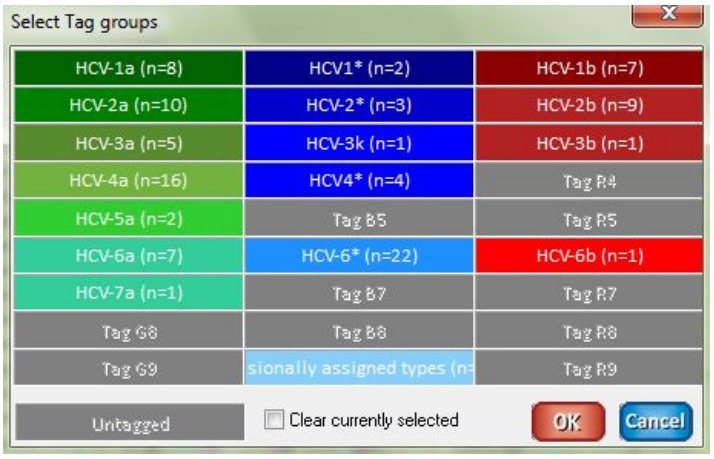
Nucleotide or amino positions can be labelled as part of the annotation of a sequence alignment. This provides a convenient navigation tool and the means to specify preset values in those functions that require sequence positions to be specified, listed in a drop down menu Separate sets of labels are provided for nucleotide and amino acid numberings. Column labels are identified through a unique name; they also have an additional field for an optional, longer description if necessary. Column labels can be marked in alignments in one of several different colours selected from the Label menu.



Column labels can be added through selection of “New Column label” in the Environment menu, or right clicking at the selected column in the alignment and choosing “Add Column Label” from the context menu.

Existing column labels can be edited, copied, sorted and deleted through the Environment menu or through context menus invoked by right clicking on the column. <RETURN TO EDITING SCREEN OPTIONS> <BACK TO DOCUMENT INDEX>

SEQUENCE GROUPS AND SELECTION

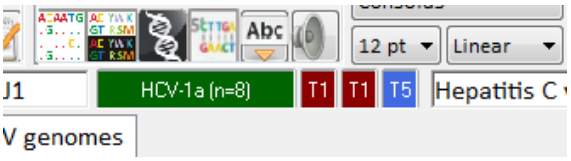


SSE can group sequences together allowing sequence sets to be selected together, labelled with group names and used for classification for various analysis programs. Sequence selection provides the means to temporarily define sets of sequences to be edited together or used for analysis.

There are several ways to select sequences. Firstly, sequences can be highlighted by dragging the cursor over the sequences required and choosing "Select sequences" from the context menu. The same procedure can be used to define a sequence block that is then pre-defined in subsequent menus. Right-clicking on a sequence provides further selection options, including selecting all members of a tag group, de-selecting individual sequences or sequence groups *etc.* The multi-select button on the toolbar (or drop-down menu) provides the means to select several sequence groups and de-select current selection if required. Finally, six buttons on the toolbars (each with a keyboard shortcut) allow rapid selection or de-selection of pre-defined groups (All, none, sequences above cursor, below cursor *etc.*). The names of selected sequences are shown in inverse colours, as are selected bases or amino acids in a sequence block.

Tag groups comprise a total of 4 independent sets of 27 different (user-defined) sequence groups. The labels of tagged sequences are shown in shades of blue, red and green while untagged sequences are shown in grey. Tag groups allow independent assignment of sequences to separate sets of tag groups, providing alternative methods to group or classify sequences within an alignment.

The tag set to be used can be selected directly from the main screen by clicking on the appropriate button (the currently selected group is the larger button). There is a similar set of button in the Edit Sequence Label window.



Alternatively, the Tag Select button can be used and this produces a list of the Tag sets and their annotation. These can be used, for example, to provide information on why tag group assignments were made (as shown on the right).

Individual tag groups within a set can also be named, and a fuller description entered from the context menu or menu list. The group name for the sequence on the cursor line is displayed on the status bar. The tag codes (1-9) for each sequence is displayed on the between the sequence label and the start of the alignment.

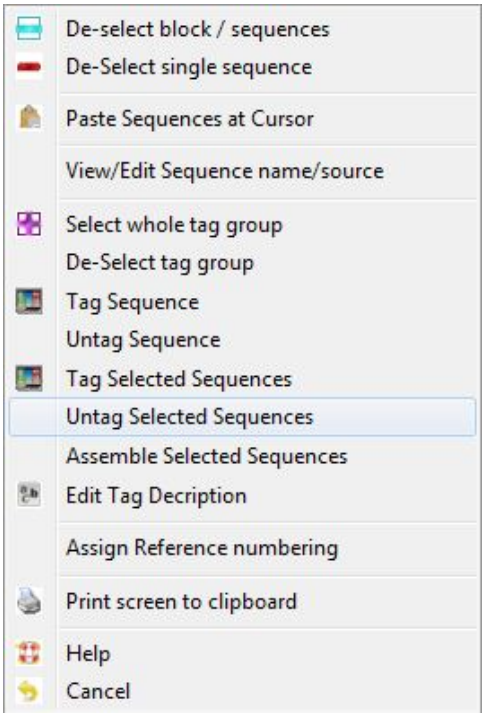
Tags can be assigned from the Edit menu using the Tag/Label edit command. As a keyboard

shortcut, directly by positioning the cursor on the sequence line and pressing any of the following key combinations assign the following tags:

<Alt>1, <Alt>2 ... <Alt>9	Tag sets 1 to 9 (green)
<Alt>F1, <Alt>F2 ... <Alt>F9	Tag sets 11 to 19 (red)
<Cntrl>F1, <Cntrl>F2 ... <Cntrl>F9	Tag sets 21 to 29 (blue)

<BACK TO DOCUMENT INDEX>

CONTEXT MENUS



Right clicking on the editing screen creates one of a number of context menus that provide convenient shortcuts to various sequence manipulations and functions. Many of these are also available on the main drop-down menus and some are on toolbar buttons. Several however are additional to the standard functions and are briefly summarised here:

SELECT SEQUENCES. This allows manual selection of sequences, either the sequence or the cursor line or a set of sequences highlighted by the selection carat.

SELECT BLOCK.
This option is available only in nucleotide or amino acid displays and allows both sequences and the highlighted nucleotide or amino acid sequence to be selected. The selected sequence block is highlighted in reverse video; the boundaries become default start and end positions in subsequently called functions that use this information.

DE-SELECT BLOCK/SEQUENCES. This option is available on right-clicking on currently selected sequences and allows them all to be de-selected

DE-SELECT SINGLE SEQUENCE. This option is available on right-clicking on a currently selected sequence and allows it to be manually de-selected

COPY / CUT / CLEAR / DELETE BLOCK. The highlighted sequence block can be copied or cut to the clipboard. Alternatively the sequence range can be cleared or deleted from the alignment.

SELECT / DE-SELECT TAG GROUP. All sequences within the tag group of the sequence on the cursor line or in the current sequence selection can be added to the selection or collectively de-selected.

TAG / UNTAG SEQUENCE. The sequence on the cursor line can be assigned to a specified tag group or de-assigned from its currently assigned tag group.

ASSEMBLE SEQUENCE. This creates a contig from selected or highlighted sequences with options to keep or delete source sequences and inherit their tag group assignments. The annotation of the contig lists the component sequences. Sequences that are assembled are checked for conflicting bases; if detected, the option to represent these by a simple consensus base or a gap is provided. For a greater range of assembly options, use the ”Sequence Join” / Create Consensus) function.

COLUMN LABELS. Right-clicking on a column label will create a context menu with additional options to edit, copy or delete the column label. See Column Labels for further information.

<BACK TO DOCUMENT INDEX> .

THE NAVIGATOR SCREEN

SSE provides a map of the whole sequence alignment in a separate window. This allows large files to be visualised and navigated with a "bird's eye" view of the file contents. The navigator screen displays sequences in a foreground colour (default yellow) against a (default) black background representing sequence gaps or unknown bases. Column labels are shown in the colours assigned to them in the editor screen and assist in navigation.

NAVIGATION

The part of the alignment shown in the sequence editor window is indicated as a box in the Navigator Window. This rectangle and the editor window are linked. Firstly, the Navigator display updates as the user moves around the alignment in the editor. Secondly, clicking or dragging the rectangle in the Navigator move the display in the sequence editor as follows:

Left Click. Centres the rectangle (and the alignment region in the editor window) approximately to the position of the mouse pointer

Left button Drag. Attaches the rectangle to the mouse pointer allowing the whole alignment to be actively scanned and visualised in the editor window.

Right Click. Moves the rectangle one rectangle width or height towards the mouse pointer position. <BACK TO NAVIGATOR OPTIONS>

Several toolbar and mouse options are available to modify the size and appearance of the navigator screen (from left to right):

RESIZE NAVIGATOR WINDOW

Two buttons are provided on the toolbar line to increase or decrease the size of the Navigator window in steps. Alternatively, the window can be resized by dragging the box edges to the desired width and height. <BACK TO NAVIGATOR OPTIONS>

PRINT DISPLAY TO CLIPBOARD

Copy Navigator screen contents to the clipboard as a bit map, that can be pasted into other programs. <BACK TO NAVIGATOR OPTIONS>

CHANGE FOREGROUND AND BACKGROUND COLOURS

Buttons labelled “F”(oreground) and “B”(ackground) allow screen display colours to be changed using the standard colour selector menu. Selected colours are stored as attributes of each alignment file. <BACK TO NAVIGATOR OPTIONS>

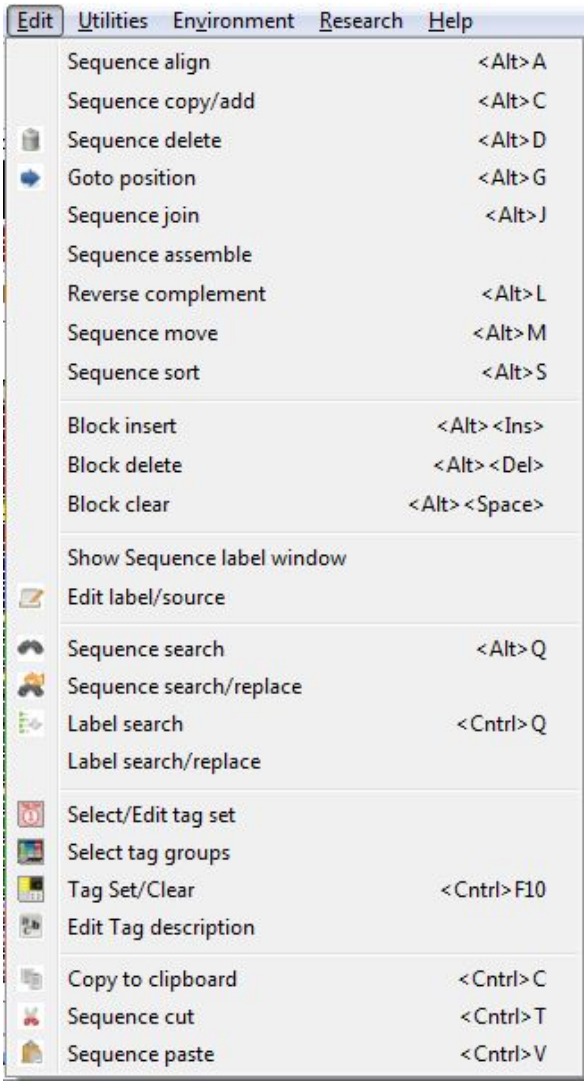
HIDE / SHOW NAVIGATOR

This hides the display. Showing or hiding the display is a stored attribute of each alignment file (for example, some containing very few sequences are not usefully displayed with the Navigator Window). A hidden Navigator window can be re-displayed from the Environment pulldown menu or from the toolbar in the sequence editor. <BACK TO NAVIGATOR OPTIONS>

<BACK TO DOCUMENT INDEX>

SEQUENCE EDITING COMMANDS

The next set of commands allows various editing actions and modifications to be carried out on individual sequences. Each can be selected from the Edit box (<Alt>E) although there are keyboard shortcuts for most of them. Many commands can be carried out on either the sequence on which the cursor is placed, extended to sequences in the same tagging group or from selected sequences. For selected sequences, commands such as Align, Delete, Move, Insert and Clear Bases can be made to apply as a group, or even to all sequences in the file. Exactly which sequences are to be altered by the commands listed below are usually selected by a dialogue box before the command is run.



Command	**Shortcut**	**Scope**[5]
ALIGN	<Alt>A	S, I

A) *Align to best sequence match*.

This command provides a rapid method to align a sequence with others in the file. Sequence matching is made between each selected sequence on the cursor line (query sequence) and to target sequences, which may be either the reference sequence on the top line of the alignment or to all sequences in the alignment. Several search parameters to be specified, including the region in the target sequence to be searched for a match, the size of the search fragment in the query sequence, and the size of the increment used to generate successive search fragments (starting from the 5' end). The "minimum match" is the degree of similarity above which a match is considered acceptable as a match, and when found will direct alignment of the query sequence to that match position. The "maximum match" is the degree of similarity sufficient to align the query sequence to the target sequence and discontinue searching for better

[5] This refers to the range of sequences to which the operations can be applied; A = All sequences, S = Selected, I = Individual sequence

matches. This greatly speeds up the alignment process when multiple, similar sequences are in the alignment.

The next option restricts the target sequence to the top sequence of the alignment only, while specifying "N" causes all non-selected sequences above the query sequence to be searched. This takes much longer, but is very useful when the alignment contains a diverse set of sequences. The next two options specify whether the query sequence, when a match is found, should be moved underneath the target sequence, and whether the sequence description should be updated with the position and name of the matched sequence. It is possible to remove the tag of the search sequence after the match is found and use search fragments generated from the reverse complement of the query sequence if its orientation relative to the alignment is unknown. The next option allows the user to specify whether searching should continue until the best match is found (up to the threshold specified) or to stop when the first acceptable match is found (above the match level specified). The latter is much quicker but may not produce reliable matches in diverse datasets. The next option allows the user to intervene and confirm a sequence alignment move that goes beyond the ends of the alignment (*ie*. cancelling the alignment move, increasing the alignment length or truncating the query sequence can be specified here). Alternatively the alignment can proceed without user intervention, in which case the alignment will be automatically expanded if necessary to accommodate the sequence move.

B) *Identify/Align reading frames*.

If the sequences to be aligned contain coding sequences, they can be initially aligned from the start of the first or the largest reading frame in the specified sequence range. The blue ("ORF finding") menu allows the range where ORFs are to be located to be specified.

The other selectable options are specifying the minimum length of an open reading frame that would be scored (either as an absolute length or proportion of total length of the test sequence, and whether sequences with identified ORFs should be automatically aligned to the start of the ORF of from the first methionine residue. It is possible to align sequences from the first ORF found that matches the minimum length, or to search the whole sequence for the largest ORF and use that for alignment.

It is possible to search the sequence in sense, antisense or both orientation for ORFs. If the ORF is on the antisense strand, the sequence is automatically reverse-complemented prior to alignment with the other sequences. If coding sequences are identified, it is possible to align sequences from the start of the ORF or from the first initiating codon (ATG in translation code 1; coding sequences with incomplete 5' ends are aligned from the start of the sequence irrespective of what option is selected here). The sequence alignment can be automatically renumbered, assigning position 1 to the start of the ORF/first initiating codon of the ORF. Sequences identified as coding can be automatically tagged or untagged using tag groups assigned in subsequent options.

C) *Align around defined motif.*

The menu structure is similar to the previous alignment method, except in this case, sequences are searched for a specific amino acid motif (such as the RNA polymerase "GDD" sequence in many RNA viruses). Motif searching permits matches to ambiguously specified amino acids to be selected as follows:

(x, y…)	Either of two or more alternative amino acids
B	Basic residues (Arg, Lys, His)
U	Acidic residues (Aspartate, Glutamate)
O	Non-polar (Gly, Ala, Val, Leu, Ile)
Z	Bulky (Phe, Trp, Tyr
x	Any

The "Search multiple reading" option allows each of the forward and reverse reading frames to be individually searched instead of the currently set frame. If matches are found, then the reading frame and orientation of the sequence is changed.

D) *Global alignment*.

This automates the process of sequence alignment, and reduces or eliminates the need for manual (and often subjective) alignments of sequence datasets. SSE comes with copies of MUSCLE version 3.8 (Edgar *et al.,* 2004) and CLUSTALW (Cheena *et al.*, 2003) pre-installed. These can be used to seamlessly align sequences within the file.

MUSCLE is both faster and in most reports, more accurate than CLUSTALW for both nucleotide and amino acid sequence alignments. MUSCLE additionally requires much less optimisation and parameter selection at the start of the run. Finally, CLUSTALW is not recommended for large numbers of sequences, being both slow and making alignment errors in the output.

Different methods are used to align nucleotide and coding sequences. In the latter case, alignment is performed on translated amino acid sequences (using the reading frame selected in the sequence editor). This ensures that all insertions preserve the coding sequences, and do not create insertion gaps across codon boundaries. Obviously highly divergent sequences with minimal regions of homology are poorly aligned by both MUSCLE and CLUSTALW, although some optimisation is possible by adjustment of the starting parameters. These are "remembered" between program runs, and can be restored to default values if required.

The Sequence Alignment menu allows the sequence range for the alignment to be specified, whether the sequence is coding or non-coding (thereby selecting for different alignment methods; see above). If the sequences are specified as coding, the alignment will proceed only as far as any stop codon that might be present in the ORF of each sequence (irrespective of the End coordinate).

E) *Add sequence to an alignment*.

This uses MUSCLE or CLUSTALW to add a sequence to an existing alignment (comprising all of the sequences above the first selected sequence). This is obviously much quicker than performing a global realignment each time a dataset is updated. The process uses the same selectable and modifiable run parameters as the global alignment option. <BACK TO EDITING OPTIONS>

COPY/ADD SEQUENCE <Alt>C I

Adds or copies a new sequences to the alignment. Options in the "Copy/Add Sequence" menu allow the number of sequence copies to be inserted to be specified (default 1, up to 30 depending on the size of the screen). It is also possible to specify in advance whether the new sequence(s) should copy the name, description and tag of the copied sequence by default, and whether the nucleotide sequence should be copied. The menu that appears next

allows the sequence name and description of each new sequence to be individually edited before they are inserted into the alignment. <BACK TO EDITING OPTIONS>

DELETE <Alt>D S, I

Allows individual sequences or sequence groups to be deleted. This operation can be drastic and confirmation is requested before proceeding. Deleting all sequences produces an empty file and the program tries to prevent you doing this. <BACK TO EDITING OPTIONS>

GOTO <Alt>G S

Moves cursor to a specified line and position in the alignment. The default is the current cursor position unless the cursor is situated in a blank area of the alignment. In this case the default base position becomes the nearest position containing a base (useful for finding the start of sequences in very wide alignments).

In addition to being specify actual coordinates, “T” and “B” can be entered in place of line numbers for moving to the top or bottom of the alignment. The cursor can also moved a relative amount using "B"(ack), "F"(orward), "U"(p) and "D"(own) prefixes. If entered before a number, they specify the number of base positions or sequences to move, not the absolute destination position. Thus, entering "B10" move the cursor back 10 bases; "U8" move the cursor up the alignment by 8 sequences. <BACK TO EDITING OPTIONS>

SEQUENCE JOIN <Alt>J A, S

Two or more sequences (tagged or all sequences in file) can be joined to produce a consensus sequence. Sites containing unknown bases do not contribute to the consensus, so the method allows two sequences (*eg*. two overlapping sequence fragments) to be joined up. This might help where the same sequence has been read in sense and anti-sense directions, and a composite sequence is required (and which highlights any differences between the two sequence readings). Combined with the Align commands, it also provides the method to assemble a composite sequence from overlapping sequences obtained from longer gene fragments.

Variable nucleotide positions are dealt with according to following options:

1) Variable sites can be represented by an ambiguity code ("A" in field 1), a gap symbol ("-") or an unknown base. For example, combining two sequences where one sequence had a T residue and the other had a C residue would produce a Y (pyrimidine) in the consensus sequence if "A" was selected.

   There is no accepted view on how to resolve a variable site that contains a base in one sequence and an alignment gap ("-") in another. The merge algorithm ignores the alignment gap character if it differs from any other base (except "N"), although this may lead to a poor representation of the consensus sequence.

2) Whether a position is considered to be variable depends on the threshold value entered. This allows one to specify the ranges over which a site is treated as conserved (above the threshold) or variable (below). For example, the default value of 0.75 results in the site being shown as variable when no single base is present in 75% or greater of the sequences. Setting the value at 1 (strict consensus) means that any variability at the site leads to it being shown as variable, while if set to zero (simple consensus), the most common base is always shown (in the event of a tie,

the base shown is determined arbitrarily by sequence order in the file).

When variable sites are displayed with an ambiguity code, each base has to be present individually at a frequency greater than 100% minus the threshold value (*eg.* > 25% for a threshold of 0.75) to contribute to the consensus ambiguity code.

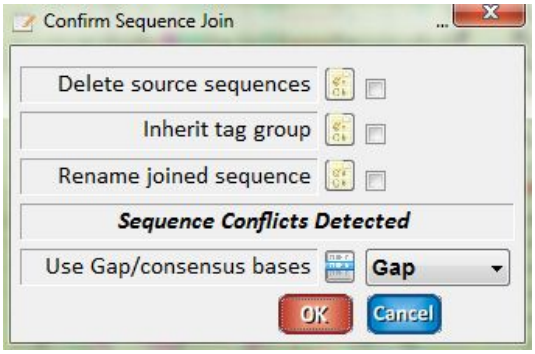
Further input is required is a sequence name and description for the merged sequence (defaults to those of the sequence on which the cursor is placed in the alignment).

Finally, it is possibly to generate a contig report (and associated graphical representation of sequence coverage) during the joining process. This is very useful when creating consensus sequences from pyrosequencing, 454, Solexa and other datasets containing multiple overlapping reads. It additionally computes a measure of sequence heterogeneity, Shannon entropy and sequence quality at each nucleotide position.

<BACK TO EDITING OPTIONS>

SEQUENCE ASSEMBLE

This is a quicker method than sequence joining to create a contig or consensus sequence from two or more selected sequences (or highlighted sequences from the context menu). This provides additional options to delete the source sequences when assembled. It additionally checks for and asks how to resolve any sequence differences between sequences being assembled (a useful check when assembling data).



REVERSE/COMPLEMENT <Alt>L A, S, I

This converts a sequence, a sequence group or the whole alignment to its reverse complement (default). Rather more weird sequence operations possible are reverse, or complement sequences (if you find yourself doing this frequently, there may be something wrong with your project!). This command will correctly resolve ambiguity codes, *eg.* R goes to Y, S and W remain unchanged, and B changes to V. N remains unchanged.

When reverse complementing a sequence that only occupies a part of the alignment, it may be useful to move it to its complementary position or keep it in its current position. The final option allows the reverse-complemented sequence to be labelled by modification of the sequence description. <BACK TO EDITING OPTIONS>

MOVE <Alt>M S, I

Moves a sequence or a group of sequences to a specified line or alignment position. “T” and “B” can be entered in place of line numbers for moving the sequence(s) to the top or bottom of the alignment. The sequence(s) can moved a relative amount using "B"(ack), "F"(orward), "U"(p) and "D"(own) prefixes. If entered before a number, they specify the number of base positions or sequences to move, not the absolute destination position. Thus, entering "B10" move the sequences back 10 bases; "U8" move the sequences up the alignment by 8 sequences. <BACK TO EDITING OPTIONS>

SORT <Alt>S A

Sorts all sequences in alignment in either ascending or descending order:

a) By nucleotide sequence
b) Overall sequence length
c) Alphabetically by sequence name
d) Alphabetically by sequence description
e) Tag group (Green, Red, Blue 1-9)
f) Randomised sequence order
g) Phylogeny (position in tree generated by PHYLIP)

Sorting by nucleotide sequence or length can both be used to arrange incomplete sequences in an alignment. Sorting by nucleotide sequence places sequences in order based on the positions in the alignment of their 5' bases, while sorting by length re-orders the sequences by overall length, irrespective of whether the 5' or 3' ends are missing. Sorting by name or label carries out an alphanumeric sort on the names of the sequences (as sorting is alphabetic, numbers as text so that sequences labelled 1, 2, 3, 10, 11, 21, 22 will be sorted in ascending order as 1, 11, 2, 21, 22, 3).

The option "Randomise" randomly re-orders the sequences.

Finally, sequences can be sorted by phylogeny, so that sequences grouping together in a tree will be ordered together in the alignment. Sorting requires PHYLIP to be installed on the computer. The tree can be based on either nucleotide or amino acid sequences and use either Neighbour-Joining or Parsimony algorithms. This method is very useful for grouping together related sequences so that they can be assigned to groups prior to other analysis. The tree generated by PHYLIP used for sequence ordering (TREESTORE) is retained in the PHYLIP program directory, and can be imported into programs such as TREEVIEW (http://taxonomy.zoology.gla.ac.uk) that allows the branching order to be visualised. Neighbor-Joining is generally much quicker than Parsimony for phylogenetic analysis, and is the default option on the menu. Sorting by phylogeny works best when there is an outgroup; this should be placed on line 1 of the alignment prior to sorting, and line 3 ticked ("Outgroup on first line").

As with the Align command, sorting can be undone by pressing <Esc> to restore the sequence file as it existed immediately before the Sort operation. <BACK TO EDITING OPTIONS>

BLOCK INSERT <Alt><Ins> A, S, I

Allows a specified number of bases (any nucleotide or ambiguity code, or alignment gap) to be inserted into a sequence, a sequence group or the entire alignment from the cursor position. As with the align command, the alignment width may need to be increased to accommodate the insertion of the block of sequence. As before, a dialogue box will appear asking you to do something sensible at this point (*ie*. it's usually a good idea not to truncate sequences that poke beyond the end of the alignment!). <BACK TO EDITING OPTIONS>

BLOCK DELETE <Alt><Del> A, S, I

This allows a specified number of bases from the cursor position to be deleted from a sequence, group of sequences or entire alignment. As this action can be drastic, a dialogue box will appear to confirm that you wish to proceed. <BACK TO EDITING OPTIONS>

BLOCK CLEAR <Alt><Space> A, S, I

This allows a specified number of bases from the cursor position to be cleared from the sequence, group of sequences or entire alignment. As this action can be drastic, a dialogue box will appear to confirm that you wish to proceed. <BACK TO EDITING OPTIONS>

SEQUENCE RENAME <Alt>N S, I

This allows the names and descriptions of individual or a blocks of tagged sequences to be edited. Names and descriptions of single or up to a maximum of 15 sequences (depending on screen size) can be edited together on the same multi-line menu. <BACK TO EDITING OPTIONS>

EDIT NAME/SOURCE I 

This allows the name, tag group assignment, description and source (sequence annotation) of an individual sequence to be edited. Clicking the current sequence display below the menu bar, the menu button or right-clicking the mouse on an individual displayed label or sequence in the editor screen also accesses this command.

The window that opens displays the sequence name tag group assignment, description and annotation of the current sequence. The label window runs independently of the main editing window and can be left to run alongside it. Changes in the current line are updated in the label window and vice versa using the up and down arrows in the tool bar of the label window to move between sequences. 

SEQUENCE SEARCH <Alt>Q or F2 A, S

This command allows the sequence on the cursor line to be searched from the cursor position to the end of the sequence (or from the cursor position to the beginning of the sequence if in reverse view).

The sequence to search for is entered through a dialogue box. To search for nucleotide sequences, G, A, T(U), C or any of the ambiguity codes can be entered (or the alignment gap symbol "-"). Ambiguities in either the search code or in the sequence in the alignment will be correctly resolved (*ie*. searching for an "R" (purine) will identify the following bases in the alignment as possible matches:

G, A, R (G or A), B (C, G or T), D (A, G or T), H (A, C, T), V (A, C, G) and N

The search string has the additional “?” code which indicates any base; this is in distinction to the code “N” which explicitly searches for an “N” in the sequence. The ability to use a search sequence with ambiguities (including "?") means that polymorphic motifs can be searched for in a sequence.

If sequences in the alignment are viewed in split or translate mode, the dialogue box also contains the option to search for amino acid sequences, using single letter amino acid codes. Stop codons, unknown amino acid residues or alignment gaps can be entered using the symbols "*", "X" and "-" respectively.

If a sequence match is found, the cursor position is moved to the position where the match occurs. Failure to find a match with the entered code either produces a dialogue box asking whether the search should be continued to the next sequence or simply continues (this option is selected on the Search box through the rest of the alignment. <BACK TO EDITING OPTIONS>

SEQUENCE SEARCH/REPLACE A

This allows nucleotide sequences (of up to 100 consecutive bases, using the same codes as Sequence Search) to be searched within an alignment, and if specified, replaced by another sequence.

Search and Replace can be constrained by codon boundaries (“Limit to codon” option on line 5) such that “CG?” will only find this motif at positions 1, 2 and 3 in the sequence (*ie.* in this case, arginine-encoding codons). The length of search strings will be normalised to multiples of three with unspecified bases replaced by “?” (*eg.* “GG?G is effectively “GG?G??”).

Replacement codes are standard with the following additional options:

Upper case (*eg.* “Y”, “N”)	Replace with the explicit ambiguity code specified
Lower case (*eg.* “y”, “n”):	Replace with a random base within the range specified by the ambiguity code (*eg.* “y”: T or C; “n”: A, C, G or T. The likelihood of using each base is proportional to their frequency of occurrence in the selected sequence dataset (calculated separately for all 3 codon positions if the search / replace operation is constrained by codon boundaries.

"?": Leave position alone (*ie.* don't change base)

Finding and replacement actions are specified separately as in the option to replace all occurrences of the sequence. <BACK TO EDITING OPTIONS>

SEARCH LABEL <Cntrl>Q A, S

This allows text (up to 255 characters) to be searched for in the sequence name, sequence description, sequence source or all three. Text matches can be case-sensitive or –insensitive (3rd option on menu). Should a match be found, the cursor is moved to that sequence, or the sequence is tagged and the search continued.

This command is useful when trying to locate individual sequences in large alignments, or to identify and label groups of sequences sharing part of their description, such as a classification level (*eg.* Vertebrata) found in the sequence description line or annotation (if imported originally from GenBank or EMBL). <BACK TO EDITING OPTIONS>

LABEL SEARCH/REPLACE A, S

This allows text (up to 255 characters) to be searched for in the sequence name or sequence description. Text matches can be case-sensitive or –insensitive. <BACK TO EDITING OPTIONS>

TAG SET SELECTION A

This allows one of the four tag sets to be selected (and annotated as required). See "Sequence Tagging"

SELECT TAG GROUPS A

This allows one or more tag groups to be selected, and (optionally) an existing selection of sequences to be de-selected (using the tick box). See "Sequence Tagging"

TAG ASSIGN/CLEAR A

This provides options to manually set or clear tag group assignments collectively from the sequences within an alignment. See "Sequence Tagging"

TAG DESCRIPTION n.a.

A tag group is selected and then a text box appears allowing entry of the (short) tag label followed by a description of the tag group as required.

COPY/CUT TO CLIPBOARD <Cntrl>C. <Cntrl>T S, I

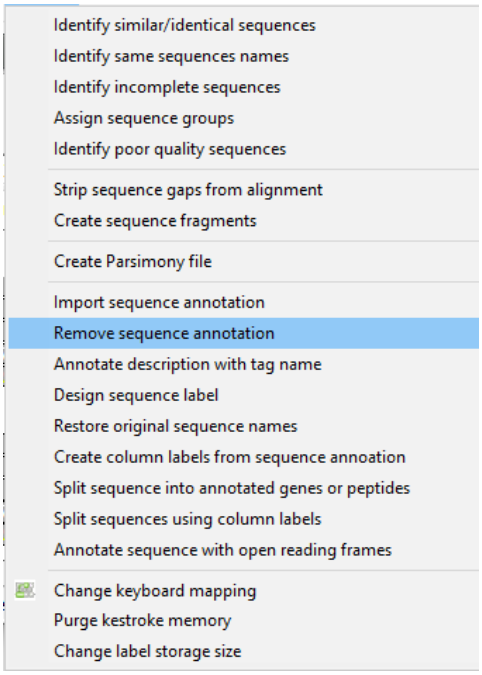
This copies or cuts a marked sequence or block onto the clipboard for rapidly pasting into other applications or other copies of SSE. The sequence is trimmed of terminal unknown bases and alignment gaps. <BACK TO EDITING OPTIONS>

SEQUENCE PASTE	<Cntrl>V	n/a

Pastes the contents of the clipboard into the current sequence line. Pasted sequences are automatically recognised as nucleotides or amino acids; non-conforming characters are removed from the sequence(s). <BACK TO EDITING OPTIONS>

UTILITIES

A series of utilities are provided for editing sequence alignments



A) Identify similar or identical sequences. This scans a block of sequences or the whole alignment for identical (or similar) nucleotide or amino acid sequences. Sequences in the specified sequence range that are identified as identical (or similar) to other sequences are tagged, and optionally, the sequence description modified to indicate the sequence to which it resembles. The similarity threshold allows the user to set the level at which to sequences would be considered to be matched. Unknown bases within a sequence can be considered as matches or ignored depending on the response to option 5.

B) Identify same sequence names. This searches for and tags sequences with the same name and/or the same description (options 1 and 2 on the "Select names and Tag" menu). Sequences that are matched are tagged and the sequence description updated as specified.

C) Identify incomplete sequences. This scans a specified block of sequences or the whole alignment for sequences that are shorter than a specified length (defaulting to half the total alignment length) after unknown bases and alignments gaps are removed. Sequences in the specified sequence range that are identified as shorter than the specified value are tagged using an unused tag number. The tagged grouped can then be collectively moved or deleted prior to any further analysis or procedure that requires sequences of a certain minimum defined length (such as sequence order scrambling).

D) Assign sequence groups. This compares sequences with each other in an alignment and measures whether they are more or less divergent than a specified nucleotide or amino acid divergence value (specified in line 11 – "Group threshold"). Sequences more divergent from those previously analysed are assigned to a new tag group or Group name recorded oin the sequence description (specified in line 12 – "Tag or label sequence groups"). Tags are better but there are only 27 of them. The program will produce an output file that lists the sequences compared and their tag or group assignments.

This program is extremely useful for initial categorisation of sequences into groups, particularly in large datasets where manual assignments based on distance tables or phylogenetic trees can be extremely tedious.

E) Identify poor quality sequences. This scans sequences for incompleteness (missing bases), unresolved or ambiguous bases or stop codons within coding regions. The identification of these markers enables poor quality sequences to be identified and excluded from further analysis. The menu allows the user to specify minimum standard for sequence quality; the program will generate a report the provides totals of each marker for each sequence scanned.

F) Strip sequence gaps from alignment. This removes alignment gaps in the specified range from individual or blocks of sequences. If the sequences are indicated as coding, then removal of gaps is limited to those spanning whole codons, and therefore preserves the reading frame of the sequence. It is also possible to strip out unknown bases (option 4). After gap removal, the leading gap can be removed if required, to collapse the alignment. Finally, if more than one sequence is selected, it is possible to remove only those gaps or unknown bases that are present in all selected sequences (option 6).

G) Create sequence fragments. This allows blocks of sequences of defined width and overlap to be generated from a sequence alignment. Sequence fragments are created over a specified range (options 3, 4). For coding sequences, it is possible to ensure that the boundaries of the sequence fragments coincide with codon boundaries (Option 5), and can be limited to the open reading of the start base in the reading frame used by the sequence editor (Option 6). Sequences can be trimmed to remove leading and trailing unknown bases or alignment gaps (Option 7).

The final fragment may be shorter than the others, and there is the option to increase the overlap between the last and second last fragments to ensure that all are of uniform length (Option 8). It is possible to retain the sequences generating the fragments in the resulting data file, or they can be eliminated (Option 9). There is the option to label the sequence fragments by their coordinates and fragment number; this is added to the sequence description line (Option 10). Finally, it is possible to omit the original sequence description from the sequence fragment label; this is often long, repetitive and redundant for most analyses.

H) Create parsimony file. This program takes a sequence alignment (in a defined range) and uses the program DNAPARS in the PHYLIP package to construct a phylogeny and the most likely (parsimonious) sequence at each node in the tree. The additional node sequences are re-imported into the alignment. The sequence descriptions of the original sequences and of the newly created nodes are replaced by a code that links each according to the reconstructed phylogeny. This code is required for analysis programs that can use phylogenetic information (*eg.* Covariance Test, Sequence Changes).

This program requires that the program DNAPARS in the PHYLIP package is installed. The program will be located automatically if it is installed on the same drive as SSE. If not found, then its location can be specified in the "Default File Location" option in the file menu.

The following options use or generate standard sequence annotations in files imported from GenBank / ENA

A) Import sequence annotation. This provides the user with ability to automatically annotate sequences in an alignment from an external source. The procedure works by matching elements in the sequence name with data imported from GenBank or user spreadsheets that match sequence entries.

i. *Import from Clipboard*. This allows data in the form of a table to be read and potential matches between sequence names and table entries to be made. Data in a table should contain tab separated columns. Each column is sequentially scanned for a match with the name of each sequence selected for data import. On finding a match, the remaining fields in the same line of the table are imported into the annotation of the matching sequence. The match position can be in any column of the table and sequences and data table lines can be in any order. The operation never alters the actual sequences in the

file. If sequences in the file are already annotated, these can either be combined with the imported metadata or replaced.

ii. *Import from GenBank.* Sequences names are scanned for accession numbers and these are then used to retrieve sequence metadata for them from GenBank. These are then incorporated into the sequence description and annotation of the sequence. The operation never alters the actual sequences in the file. If sequences in the file are already annotated, these can either be combined with the imported metadata or replaced.

B) Remove sequence annotation. The existing description and associated annotation are removed from specified sequences.

C) Annotate description with tag name. This simply prefixes sequence descriptions with the names of the tag groups they are assigned. This can be useful for sequence labelling and identification in output files, sequence sorting and searching.

D) Design sequence label. This allows various elements in the annotation to be used to create or modify an existing sequence label so that they are more informative as labels in trees or other outputs. The elements that can be selected for annotation include a range of elements codified in a standard GenBank / ENA annotation (*eg.* accession number, sample date, country, organisms, Tax ID, family, genus or other derived values (country code, sample year, tag name). Or include the current name of the sequence as an element in the label. Multiple entries of accession numbers can be optionally excluded (penultimate line). The final box option allows the delimiter between these elements to be selected – *eg.* a dash, underline, slash etc. Note that not all items of metadata listed in the option box will be present in the annotation.

E) Restore Original Sequence Names. If sequence names have been replaced using the <DESIGN SEQUENCE LABEL> function, this function allows the original name to be restored.

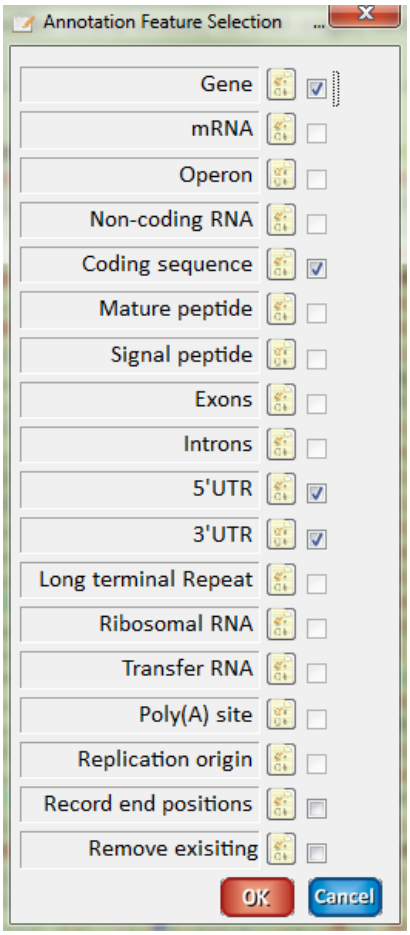
F) Create Column Labels from Annotation. A number of standard features in a sequence annotation can be used as to form column labels in the alignment.

To add column labels, move the cursor to a suitable reference sequence whose annotation is to be used. Normally this would be the sequence used for alignment numbering (see <SEQUENCE NUMBERING>).

Running the command open a menu from which available annotation features can be selected (unavailable annotations for that particular sequence are grayed out and cannot be selected).

The final two choices are:

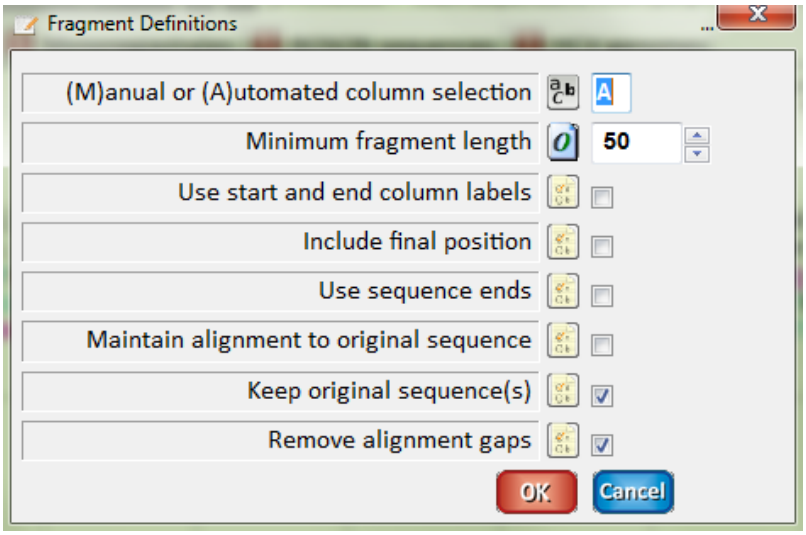
- Whether to insert a second column at the end of the annotated feature, for example, mark both the start and the end of a coding sequence (CDS) or only mark the start.
- Removing or keep existing column labels.



G) Split sequence into annotated genes or peptide sequences. This function is equivalent to the option available on <SEQUENCE OPEN> to convert a nucleotide sequence into one or more sequences based on the coordinates provided in the annotation for gene sequences or peptide cleavage sites. The original sequence is kept in the file.

The newly created sequences are aligned to the start of the file with the first codon in reading frame 1 – this may differ from the reading frame used for the existing alignment.

H) Split sequences using column labels. This option allows one of more sequences in an alignment to be split into fragments defined by column labels. The column labels to use for this purpose can be manually selected or the process automated. In either case, there are few options to consider when performing the operation offered in the main menu. These include specifying a minimum sequence length and importantly whether the fragment definition is based on separate start and end columns (as might be available for annotated gene sequence) or whether the fragment ends just before the start of a new fragment (as you might do when dividing sequences based on protease cleavage sites in a polyprotein).



I) Annotate sequences with open reading frames. This function identifies open reading frames within a sequence and adds a description of these into the sequence annotation. The format of these follows that of GenBank gene annotations. Having performed this step, the sequence can then be split using the <SPLIT SEQUENCE INTO ANNOTATED GENES OR PEPTIDE SEQUENCES> described above if required. Alternatively, the annotation can be used to create column labels and several or all sequences in the alignment split using the <SPLIT SEQUENCES USING COLUMN LABELS> function.

The options available for selecting open reading frames for annotating sequences are similar to those in the <IDENTIFY READING FRAME> function in the <SEQUENCE ALIGN> menu. These include specifying a minimum ORF size and/or proportion of total sequence length and sequence orientation

Other utilities

A) Change Keyboard Mapping. (<Alt>K). This is a bit passé with the passing of manual sequence entry, and was provided to assist entering data from sequencing autoradiographs or Chromas files through the keyboard. The numeric keys along the top of the keyboard (and the keypad to the right) can be mapped to specific bases. By default, the following assignments are made:

Key	Base	Key	Base
1	G	6	G
2	A	7	A
3	T	8	T
4	C	9	C
5	N	0	N

Whether this is convenient or not depends on the order in which tracks are loaded on a gel. The altered key assignments are permanently stored, and are used as the default for all subsequent files. Note it is possible to assign keys to any of the ambiguity codons (such as "R") if these are frequently encountered in the sequence (although let's hope they aren't!).

B) Purge keystroke memory. User entry in the various input and selection boxes is retained during and between editing sessions to speed up the selection on commonly selected options. This can occasionally become a hindrance and this provides the option to get rid of all those entries and start again from the program-supplied default values.

C) Change label storage size. This allocates the storage space for the sequence description and annotation for each sequence within a file. This is adjusted automatically when sequences are loaded or imported into an existing file, but it does increase file sizes and may not be optimal for alignments containing a very large number of sequences (and where annotation may not be needed on individual sequences). This option allows the storage size to be re-allocated. Note however, that reducing its size will truncate annotation data for sequences within the alignment. <BACK TO UTILITY SUMMARY>

ENVIRONMENT

The following menu options allow the display of sequences to be modified, and various settings such as sound, insert mode, sequence numbering and labelling to be modified. Each of these settings and the cursor position are stored when the file is saved in SSE format.

SHOW NAVIGATOR WINDOW

This opens the sequence navigator window if not already displayed. See NAVIGATOR WINDOW

CHANGE SOUND SETTINGS

Not one but three possible sets of merry error messages to speed you along your way when using SSE (Default, Father Ted and 2001 – A Space Odyssey). These sounds have wasted half a megabyte of your hard disk! <BACK TO ENVIRONMENT OPTIONS>

REVERSE VIEW <Cntrl>R

This changes the representation of the sequences on the screen (but without changing their underlying representation in the file). This allows nucleotide sequences to be read in either orientation (without changing the keyboard assignments). Generally, editing in reverse view is similar to sense view, although there are minor differences in how sequences are altered by the delete keys. Reverse view can be switched between the two states by pressing the function button, accessing the Environment menu or the keyboard shortcut. <BACK TO ENVIRONMENT OPTIONS>

CHANGE NUMBERING <Alt>Y

Nucleotides (split and translated) and amino acids can be separately numbered based on their position in an alignment or they can be numbering relative to individual reference sequences. The following three numbering options are selectable from the “Renumber Alignment” box:

1) *Numbering Reference Sequence*. This allows sequences to be numbered relative to a reference sequence selectable by number in the first input line (selecting zero reverts to alignment numbering used in previous versions of SSE). Separate reference sequences and start positions can be selected for nucleotide and amino acid numbering. Numbering by reference sequence accommodates variation in the length of other sequences through nucleotide/codon insertions through the introduction of alignment gaps. This method of numbering requires a method for identifying bases or amino acids not present in the selected reference sequence. These are labelled by the last nucleotide position before the insertion appended sequentially with lower case letters, roman numbers or decimals (selectable by menu option 3; see below). As an example, the three inserted “T” residues in the variant sequence (not present in the reference sequence):

```
                                  1 1       1
                1         7 9     0 2       8
Reference :     CCC GGG AGT ---   AAC TGT GGA
Variant   :     CCG GGG AGT AAA   AAC TGT GGC
                          abc
```

would be labelled sequentially as 9a, 9b, 9c (or 9i, 9ii, 9iii, 9.1, 9.2, 9.3).

The label for the reference sequence is shown in pink in each display mode. Alignment numbering is recalculated "on the fly" if the reference sequence is edited, and its position in the alignment is recorded enabling sequence sorting and other sequence moves to be accommodated without altering nucleotide numbering. When selected, reference sequence numbering is used for all program data outputs (such as the analysis methods in the research section).

2) *Required current position.* Numbering may start at "1" for the first base in the alignment or reference sequence, although an offset can be specified to enable correct numbering of partial sequences. The nucleotide position of the base on the cursor position can be entered in menu item 2 within the range 999,999 to -999,999. This option is available for both alignment and reference sequence numbering systems (see option 1)

3) *Gap labelling method.* This option applies only to reference sequence alignment numbering (selected in option 1). Three different methods for labelling inserted bases or amino acids relative to the reference sequence can be used. The alphabetic system follows the series a, b, c, …z, aa, ab, az, ba, ….. zz, aaa, aab, …. zzz (maximum insertion length 17576 bases before reverting to a, b, c.. in a new cycle). Roman numbers in lower case (I, ii, iii, ….ix, x, xi, ..) can accommodate insertion lengths of up to 1000 bases before recycling. Finally, decimal labelling follows the series .1, .2, .3 … .9 for gaps lengths of less than 10, 0.01, 0.02, 0.03…. 0.99 for gap lengths of 10 or more and less than 100, and so on up to a maximum length of 10,000 bases. These three different systems have advantages and disadvantages. The alphabetic system is most commonly used, but the decimal system is often preferable as an output format for analysis of data where nucleotide or amino acid positions are assumed to be numeric. . <BACK TO ENVIRONMENT OPTIONS>

SOUND

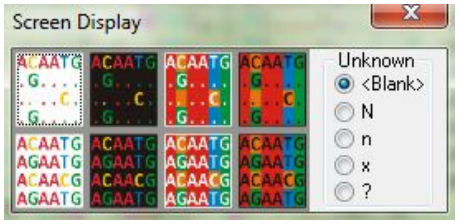
Sound is a three-state option (Disabled, Off, On), that is used for manual sequence entry (rather passé as described above), and to indicate errors and the progress of certain file and sequence operations.

During sequence editing, bases entered that are different from the base in the sequence on the first line of the sequence file will be indicated vocally if sound is ON. Sound can be switched between the three states using the Environment menu or the supplied function button. <BACK TO ENVIRONMENT OPTIONS>

SWITCH DISPLAY MODE <Alt>U

Generally the most useful display for sequences in an alignment is to indicate only differences from the reference sequence on the top line. Bases that are the same are displayed as "."'s. The display mode can be changed as follows:

1. Sequence identities with Sequence 1 shown as ".". Sequence 1 always displayed.
2. All bases. Sequence 1 always displayed
3. All bases. Sequence 1 scrolled off screen
4. As 1. but in reverse video
5. As 2. but in reverse video
6. As 3. but in reverse video



Each of the six view modes apply to the nucleotide, split and amino acid sequence display. View mode can be changed by typing <Alt>U or selecting the option from Environment Menu box. The menu additionally provides the option to change the way unknwn bases (N" are presented in the screen display (blank characters by default) <BACK TO ENVIRONMENT OPTIONS>

EDIT BASE/AMINO ACID COLOURS

Different nucleotides and amino acids are coloured to make sequence variability in the alignments easier to visualise. For the default display, amino acid residues are coloured according to their properties:

Pos. / Polar	Red / Pink
Neg. / Polar	Blue / Light Blue
Uncharged	Grey
Cysteine	Yellow
Large /Other	Black or White

The colours of nucleotides and amino acids is fully editable from buttons on the toolbar. <BACK TO ENVIRONMENT OPTIONS>

TOGGLE SEQUENCE DISPLAY MODE <Alt>T

This command toggles between five different views of the sequence dataset:

1) The sequence names and a window displaying their nucleotide sequences. This display mode allows editing and alignment of sequences, and is the default on first running the editor.

2) A split display in which the nucleotide and translated amino acid sequences are shown on upper and lower parts of the screen. This display mode also allows both editing and alignment of sequences.

3) A display of the translated nucleotide sequences of each sequence. This display mode allows only alignment changes between sequences, not alteration of the sequences.

4) A display of amino acid sequences of each sequence, using amino numbering. This display mode allows full editing functions to be carried out (including manual entry and deletion of amino acid residues).

5) A list of sequence filenames (with their associated tag).

In general, translation is essential for sensible alignment of coding sequences where gaps

introduced should not cut across codon boundaries. One letter amino acid codes are used, and translation proceeds simultaneously with sequence entry. Translation of codons is carried out in real time during sequence editing, and is completed wherever possible when ambiguous nucleotide codes are entered (*eg*. GGN [but not GG-] would be translated as Gly, while ATH would be translated as Ile). <BACK TO ENVIRONMENT OPTIONS>

CHANGE READING FRAME <Alt>"+", <Alt>"-"

The reading frame of the translated nucleotide sequences can be changed. Consecutive triplets starting from the first position in the alignment are translated in Forward Frame 1, from position 2 as Forward Frame 2, and from position 3 for Forward Frame 3. It is also possible to display Reverse reading frames (1-3), in which the nucleotide sequences are translated from the nucleotide sequence in reverse complement orientation (*ie*. GGA at positions 1-3 in the alignment would be translated as G (Gly) in forward Frame 1, but translated as S (Ser) in Reverse Frame 3 (*ie*. as TCC). All 6 possible reading frames can be cycled through using the following keyboard shortcuts:

<Alt>"+"	Increase reading frame
<Alt>"-"	Decrease reading frame

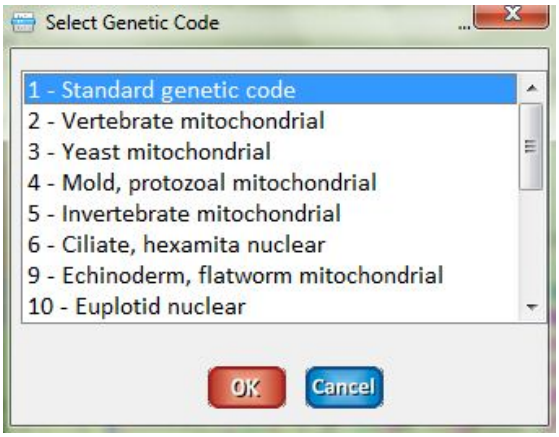
The reading frame can also be changed using the radio buttons on the toolbar. <BACK TO ENVIRONMENT OPTIONS>

DISPLAY DNA/RNA

The symbol T or U are used in the representation DNA or RNA sequences, and can be specified during sequence editing. This selection will apply not only to the screen display, but also sequences printed to file or exported. The DNA/RNA display selection will be saved as a sequence file attribute. It can be selected from the Environment menu box, or by entering bases with the "T" or "U" keys on the keyboard (this does not affect keyboard mapping selections, and T/U are interchangeable when searching sequences). <BACK TO ENVIRONMENT OPTIONS>

SELECT GENETIC CODE

The genetic code used to translate nucleotide sequences for display, printing and most analysis program can be changed between the standard code to modified codes used by, for example, some bacteria and by mitochondria. The genetic codes selectable in SSE follows the standard assignments for codon tables:



1: Standard
2: Vertebrate mitochondrial
3: Yeast mitochondrial
4, 5: Invertebrate mitochondrial
6, 9, 10, 11: Bacterial and plant plastid code
12, 13, 14, 15, 16, 21, 22, 23: Other mitochondrial.
(see http://www.ncbi.nlm.nih.gov/Taxonomy/Utils/wprintgc.cgi)
The genetic code assigned to an alignment (1 - 23) is indicated in parentheses after the reading frame on line 3 of the screen display.

Apart from displaying and printing amino acid sequences, the code used for translation potentially influences many aspects of analysis built into SSE. The following analyses use amino acid sequences translated according to the selected genetic code:

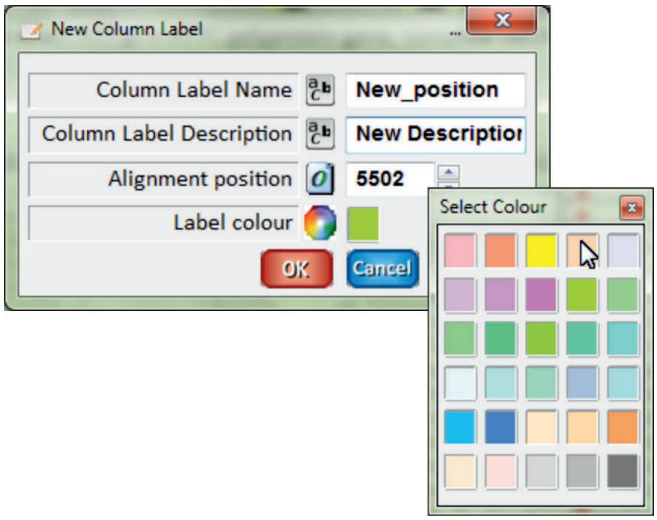
a) Amino acid sequence distances in <SEQUENCE DISTANCES>
b) Trees based on non-synonymous changes in <TREEORDER SCAN>.
c) Corrected ratios of dinucleotide frequencies, frequencies of codon use and relative synonymous codon usage, amino acid frequencies, and amino acid variance and bias measures in <SEQUENCE COMPOSITION SCAN>.

Due to limitations in the underlying algorithms, the following analyses remain based on Codon Table 1 irrespective of what is selected:

a) Synonymous and non-synonymous distances in <SEQUENCE DISTANCES>
b) Effective codon number in <SEQUENCE COMPOSITION SCAN>.

COLUMN LABEL FUNCTIONS

This provides a series of options for adding a new column label or modifying, copying or deleting an existing one. Each calls up a standard menu that asks for the column name, column description, its position and display colour. Column lables must have unique names as they qre used as identifiers in the preset menu of other programs that require sequence positions to be entered. Column labels are stored with the file when saved in SSE format.



The same functions can be selected, where appropriate, from context menus. See <COLUMN LABELS> for further information

LINEAR/CIRCULAR

Alignment can be marked as linear or circular, depending on the nature of sequences. This influences the editing of sequences; such as alignment of sequences and dragging individual or blocks of sequences beyond the left and right boundaries of the alignment <SHIFT><Left Arrow>, <SHIFT><Right Arrow>. The alignment attribute is preserved when sequences are stored in SSE format. <BACK TO ENVIRONMENT OPTIONS>

SWITCH FILES

This menu item (or the keyboard shortcuts <Cntrl><Tab> or <Cntrl>W can be used to run through currently open files (as an alternative to clicking on tab headers). <BACK TO ENVIRONMENT OPTIONS>

<BACK TO DOCUMENT INDEX>

RESEARCH SECTION

The following commands form a set of sequence analysis methods for research analysis. The following sections describe the operations in brief, but the reader is recommended to find fuller descriptions in the references cited. All of these programs are experimental, and are obviously not bomb-tested in the same way as the rest of the package. It is also quite possible to produce highly erratic and extremely un-useful results if run with inappropriate datasets!

A) Special requirements for analysis programs

There are some additional requirements for running many of the programs in the Research section. Firstly, some programs need PHYLIP or RNAFold to be installed on the computer. Information on how these can be installed and linked to SSE is described in the <SETUP SECTION>.

The special requirements are listed below:

1) The following commands require that PHYLIP is installed on the hard disk to enable the program to run:

Covariance Scan (if using parsimony scoring)
Scanning a sequence for covariant sites, and associated regions of internal base-pairing associated with RNA secondary structure.

Association Index
Scoring phylogenetic segregation of sequence groups.

Grouping Scan
Scoring group membership of individual sequences in phylogenetic trees.

BootScan
Bootstrap support for phylogenetic grouping of a query sequence with control groups.

TreeOrder Scan
Comparing phylogenies of trees generated from different regions of sequence alignments.

2) The following commands require that RNAFold is installed on the hard disk to enable the program to run:

Folding Energy Scan
Detection of sequence order-dependent RNA structure

StructureDist
Identification of conserved (predicted minimum energy) base pairings between sequences within an alignment

3) Most programs generate output files and graphical output. Output data is created in Text (ASCII) format, and can be imported into more or less any spreadsheet or statistics package. Data fields are delimited by Tabs.

The nine programs generating data allow the directory and file name of the data to be specified at run time, and the defaults to be changed and saved. This allows the creation of analysis folders separate from those that contain sequence or other project files. Although the same default directory and filenames are shared between the programs, files names also contain a descriptive element, ensuring that the output file name from each program is distinct.

4) Installation of SSE includes a graph plotting package that is interfaced with most research programs to generate graphical output. The appearance of output graphs (x- and y-scales,

etc.) can be edited and subsequently copied and pasted into a drawing package (.EMF format recommended) for final editing and storage. An option to save files in DPlot format is provided by the DPlot program, but loading these requires purchase of the full DPlot package.

<BACK TO RESEARCH OPTIONS>.

B) Listing of analysis programs. The following lists the available analysis programs selectable from the Research menu. An expanded list with brief description of the programs and brief (editable) instructions for running each program is provided by clicking the Research menu button on the toolbar or by the <Alt>R shortcut.

Research programs have been divided conceptually into functional groups, comprising:

- Distance and composition measurements (Sequence Distances, Sequence Motif scan, Similarity Scan, Composition scan, Sequence Changes
- Phylogeny grouping methods (Association Index, Grouping Scan, Boot Scan and TreeOrder scan)
- RNA structure prediction methods (Folding Energy scan, StructureDist, Covariance scan).
- Utilities (Mutate sequences, Scramble sequences)

Program	**Scope**
SEQUENCE DISTANCES	A, S

This program calculates divergence scans, matrices and listings of pairwise distances between nucleotide and translated amino acid sequences using a variety of evolutionary models. Analysis of pre-defined sequence groups (selected as described above: <SEQUENCE SELECTION>) allows mean pairwise value to be separately calculated (*eg.* within and between assigned tag groups).

Standard options select the name of the output files, sequence range for distance measurement, the format of the output files (whether distance matrices or lists). The numeric precision of the distance measurement can be specified, as well as the method to treat alignment gaps or missing bases (global deletion from the dataset or ignoring positions only in pair-wise comparisons where such bases or gaps occur). Standard errors can be calculated for each method for measuring distances, and these will be outputted as separate matrix files (codes as listed in footnote), or as separate columns in the list format (in the file "<UN> Distance.TAB). It is also possible to specify the parameters for a sequence scan. The final option allows the selected filename to be saved as a default name for analysis files.

Running the program computes pair-wise distance and standard errors between each selected sequence. It is also possible to compute mean pair-wise distances and errors between selected groups that are tagged with two or more different tag labels. If the sequence scan option is selected, the next menu allows the parameters to be entered (fragment size, increment between fragments). Running a sequence scan adds columns indicating the fragment number, location (alignment position of the mid-point of the fragment), and alignment coordinates to each line of the output file (recording pair-wise distances between tagged groups).

Sequence outputs include:

a) Tabulated lists of pairwise distances with columns identifying the sequences or groups being compared followed by columns listing the pairwise distances (one column for each distance measurement selected)

b) Half-diagonal (lower left) distance matrix formatted in either user-friendly, PHYLIP or MEGA format[6]. Separate matrix files generated are generated for each sequence measurement selected. Matrices can contain pairwise distances between individual sequences or between different tag groups. For the latter, within-group pairwise distances are included on the half-diagonal position in the user-friendly output format.

c) If graph output is selected, histograms of pairwise distances between groups are generated. Graph output required that distances are generated in table format rather than as matrices.

d) For sequence scans, graph output for group comparisons is plotted as genome position (x-axis) and distance (y-axis), one graph for each distance measurement selected.

e) For sequence scans, file output is restricted to table format where additional columns identify fragment coordinates and length.

The algorithms used are based upon distance and standard error formulae presented in (Li & Graur, 1991). The calculation of matrix distances between translated amino acid sequences used a standard similarity matrix. These distances are read from a matrix file generated automatically when SSE is first run. These corrected distances can be individually edited as required. <BACK TO RESEARCH OPTIONS>.

Program	Scope
SEQUENCE MOTIF SCAN	A, S, I

The program scans sequences to calculate within-site variability across a sequence alignment by computing a Shannon entropy score. The same program can also compute mono- or polynucleotide frequencies, with the option to select up to 16 different bases or combinations of bases. The number of search elements is selected and then specified using the subsequent "Nucleotide Elements in Scan" and "Select Bases for Scanning" input boxes. As well as specifying individual bases, it is also possible to measure frequencies of combinations of between 2 to 5 nucleotide or ambiguity codes (such as "ACRNU").

In addition to the graph output, two files are produced, one lists the positions of motifs in the alignment and a windowed average. A second file provides data on the frequencies of spacing between each motif selected for analysis in range from 1-59 and ≥60

The progress and variability calculated during the scan is displayed. The process can be interrupted by pressing any key. <BACK TO RESEARCH OPTIONS>.

[6]Matrix output file names are coded to provide information on the contents, sequence type and comparison method in three letters:

1) Distance/Standard Error)
2) All/Synonymous/Non-synonymous/Protein)
3) P distance/JC distance/2-parameter/Tajima & Nei / Li,Pamilo & Biancho / Kimura / Matrix

Program	Scope
SIMILARITY SCAN	A, S, I

This program scans pairs of sequences for regions of nucleotide or amino acid sequence similarity; regions meeting or exceeding the set criteria of number of matches over a defined window size are plotted on a x/y dot plot graph with colour depicting the degree of match and/or printed out to a file. This lists the sequence matches as X and Y coordinates, along with the matching score, spacing and the sequences of the matching window.

Comparisons are made between or within each sequence in the selected block. The program can compare both nucleotide and amino acid sequences (selected in the "Similarity Scan" menu). For both, the sequence range over which the comparison is made, and the name and location of the output file is then specified in the "Nucleotide Similarity" or "Amino Acid Similarity" menus.

A) Nucleotide similarity scan. For nucleotide sequence comparisons, the number of matches and the length of the comparison window are specified and the sequence range over which the comparison is made. For approximately co-linear sequences, homologous sites may differ in position in the two sequences by much less than the total length of the alignment. For aligned sequences, a value of zero can be specified here. Since similarity scanning is highly computer-intensive for long sequences and for multiple sequence comparisons, specifying a minimum look-ahead/-behind value is useful for speeding up analysis.

It is possible to specify a fractional score for sites differing by a transition (*ie.* G and A, C and U/T) rather than a transversion (identical bases score as 1. The "Scan single sequences" option allows sequence comparisons to be carried out between sequences or within each sequence (where it may detect internal repeats or regions of complementarity). For the former, at least two sequences have to be selected for comparison.

B) Amino acid similarity scan. This scan pairs of sequences or internally within a sequence for similarities in the encoded amino acids, using the same options to select number of matches, comparison length and look-ahead/-behind range as for the nucleotide similarity scan. Options specific to this scanning method include the use of the standard PAM-Dayhoff matrix to allow similarities within different classes of amino acids to be incorporated in the matching calculation . These distances are read from a matrix file generated automatically when SSE is first run. These corrected distances can be individually edited as required.

It is also possible to specify the reading frames of the two sequences being compared (the orange "Select Reading Frame" menu). The default is "All reading frames", which means that for each window, comparisons are made between the X sequence translated in the 3 forward and 3 reverse reading frames with each of the 6 translations of the sequence in the Y window. This provides a useful method to detect homologous amino acid sequences where the orientation and location of open reading frames in the test sequences is not determined. It is possible to limit which reading frames should be compared (often all three forward reading frames will suffice if the orientations of the test sequences are known). In this case, the reading frame used in the sequence editor corresponds to Frame 1 in the check-box list.

The output file, as well as including the coordinates, score and amino acid sequences of the matches, also records the reading frame of the X and Y sequences where the matches were found. For homologous sequences, sequential matches should generally be in the same X and the same Y reading frames. To help in the analysis, a composite reading frame number (1-36) is provided in the

last column; this uniquely identifies each combination of X and Y sequence reading frames, and should be constant across regions of similarity between sequences <BACK TO RESEARCH OPTIONS>.

Program	Scope
COMPOSITION SCAN	A, S, I

This program takes one of more sequences and analyses their nucleotide composition, their dinucleotide frequencies, ratios to expected values, and for coding sequences, codon usage tables. After selection of all or individual sequences or sequence groups, the standard menu options selectable include output filename, region of sequence to be analysed,

The "Composition Variables" option box allows the name of the output file, a name for the analysis (used in Table format; see below) and the sequence range to be analysed. The "Analyse by codon" option (option 6) applies only to coding sequences, and allows separate composition measurements to be listed in the output file for 1st, 2nd and 3rd codon positions. The analysis can be automatically restricted to the open reading frame, and will not include any sequence downstream of a stop codon, irrespective of the End Position. The reading frame used to translate sequences for codon usage analysis is the same as used for the Editor screen.

Similarly, it is possible to restrict analysis to the informative sequence range only, and to not include leading or trailing unknown bases or alignment gaps if they impinge on the specified sequence range (tick the "Trim Sequence" option).

The next option allows mean composition values for pre-defined groups to be calculated, selected as described above (Sequence Selection). The options are:

Single:	List the composition of each sequence individually.
Tags:	Calculate mean composition of each sequence group labelled with different tags in the selected group.
All:	Calculate mean composition of all selected sequences, irrespective of their tag group assignment.

The penultimate options species the format of the output file. (L)ist format creates separate tables of mononucleotide counts and frequencies, dinucleotide counts, frequencies and ratios to expected frequencies, and finally three tables of codon counts, frequencies and relative synonymous codon usage (see below). The (T)able output option creates a much larger set of analysis indices, with the data arranged on a single line for each sequence or codon position. This output format is useful for larger scale and comparative analyses of large numbers of sequences or sequence groups, and is directly importable into spreadsheet programs and statistical packages. Tables are cumulative; selection of an existing file will append data to the table rather than replacing it; column 1 records the analysis name selected at option 3 of the menu so that individual analysis runs can be identified.

If (L)ist format has been selected for output, analysis can be restricted to nucleotides, dinucleotides or codon usage. There are selectable by entering "Y" in the appropriate check boxes on the "Select Statistics" option box.

The program displays a progress meter while running, although it is relatively fast even with large datasets. The following sections describe the output values created by the program:

Nucleotide frequencies. Analysis of nucleotide composition produces a total of three sets of results. The first provides totals of each nucleotide and ambiguity codes, unknown bases and alignment

gaps (List format only). These may be listed for different codon positions, and by sequence or sequence group depending on selected options (see above).

The second table (List format only) lists the total numbers of unambiguous bases used for composition analysis, subdivided by codon position and/or tag group as selected.

The third set of data (recorded as a separate table in List format, and as separate columns in Table format) lists the base frequencies, using the total number of unambiguous bases as the denominator.

Dinucleotide frequencies. In List format, three tables are created recording total numbers of each of the dinucleotides (excluding any ambiguous or unknown bases, or alignment gaps), frequencies of each dinucleotide, and in the final table, the ratio of the observed frequency of each dinucleotide to the frequency expected from the base composition of the sequence (using data from the nucleotide composition calculation; see above).

The tables can be divided by codon position, or by sequence group. Dividing by codon position produces separate totals and frequencies for the dinucleotides spanning codon position 1 & 2, 2 & 3 and 3 & 1.

In Table format, separate columns record dinucleotide frequencies (prefix: "f"), ratios to expected frequencies based on mononucleotide compositions (prefix: "r"), and a newly developed output prefixed "cr". These represent ratios to expected dinucleotide frequencies that are further normalised by considering the amino acid usage of the nucleotide sequence. The correction is based on the observation that the amino acid sequence of an encoded protein predominantly dictates base composition at first and second codon positions, and a proportion of 3rd codon position choices. Non-random choices of various amino acids may lead to skewed mono- and dinucleotide frequencies at each codon position that is untypical of the overall base composition of the sequence. As an example, it is quite easy to conceive of a sequence that required several glycine residues, but otherwise showed a low G+C content. Taking amino acid content into account would therefore correct for the over-representations of the frequencies of the dinucleotides GG (at the 1st codon position), GN (2nd position) and NG (3rd position) that resulted from this glycine codon excess in the sequence. This correction method has not been extensively applied in my own or others' research but potentially represents a much better method to analyse dinucleotide frequency biases in coding sequences.

Codon usage. In list format, three sets of 3 x 4 x 4 tables are created, listing the totals of each coding triplet, frequencies of each codon using the total number of codons in the sequence as the denominator, and a third table that records relative synonymous codon usage (RSCU; (Sharp & Li, 1986)). This records the relative frequencies of each codon specifying an individual amino acid. If a sequence encoded 20 glycine residues and it contained 5 GGA codons, 10 GGC codons, 3 GGG and 2 GGU codons, then their RSCU values would be recorded as 1, 2, 0.6 and 0.4 respectively. RSCU values do not correct for base composition or dinucleotide frequency biases that would naturally be expected to skew codon usage, and represent a relatively crude analysis of potential biases in codon use.

In Table format, frequencies (prefix "f") and RSCU of each of the 64 codons are listed in separate columns. This is followed by amino acid totals and frequencies (fMet, fTrp *etc.*). The next four columns records effective codon number (ENc) and associated data (Wright, 1990). These comprise G+C content at third codon positions (GC3; required for calculation of expected values), the observed ENc value for the sequence or sequence group (Obs_Enc), the expected number of codons based on G+C content (Exp_ENc), and finally the ratio of observed to expected values.

If advanced program access is activated, the final column records the Codon Pair Bias (CPB) of the amino acid sequence. This is a composite metric calculated on the mean overrepresentation or underrepresentation of codon pairs compared with their random expectations (Gutman & Hatfield, 1989). Its significance is based on a hypothesis that non-typical utilisation, particularly the choice of rare codon pairs, influences the translation rate of mRNA. It has been proposed as a mechanism underlying the attenuation of mutant polioviruses and influenza A viruses in which codon pairs that are rare in mammalian coding sequences have been deliberately selected (Coleman *et al.*, 2008; Mueller *et al.*, 2010). The standard installation of SSE uses a codon pair frequency table (in the file "CPB_Bias.TXT" in the "SSE_v1.4 Userfiles" directory) based on human coding sequences that was provided in Tulloch et al. (2014).

It is however, possible to substitute alternative codon pair frequency tables based on other organisms if they are formatted identically to the file provided. This should be named as "CPB_Bias.TXT" and placed in the "SSE_v1.4 Userfiles" directory (keeping a copy of the original file perhaps!). CPB tables are large and complex to generate. To assist in this, an option to generate a CPB table from an existing sequence dataset is provided in the Composition Scan menu (Option "Create Codon Pair List"). This will be initially named as <UN> Composition CP List.DAT and will have to be renamed and copied if to be used for further composition scans. Note that these tables are only useful if generated from a substantial amount of coding sequence data (*eg.* all mRNA sequences from a specific organism). <BACK TO RESEARCH OPTIONS>.

Program	Scope
SEQUENCE CHANGES	A, S

This analysis program catalogues sequence differences between sequences or set of sequences, and uses a parsimony or a consensus sequence to infer the direction of sequence change between descendants and ancestors. Parsimony is a better method than construction of a consensus sequence to infer the direction of sequence, as it better able to reconstruct ancestral sequences from which the diversity in the dataset derived from. To use Parsimony, the sequence alignment will need to be run through the Program "CREATE PARSIMONY" (from the Utilities menu) to generate a coherent set of nodes to link the descendant sequences.

In the parsimony method, only sequence changes between a descendant sequence or node and its immediate ancestor are scored, allowing both the nature of the nucleotide change and its direction to be inferred (without multiple scoring where there are many similar sequences). The consensus method relies on creating a sequence consensus from the whole alignment, and recording sequence changes and direction of change between it and each sequence in the dataset. This method makes the assumption that the consensus is the ancestral sequence, although there are many reasons why that might not be the case. This issue is discussed extensively in the literature in connection with other forms of analysis by parsimony and will not be elaborated further here.

After selecting sequences to be analysed, an option box will appear asking whether "Changes from Consensus" or "Parsimony changes" should be counted (this will not appear if "Create Parsimony" has not been run first). The third alternative is to include a pre-determined ancestral sequence by other external methods. To use this option, place the cursor line on the ancestral sequence and changes between it and the selected sequences will be recorded.

The main menu allows standard options (location and filename of output file, sequence range) to be selected (lines 1-4). Line 5 provides the option to list changes in individual sequences rather than totals. Line 6 allows separate tallies of sequence changes to be computed for each of the three

codon positions using the currently selected reading frame to define codon boundaries. Lines 7 and 8 allow sequence changes to be separately listed based on their immediate context, *ie.* separate lists for changes in upstream of downstream of the four nucleotides A, C, G and T. This has been included to study, for example, the mutagenic effects of methylation at certain dinucleotide positions (*eg.* CpG*)*. Line 9 enables frequencies of each bae around the sequence change (an extended context) to be calculated and listed in output file. Other output option include a listing of where sequence changes occurred in an alignment and a list of amino acid changes produced by the identified nucleotide sequence changes. The final option (Maximum Variability) when the consensus sequence-based method is used provides a filter to restrict sequence changes to those occurring below a specified frequency. Sites with low frequencies of sequence changes are more likely to represent directional changes from a consensus.

The information recorded in either format comprises the number of sequence identities and changes on analysis of the whole dataset (prefix: "n"), the frequencies of identities and changes (prefix: "f"), and finally the frequencies of changes only (prefix: "cf"). If the "Analyse by codon" option is selected, then the output file will contain separate lines or tables for all sites, and each of the three codons listed in turn. As with some other analysis programs, output files are concatenated if a file of the same name as a pre-existing one is selected. This is particularly useful for the Table option, as it allows a composite table to be generated with different datasets. <BACK TO RESEARCH OPTIONS>.

Program	Scope
ASSOCIATION INDEX	A, S

This program was developed to score the degree of phylogenetic segregation between groups of sequences (Wang *et al.*, 2001). The method uses a tree scoring method that analyses trees constructed by PHYLIP for incompatibilities between the phylogenetic grouping of the sequences with the group membership (as pre-defined by tagging is this implementation). The association value *A*, is the sum of the dispersion values from each node (1 to *n*) within the phylogenetic tree. The dispersion value represents the degree of heterogeneity of group membership among sequences below each node, calculated using the following formula:

$$A = \sum_{1}^{n} t - g_{\max} / 2^{t-1}$$

where t = number of sequences below node, and f_{max} is the number sequences in the most abundant group. The Association value calculated for native sequences is compared with the nul expectation, *ie.* the score of a tree where group membership is randomly re-assigned to produce the Association Index (A of native sequences divided by the mean A of group re-assigned trees).

To allow for differing degrees of robustness of the trees used for AI calculations, the comparison between native and re-assigned control values is carried out on multiple times on bootstrap re-sampled datasets, either using nucleotide re-sampling as implemented in PHYLIP, or by carrying out repeated analysis on random subsets (66% of total) of sequences taken from each pre-assigned group.

This program requires that the programs SEQBOOT, DNADIST and NEIGHBOR in the PHYLIP package are installed. These programs will be located automatically if they are installed on the same drive as SSE. If not found, then their location can be specified from the "Default File Locations" option in the File menu. A related program (SEQUENCE GROUPING) that uses the same

tree scoring method calculates the degree of group membership of a test sequence in two or more pre-defined sequence groups (see next section).

Selected sequences are arranged with an outgroup for phylogenetic trees used in the AI calculation on the first line of the alignment. The selected sequences below must be labelled into two or more groups by tags as described above (SEQUENCE SELECTION). The program will not run and an error reported if this is not done. The "Association Index Parameters" menu allows a series of standard options to be selected (output filename, sequence range, saving of a new default file name). Other options are:

Option 6: The number of data bootstrap re-samplings
Option 7: The number of group re-assignments used for calculation of the tree score of the nul expectation for each tree (usually 10)
Option 8: The percentile range of grouping scores between bootstrap replications outputted into the summary file
Option 9: To specify whether bootstrap re-samplings are of nucleotides or sequence subsets
Option 10: Neighbour joining trees are used for phylogeny, and this allows the method for calculating distances to be specified.
Option 12: Whether to run a sequence scan. When selected, the alignment is analysed as a series of sequence fragments of length and overlap selected in a subsequent purple menu ("Enter Scan Variables"). The output files generated when this option is selected are formatted with extra columns to indicate the sequence fragments analysed. For this reason the word "Scan" is inserted in the filename to keep the output files separate from those of other analyses.

Finally, the various types of group comparisons possible when three or more sequence groups are analysed can be selected from the "Select Scanning Groups" menu. This allows a composite tree score and association index to be calculated by analysis of all of the sequence groups together. This provides the most sensitive test of whether there is segregation between groups, but it does not calculate which group(s) is/are separate. The second option allows pairwise comparison of each sequence group with every other group in turn to be automatically carried out. This is naturally limited to subsets of the data, but does provide more interpretable data when several groups differing in their segregation are analysed. The final option is to compare each group in turn with a new group formed by temporarily combining all other groups together. This allows a direct indication of whether any particular group is more phylogenetically distinct than any other. All three types of analysis can be carried out at the same time by entering "Y" into every check box.

The program generates two output files. One combines results of the replicate bootstrap re-sampling of the data, providing mean and percentile ranges of tree scores for native sequences, for re-labelled control sequences, and finally the Association Index (the mean tree score of native sequences divided by the mean tree score of the controls). For comparisons of non-segregated groups it is expected that 50% of bootstrap re-sampling will produce tree scores for native sequences greater than those of control sequences, and 50% will be lower. For segregating sequences, most of all native sequence scores will be lower than controls. These figures are indicated in the final column. Each of the result lines is appended to the same file, even for different analyses if the same output filename is selected.

The second file is formatted similarly except that the tree scores for native and control sequences are listed for each individual bootstrap replication. This enables more detailed analysis of the distributions of values during re-sampling to be carried out in a statistics package.

This program has been used extensively to investigate segregation of different variants of HIV in different anatomical compartments (brain, lymph node, etc.; (Wang *et al.*, 2001),, and between

HCV variants infecting injecting drug users in different cities in Scotland and elsewhere (Cochrane *et al.*, 2002) <BACK TO RESEARCH OPTIONS>.

Program	Scope
GROUPING SCAN	A, S

This program used the tree scoring method of the Association Index program to calculate the existence of group membership of a test sequence in two or more pre-defined sequence groups (see "Single Sequence Scan", below).

For this program, the query sequence is the first in the selected block, and the sequences assigned to tag groups below. The program scores how deeply embedded a query sequence lies within each of the clades formed by standard phylogenetic analysis of pre-assigned groups (classified by tag groups). For the Grouping Scan, the grouping score of a test sequence in a phylogeny is computed by reference to the assigned groups of its nearest neighbours in unrooted trees. Grouping scores (G) for a predefined group, a, can be represented by the following formula:

$$G_a = \sum_1^y 1/2^N$$

where N = the number of nodes separating the test sequence from each member of the group, and y is the total number of sequences in group a. By definition, the total of grouping scores of each pre-assigned group, G_a to G_n,, adds up to one. For robustness, each score is computed as the mean of a pre-defined number of bootstrap replicate trees (specified as a menu option) generated from permuted datasets where either the nucleotide sequence data (using SEQBOOT in the PHYLIP package) or the population (selection of a random 2/3rds of sequences from each pre-assigned group) is re-sampled. A high value indicates no segregation of the test sequence from the pre-defined group, indicating that it clusters with it. Low values show the converse. To localise sites where sequence groupings changed (frequently indicative of recombination), the program automates the analysis of sequential fragments through an alignment, where fragments lengths and step sizes through the alignment can be entered. Multiple analyses of several query sequences can be carried out sequentially by assigning additional sequences to the same tag group as the sequence in the first line.

This program requires that the programs SEQBOOT, DNADIST and NEIGHBOR in the PHYLIP package are installed. These programs will be located automatically if they are installed on the same drive as SSE. If not found, then their location can be specified in the "Default File Locations" option in the File menu.

The run options and format of the output file are the same as those of the Association Index program (PHYLIP location, output filename and bootstrapping options; see options for the ASSOCIATION INDEX program, above). As for the Association Index program, there are two output files, one that records the individual bootstrap values and one for mean values.

This program can be used to scan successive fragments across sequences, and therefore provides a powerful, alternative method to identify recombination events. In this latter application, it generates results that are formatted similarly to those of the SIMPLOT program. While SIMPLOT may predict recombination events in regions with no evidence for phylogenetic segregation between the sequence groups, this would be detected using the Sequence Grouping program. This program has been used for the investigation of recombination in hepatitis B virus (Simmonds & Midgley, 2005), and a formal comparison with other methods for detecting sites of recombination is planned. <BACK TO RESEARCH OPTIONS>.

Program	Scope
BOOTSCAN	A, S

This program implements the bootscanning method first described by Salminen *et al* .(1995) for investigation of recombination in HIV sequences and subsequently implemented in the program Bootscan in the Simplot program (http://sray.med.som.jhmi.edu/SCRoftware/simplot/). Its inclusion in SSE enables easier comparison with recombination detection using the Grouping Scan described above. This program requires that the programs SEQBOOT, DNADIST and NEIGHBOR in the PHYLIP package are installed. These programs will be located automatically if they are installed on the same drive as SSE. If not found, then their location can be specified in the "Default File Locations" option in the File menu.

Sequences assigned to different tag groups (minimum of two required) are used to generate consensus sequences (analogously to sequence groups used in GroupingScan), using a defined threshold value (50% by default) for creation of group consensus sequences. Other parameters that can be specified include the number of bootstrap replicates to perform, the method for measuring pairwise distances for the neighbour-joining tree and an option to scan the alignment (default). For this program, the query sequence is the first in the selected block, and sequentially to subsequent sequences assigned to the same tag group below. <BACK TO RESEARCH OPTIONS>.

Program	Scope
TREEORDER SCAN	A, S

This research menu option provides access to several methods that analyse and compare sequence orders in phylogenetic trees. Typically the program is used to compare phylogenies at different positions in sequence alignments, *eg.* across a viral genome or set of concatenated genes. Changes in phylogeny often provide evidence of recombination, re-assortment or other forms of linkage disruption in the past evolution of the sequence domain.

At the core of all the program outputs is the symmetric difference or partition metric (Robinson & Foulds, 1981) that compares the branching orders of phylogenetic trees, and enumerates the number of phylogeny violations between them. The method has been adapted to compare trees with branches collapsed to (user-specified) bootstrap support values, and has therefore to compare trees with differing degrees of phylogenetic informativeness (rather than using completely resolved trees with strictly bifurcating branches in the original algorithm). Thus, violations are only scored if a sequence or a sequence group changes position relative to bootstrap-supported clades present in both trees. Boot-strap re-sampling of sequence data is frequently used to determine the frequency with which individual sequence groupings occur, an indicator of their "robustness". In the new algorithm, congruence of phylogenies ignores groupings supported by bootstrap values below that specified by the user.

The basic program outputs comprise:

a) The positions of changes in tree order across an alignment. In this case, tree orders only change when forced to by re-grouping of sequences into different, bootstrap supported clades.

b) Alignment scans that record the phylogenetic informativeness of each position in an alignment. Sequence order is effectively randomised, and its restoration by phylogenetic analysis recorded by the ordering of sequences in the tree.

c) Frequencies of phylogeny violations between trees for a pre-specified bootstrap value.

d) Listing of individual sequences or sequence groups with different phylogenetic relationships in different trees, and which are putative recombinants.

Output files containing results from each of these forms of analysis are generated. The following summarises the main menu options, and then describes the analytical options available.

MENU 1 - TreeOrder Scan Variables

On the main menu, the first four lines allow the data destination, directory, generic name of the output file and analysis name (used in cumulative files) to be specified. The last line (Option 21) allows these options to be saved as default. Option 20 allows the parameters selected for the program run to be printed a header on output files (useful for archiving files, but information that has to be stripped off when using the table for analysis). Option 14 allows a specific random number seed to be entered that governs the randomisation of sequence order in the Phylogenetic Signal Scan ("R" specifies that a random seed is generated each time). Options 5 and 6 define the region of the alignment to be analysed, while Options 10 and 11 specify the fragment size and step size used for the alignment scan. Fragments so defined correspond to the sequence sets used for construction and comparison of phylogenetic trees.

The other options govern the more specific aspects of the analysis. Option 9 allows an outgroup to be specified for phylogenetic analysis of each sequence fragment (placed on the first line of the alignment). Using an outgroup is not essential although it allows the analysis of rooted trees. Option 7 specifies the number of bootstrap replicates to be carried out, while Option 8 allows the bootstrap threshold(s) to be set. A single value can be entered here (*eg.* 70%), or "M" which allows up to 12 different values to be entered on a subsequent menu. Use of multiple thresholds is frequently of value as this variable has a crucial effect on the analysis.

Option 13 specifies whether the neighbour-joining phylogenetic trees used for tree constructions are to be based on nucleotide distances at all sites (A), synonymous sites only (S) or non-synonymous sites only (N). For each distance calculation, a simple Jukes-Cantor correction for multiple substitutions is made (more elaborate models have little or no effect on branching order or bootstrap support for clades, and have therefore not been implemented).

Option 15 allows the user to specify the minimum completeness of each sequence in every fragment used for tree construction. Because tree comparisons require the same number of sequences, any sequence that is incomplete in any region of the alignment will be globally excluded for the analysis. The number of excluded sequences will be notified before the program runs. It is advisable not to include ragged ends of alignments since some sequences may be unnecessarily omitted if they fail the completeness test. The minimum selectable completeness is 10%, reflecting the fact that tress constructed from such few nucleotides will be of little value in the analysis.

Specifically for some forms of grouping analysis, Option 16 allows sequences that constitute the sole member of a sequence group to be automatically excluded. Their inclusion may or may not desirable depending on the analysis method.

Option 17 when selected, specifies that the sequences or sequence groups responsible for phylogeny violations are listed in additional data columns in the Violation_List file. This can be useful for identifying individual recombinant sequences or groups, but would not be required is the data is simply to be used for constructing violation matrices (see below).

Option 18 specifies the minimum group size used to refine the analysis of tree violation frequencies. In the Violation_Frequency file, total and normalised frequencies of violations are reported for the whole alignment, figures which are then split into those involving sequence groups equal to or below the minimum group size, and those that are greater than the specified value. The purpose of this subdivision is to differentiate between phylogeny violations involving single or small groups of sequences from those involving whole clades (such as changes in the branching order at the base of the tree) and which, for example may not always be indicative of recombination.

Option 19 allows automated repeated analysis of the dataset in which each sequence in turn (apart from the outgroup) is removed. Comparison of violation frequencies for each analysis therefore provides an alternative means to identify individual sequences that are responsible for phylogeny violations

Option 12 specifies the types of analysis and outputs; if "(S)elected" is chosen then a subsequent menu allows each of the six to be individually chosen from Menu 2.

MENU 2 – Analysis Methods Selection

This appears if Option 12 was set to "(S)elected", allowing each of the six types of analysis to be individually chosen. For all analysis option, data produced is exported as Tab-separated variable ASCII (text) files which can imported in spreadsheet, statistics and graph-plotting programs for analysis and display. Graphical outyput can be selected for scans. Common features of these files are:

> Optional presence of "Run Information" (selected from Option 20 of Menu 1), which is a listing of the settings and options selected for the analysis. This can be useful data for archived files.
>
> A variable number of data columns may be added to the start of data fields depending on the analysis options chosen. The initial column records the selected bootstrap value if multiple values were chosen. The next two identify the nature of sequences being analysed. If a control analysis was carried out (see below) then the first indicates whether the sequences analysed were native sequences or control sequences, while the second records the control dataset number if more than one was run. Both columns are omitted if control data is not analysed. The final optional column records the sequence excluded if sequential exclusion is carried out (selection from Option 19 on Menu 1).

The analysis options comprise the following:

1) *Retained Sequence Positions*. This analysis method conducts a scan of the positions of individual sequences in sequentially generated trees (specified by fragment length and step size; see above) across the alignment. The algorithm attempts to retain the order of sequences through branch rotation and movement of sequences within clades, provided their groupings are supported by less than the specified `bootstrap value(s). The output file named <UN>Array_R.DAT comprises an array that records the positions of individual sequences listed in separate columns in the first row (starting positions in the second row). Successive lines plot the migration of individual sequences

in sequentially generated trees (mid-points of fragments recorded in left hand column). If multiple bootstrap threshold values have been selected, then the first column records the bootstrap value for each set of sequentially generated trees. The format of this file is designed to allow individual sequences to be plotted as separate data series in programs such as Excel, where the bootstrap values in the first column can be filtered to produce multiple analyses. Graphical output can be produced for each bootstrap value, although this is limited to datasets with 100 or fewer sequences (if more are selected, graph output will not be generated).

Bootstrap values in the range 65% - 75% are the most suitable for this type of analysis.

The associated file (<UN>Boundaries_R.DAT) is a list of the positions of boundaries between bootstrap supported clades in sequentially generated trees across the alignment. Tree positions of the "(S)tart" and "(E)nd" of each clade ("Transition"; column 4) are listed in the column headed "Rank" (Column 6). (These column numbers and all those subsequently referred to refer to tables containing a bootstrap threshold value column in column 1 [*ie.* when multiple bootstrap thresholds have been used]. If not, then column numbers are one less than listed.) Boundaries of clades can be superimposed on the graph of tree positions of individual sequences. The output option is restricted to bootstrap values of greater than 50%.

2) *Phylogenetic Compatibility*. This analysis records congruence between phylogenetic trees. In this implementation, pairwise comparison is carried out between trees constructed from each sequence fragment, recording the number of phylogeny differences (violations). As described above, the Robinson-Fould algorithm used to compare trees minimises changes in the tree order of sequences by branch rotation, and by movement of sequences linked by branches below the pre-specified bootstrap threshold value(s). The number of remaining differences in sequence order is recorded separately for order differences between pre-defined sequence groups (group events) or between sequences within groups (intra-events), and split into those involving group sizes above and below that specified in Option 18 in the previous menu.

This analysis option produces graphical output (half-diagonal matrix) and data files. The firstdata file (<UN>Violation_List.DAT) records the numbers of violations for each fragment comparison. The data columns comprise the following:

> Columns 1, 2, 3 and 4 (following the optional data columns described above) record the mid-point position and number of the fragments being compared. Columns 5 and 6 record the number of clades in trees generated from the two sequence fragments. If both contain more than one clade then the subsequent columns record violation frequencies broken down in various ways.
>
> Violation data is reported in 3 sets of 9 columns each. The three sets comprise AG corresponding to all phylogeny violations, while the BG and WG sets break this global figure into violations recorded between pre-defined groups, and those occurring entirely within groups. The latter might correspond to intra-serotype recombination events, while the BG columns record violations between members of different serotypes in this example. Each set of data is broken down in the same way. The first column records the total number of violations arising from the comparison of the two trees. This figure is then normalised by calculation of violation per sequence, and violations per clade. Data in these three columns is then further broken down into total and normalised frequencies of violations arising from sequence groups equal to or below the pre-specified group size (Option 18 in the first menu) or above.

There are a final two optional columns that list the sequences or sequence groups responsible for the sequence violations. Whether these additional data are included is specified by Option 17 of

Menu 1. The two columns separately record sequence or sequence groups of size equal to or below the maximum group specified in Option 18 of Menu 1, and sequence groups greater than the specified size.

Bootstrap values in the range 65% - 75% are the most suitable for this type of analysis.

3) *Phylogenetic Signal Scan*. This analysis option operates exactly as described for Analysis 1 (Retained Sequence Positions) except that rather than trying to retain the order of sequences when comparing trees, sequence order is randomised to the extent permitted by the phylogeny of the sequences. Therefore only sequences within clades with bootstrap support above the pre-specified threshold(s) will remain grouped together in the output sequence order. This method is effective at identifying phylogenetically informative regions of a sequence alignment (where sequences of the same species or genotype group together); in regions without phylogenetic information, these groups will become randomly dispersed in tree order. As for Analysis 1, the most effective way to represent the output from this analysis is by plotting the position of each sequence in each tree generated across the alignment (x-axis) as a data series (position recorded on y-axis). In this case, genotypes or species can be colour coded to indicate the extent to which their grouping is preserved.

Output files comprise arrays of sequence positions (rows) for each sequence (separate columns) I a file designated as <UN>Array_S.DAT, and a separate list recording the boundaries of phylogenetically supported clades (<UN>Boundaries_S.DAT).

4) *Group Segregation*. This analysis option uses the same tree-ordering procedure used in analysis <METHOD 3> (Phylogenetic Signal Scan), but records the degree of heterogeneity (or mixing) of sequences from different pre-assigned groups (*ie.* labelled with different tag number in this implementation; selected as described above: <SEQUENCE SELECTION>). Regions of the genome with phylogenies that support the assignment of sequences into the pre-assigned groups will show low heterogeneity values, while those without phylogenetic signal will high values.

Graphical and data file output is available for this program. For scoring, the heterogeneity values are recorded in column 4 of the cumulative table in the <AN>Signal_Scan.DAT file. This represents the number of tag label changes across a tree (*ie.* from first to the last sequence) divided by the number of sequences. The y-axis of graphical output and Column 5 in the output file provides the ratio to the expected value, with values normalised to values from zero to one, where zero represent a tree with sequences perfectly segregated by their pre-assigned groups, and one where there is no association between group assignment and tree position.

A bootstrap value of zero are the most suitable for this type of analysis.

5) *Clade Distribution*. This records the number of bootstrap-supported clades above a pre-specified bootstrap threshold in trees from each fragment in the alignment (filename <AN>Clade_count.DAT). The table provides data required for other aspects of phylogenetic tree analysis.

6) *Summary Data Values*. This produces a cumulative file (<AN>Violation_Freq.DAT) in which total segregation, clade count and violation frequency data is summarised for each analysis (total and mean values from analysis of the whole alignment). The initial columns record the analysis name, the alignment boundaries, number of fragments, fragment size, number of pairwise comparisons of fragments followed by the mean number of bootstrap-supported clades per tree, mean segregation value and index, followed by summary data on phylogeny violations. This is arranged in a similar way to the data in the <UN>Violation_List.DAT file, which records violation data for each individual fragment comparison, except that the data is further normalised by

expressing results as mean values per fragment comparison, followed by further division by the number of sequences and the number of clades.

7) *List Recombinant Sequences*. This option creates two output files, <UN>Recombt_List.DAT) and <UN>Recombt_Freq.DAT records the numbers of violations broken down into group sizes. The data columns in the first file comprise the initial optional columns, followed the group size of the group involved in the phylogeny violation. Subsequent columns record the violation totals associated for each individual sequence. An individual recombinant sequence in the data would produce a high value in the group size 1 row, and perhaps in larger grouping with other sequences.

In the second <AN>Recombt_Freq.DAT (cumulative) file, the initial data for the program run is followed by combined data on the number and proportion of different sequences in the alignment that were involved in phylogeny violations, followed by two columns that record the total number of phylogeny violations and the mean number per fragment comparison. Each is broken down in separate rows to indicate the sizes of the clades involved in the phylogeny violations. <BACK TO RESEARCH OPTIONS>.

Program	Scope
FOLDING ENERGY SCAN	A, S, I

This program uses RNAFold to calculate the minimum folding energy (MFE) of a test sequence and to compare this value with the MFE of the same sequence scrambled in sequence order (see section on <PROGRAM INSTALLATION> for further information). The difference (MFED) value represents the sequence-order dependent component of RNA (or DNA) folding. A second Z score statistic records the position (in standard deviation values) of the native sequence within the distribution of control values. The analysis can be performed on single sequences or sequence groups (where values will be averaged). MFED values can be computed as a scan across a sequence alignment where sequence fragment lengths (typically 200-300 bases) and increments can be specified. This can be used to localise areas of RNA secondary within a genome.

Although seemingly simple conceptually, the validity of MFED and associated Z-score calculations depends on the method used for sequence order randomisation. Retention of any biases in dinucleotide frequencies in the native sequence is generally essential. As an example, normalising frequencies of self-complementary dinucleotides (normally underrepresented in mammalian sequences) such as CpG), reduces the likelihood of self-complementarity and generates an MFED value that is artifactually high (Rivas & Eddy, 2000). Other more elaborate methods designed for coding sequences retain codon structure and in the most complex case, dinucleotide frequencies as well.

Scrambling methods comprise the following:

1) NOR Completely randomises sequence order within a sequence. Not recommended (see above).

2) NnR Randomises sequence order while preserving frequencies of dinucleotides (NDR), trinucleotides (NTR), tetranucleotides (NRR) or pentanucleotides (NPR) found in the native sequences. These options are selectable in a subsequent menu ("Preserve"). The extent of sequence randomisation can be controlled in the last menu ("No of Sequence Changes") to enable similar levels of sequence scrambling to be achieved by the different methods (see (Simmonds *et al.*, 2004)..

3) NnS This is the same as NnR, except that randomisation is carried out by swapping of bases between adjacent matched sites rather than global randomisation.

The remaining methods are restricted to coding sequences, and use codon boundaries defined by the reading frame used in the sequence editor.

4) COR Completely randomises the order of codons
5) CLR Randomises the order of the complete set of codons specifying each amino acid. The resulting sequence has the same amino acid sequence as the original.
6) CLS The same as CLR except that matching codons are swapped rather than globally scrambled.
7) CDR Scrambles codon order while preserving dinucleotide frequencies (that may be biased in the native sequence).
8) CDS As CDR, but where randomisation is limited to swaps between matched codons.
9) CDLR A complex scrambling method that both preserves the coding sequence and dinucleotide frequencies. This is the option to choose in most cases for coding sequences, as it is the least disruptive to sequence ordering.

In general, NDR and CDLR should be use for non-coding and coding sequences respectively.

A series of menus allow entry of the region to be examined, the number of sequence order-scrambled control sequences for MFE comparisons, and the option to scramble individual sequences within an alignment or coordinate sequence exchanges between sets of selected sequences. The latter is of value in preserving the phylogeny of the scrambled sequence dataset (required in some circumstances) but is restrictive in that constraints on sequence change have been followed for every sequence.

A parallel screen of reverse complemented sequences can be selected by ticking the box on line 6. The final option allows a scan of the selected range in the alignment using defined fragment sizes and increments. For sequences of several bases in length, averaging the results from a windowed scan (typically with fragments of 250-350 bases in length) is preferable to calculating MFED values from the whole sequence (because of sequence length constraints on RNAFold and because MFED values become artificially small from the inclusion of likely non-physiological long range pairings that emerge on *in silico* folding a long nucleotide sequence).

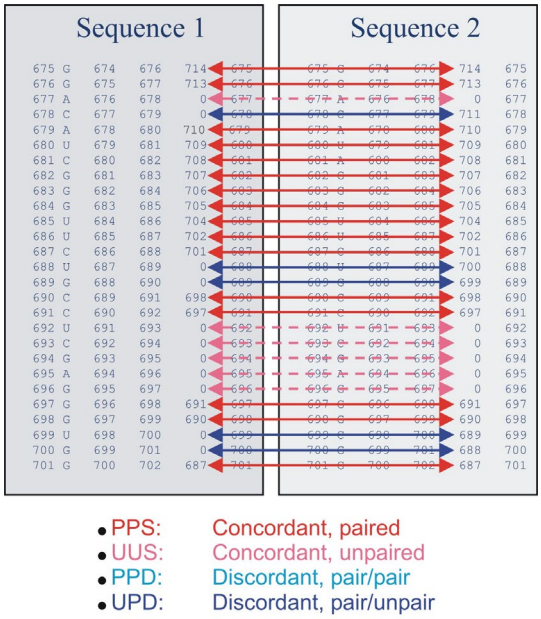
Output options include data destination (graph, output file), file location, file and run name, and format of the output files. Individual folding energies of each native sequence and control can be outputted, or alternatively a more useful format where the MFED and Z-score calculations are performed. A third option takes average values between sequences of the same tag group. This is very useful to reduce noise in the output.

Standard thermodynamic parameters (folding conditions, loop size constraints) are entered followed by the selection of sequence scrambling methods described above. Finally, if NnR (2nd in the list) has been selected, it is possible to enter the actual size of motif to preserve. As described above, preservation of dinucleotide frequencies is the norm but it is possible to preserve higher order motifs, although the numbers of these declines dramatically with motif length and become ineffective as a means for scrambling relatively short sequences. <BACK TO RESEARCH OPTIONS>.

Program STRUCTUREDIST

Scope A, S, I

StructureDist compares the most energetically favoured RNA structure predictions for a set of aligned sequences. Minimum energy structures are computed for each selected sequence by RNAFold and pairings for are each compared to identify conserved and non-conserved structure predictions. Many output options are therefore based upon pairwise comparison of each connect file with each other one, and listing in a variety of formats the result of pairing predictions of each nucleotide site.



This program requires that the program RNAFold is first installed (see section on <PROGRAM INSTALLATION> for further information).

For each site, a pairwise comparison between two structure predictions can have the following outcomes:

1) *PPS (paired, paired, same)*. Here the two connect files predict that for a site x, there is a predicted pairing with a downstream site, y, and that both files contain the same prediction.

2) *UUS (unpaired, unpaired, [same]).* For both sequences, site x is predicted to be unpaired.

3) *PPD (paired, paired, different)*. Site x is predicted to be base-paired in both structure files, but the downstream bases, y_1 and y_2 are different.

4) *UPD (paired, unpaired, [different])*. In the two structure predictions, one of the bases at site x is predicted to be base-paired, while the corresponding base in the other file in unpaired.

The results from comparison of connect files are output in a series of files and as graphs showing frequencies and positions of predicted pairs and unpaired bases across the alignment. Output files are tab-delimited text files and can be directly imported in spreadsheet or statistics packages for further analysis and graph plotting.

A menu allow entry of the data destination, output file path and name, sequence range and a series of output options: The available data outputs are as follows:

1) *Position File.* This lists the sums of outcomes of each pair-wise comparison of sequences at each nucleotide position in the alignment.

Pos	Cons	Comps	PPS	UUS	PPD	UPD	Freq[PPS]	Freq[UUS]	Freq[PPD]	Freq[UPD]
-344	G	780	92	231	61	396	0.1179	0.2961	0.0782	0.5076
-343	C	780	106	210	65	399	0.1359	0.2692	0.0833	0.5115
-342	C	780	106	210	65	399	0.1359	0.2692	0.0833	0.5115
-341	A	780	4	496	24	256	0.0051	0.6359	0.0307	0.3282
-340	G	780	67	105	233	375	0.0859	0.1346	0.2987	0.4807
-339	C	780	135	1	568	76	0.1730	0.0012	0.7282	0.0974
-338	C	780	144	1	559	76	0.1846	0.0012	0.7166	0.0974

The output format was used to display the frequencies of conserved paired (PPS; red) and non-non-paired (UUS blue) sites in the NS5B region of hepatitis C virus (Tuplin *et al.*, 2004). Graph output can be directly generated using this analysis option.

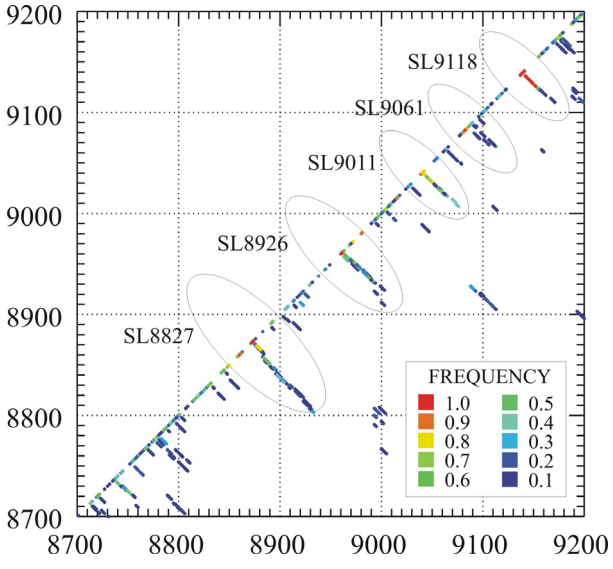
2) *Matrix Data.* This is a complete listing of every base-pairing and non-paired base found on pairwise comparison of all sites in all connect files. The output file lists the outcome of each comparison (*ie.* PPS, UUS, PPD, UPD), its frequency, X and Y coordinates, and spacing between upstream and downstream bases.

Result	Pairing	X_Pos	Y_Pos	Total	Frequency	Spacing
PPS	GC	76	55	1128	0.75960	21
PPD	GC	55	76	336	0.22626	21
PPD	GC	56	76	54	0.03636	20
UPD	G	-347	77	54	0.03636	
PPD	GG	53	77	54	0.03636	24
PPS	GC	77	54	1128	0.75960	23
PPD	GC	54	77	336	0.22626	23
PPD	GC	55	77	54	0.03636	22
UPS	C	78	78	1	0.00067	
UPD	C	-347	78	106	0.07138	
PPS	CG	78	53	1128	0.75960	25

This format can be plotted out directly as a coloured 2D dotplot. The output format of the graph can be selected to:

a) Show only concordant pairing predictions (PPS) or include discrepant pairing predictions (UPD and PPD). If the latter are selected, PPS pairing coordinates are shown in the right lower quadrant of the graph and UPD/PPD in the upper length. If PPS predictions only are selected they can be shown in either graph quadrant or both.
b) Unpaired base predictions can be shown on the x=y line.
c) A frequency threshold can be set over a continuous range from 0 – all pairing predictions to 1.0 (only completely conserved pairings). This can clarify the display of conserved pairings but showing the positions of all predictions can also be useful in other circumstances.

The following matrix plot shows concordant predicted pairings (PPS) and predicted non-paired bases (UUS) in the NS5B region of HCV allowing a series of predicted stem-loops to be visualised. Frequencies of occurrence have been indicated on a colour scale.



3) *Position Scan*. This is a list of total numbers of the different possible predictions (PPS, UUS, PPD and UPD) over the whole fragment analysed. The relative frequencies of each outcome provides an indication of the overall degree of RNA structural similarity in the connect files for each sequence.

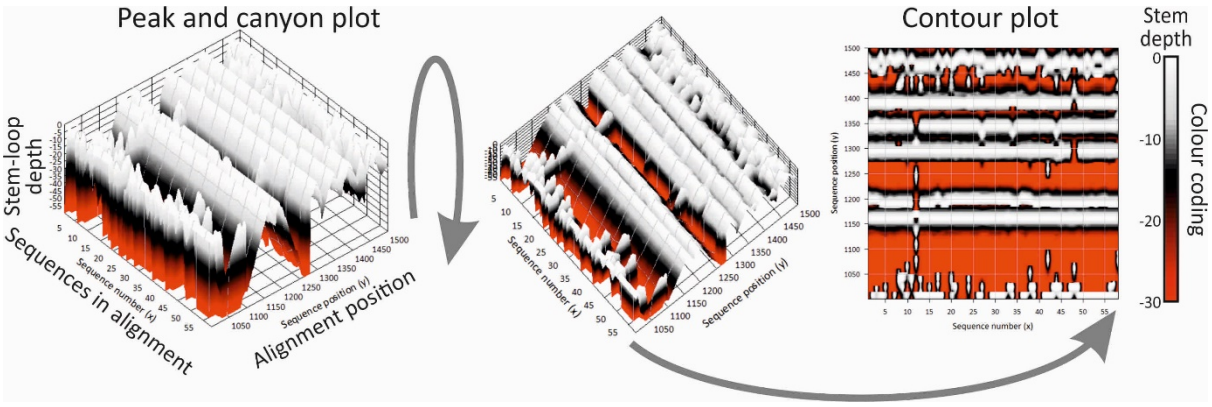
4) *Treefile.* The data from the pairing scan can be converted into a "distance" between pairs of sequences in terms of RNA structural similarity. The distance measure is 1 – the frequency of shared paired sites (PPS). The file can be formatted for MEGA or PHYLIP and used as a half-diagonal distance table for use in a neighbour-joining tree.

5) *Loop statistics.* This is output as a series of files (file extensions .DT1 – .DT4) listing the various structure elements and their degree of sequence conservation (.DT1), the frequencies of each base-pair in duplex regions and of each base in unpaired regions with differing degree of structural conservation (.DT2), the further breakdown of frequencies listed by upstream and downstream contexts (.DT3) and the numbers, frequencies and ratio of observed pairings to those expected from base composition of the sequences (.DT4).

6) *Pairing distances.* This produces a list of distances between pairing positions (up- and downstream bases) between sequences in an alignment.

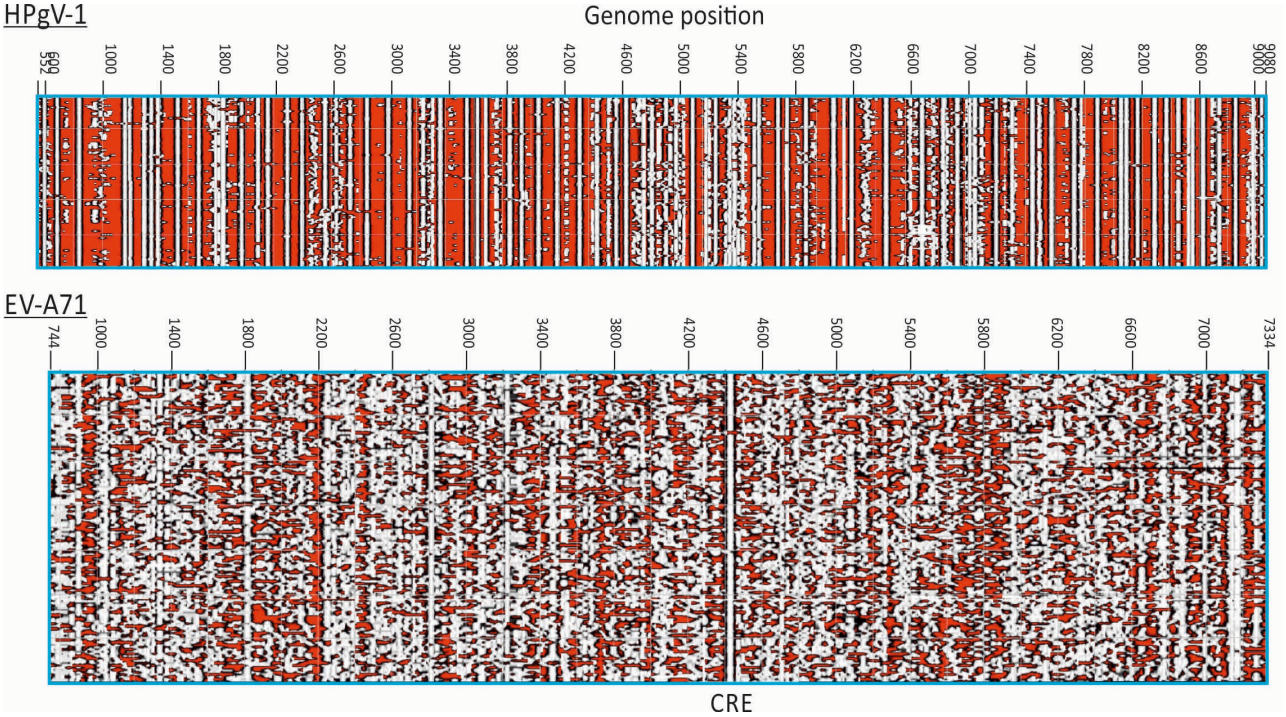
7) *Structure map.* This enable visualisation of RNA structure similarities and differences between sequences in an alignment. In this program, connect files generated from each sequence are analysed to identify the positions of predicted stem-loops within each, and to code their positions in terms of heights or depths. Height are calculated based on the unpaired flanking sequence being assigned as zero, while each paired position in a stem-loop increments the height of the bases involved by one. The unpaired terminal bases in a long stem-loop will therefore achieve a high height value. Alternatively, structure can be expressed as depths where the terminal unpaired bases in a stem-loop are assigned a depth of zero, while paired bases either side show increasing depth as cfar as as the base of the stem. Plotting depths is celearer as it enables the tips of all predicted stem-loops to be visualised.

   The program will output 3D (peak and canyon) or 2D (contour) plot representations of the output depths, arranged so that the x-axis records the alignment position, the y-axis, the sequence number in the alignment and the z-axis for depth:



As indicated, the depiction of the data with colour-coded stem-loop depths provides an effective way to immediately visualise the positions of predicted RNA structures within an alignment and the conservation of pairing between sequences.

The method is suitable for very large scale structure prediction, extending over whole virus genomes. The following output shows ta contour plot of 100 whole genome sequences of human pegivirus, and the occurrence of a large number of stem-loops throughout the coding region.



Underneath is a similar analysis of enterovirus A71, that possesses a largely unstructured virus genome. However, the conserved *cis*-replication element (CRE) around position 4400 can be readily observed.

The output file, named <UN>_Map.DAT from the program contain lists of x, y and z coordinates that can be used for other graphical representation of the output. A second file (<UN>_Het.DAT) records the heterogeneity of pairing predictions at each genome position.

The folding of sequences by RNAFold can be extremely time-consuming and the task can be allocated to more than processor on the computer to run in parallel ("Number of processors to use"). For multi-processor machines, the default value is one fewer than the number identified by the program (to avoid the computer freezing up), although this can be changed if required. RNAFold will generate several structure predictions of a single sequence within a user-specified range of MFE values. These can be generated and incorporated into the structure comparisons as they can contain data on prediction variability that assists in the identification of conserved paired and unpaired regions (equivalent to the pNum value generated by MFOLD on single sequences). Finally, there is an option to keep connect files generated by RNAFold for other analysis programs. <BACK TO RESEARCH OPTIONS>.

Program	Scope
COVARIANCE TEST	A, S

This program takes an alignment of sequences, and scans it for co-variant sites (*ie.* paired sites that are self-complementary, and which contain a minimum specified number of sequence substitutions that maintain complementarity). Following the identification of candidate covariant sites, the program then searches for adjacent potential base-pairing either side of the covariant site.

The "Covariance Settings" dialogue allows run parameters to be entered that govern the minimum requirements for scoring covariant sites ("Covariance value"), as well as the criteria for reporting adjacent RNA structure.

Scoring is based on the number of covariant substitutions away from the calculated consensus sequence (mismatches score as minus values), but has the disadvantage of assuming each change from the consensus is phylogenetically independent, a false premise when comparing an alignment of sequences forming a highly structured phylogenetic tree.

To avoid multiple scoring of phylogenetically linked substitutions, the sequence alignment can be run through CREATE_PARSIMONY, after which it is possible to select for Parsimony scoring (last but one option), in which case only sequence changes between sequence/node and its immediate ancestor node are scored. This issue is discussed in detail in (Tuplin *et al.*, 2002).

The other options in the Option box are largely self-explanatory. The minimum and maximum look-ahead values refer to the minimum and maximum spacing between the upstream and downstream bases scanned for covariance. As RNA secondary structure tends to be short-range, the maximum look-ahead value can be often set to 300-500 bases, as this greatly accelerates scanning speed.

G-U pairings can be scored as a fraction of the value of G-C and U-A pairings (default 0.9; option 10), while the minimum match for a covariant site is set in Option 11. Options 12-15 determine the scanning window used to detect adjacent RNA structure, and the fractional contribution of G-U pairings to the match calculation.

The output from the program comprises a listing of all the covariant sites detected in the alignment (".DAT" file extension), irrespective of whether adjacent RNA is detected. A second file ("SEQ file extension) draws the sequence alignments either side of the upstream and downstream covariant sites where the minimum specified RNA structure was found, marks any potential base-pairing that would be formed on RNA folding. <BACK TO RESEARCH OPTIONS>.

Program	Scope
MUTATE SEQUENCES	A, S, I

The program mutates a sequence or sequences and adds them into the alignment file. Each sequence can be mutated over a user-defined range leaving the rest unaltered. A random number generator is used to select the sites and base changes into the sequence and can be seeded with different numbers to generate independent sets of mutated sequences on repeat runs. The number of mutant sequences generated, their tag group assignment and labelling is selectable in subsequent lines. For example, new mutant sequences can be automatically tagged with the tag of the native sequence if required (Option 8). The copy number of the mutated sequence can be added to the sequence name (Option 9; default) or as a prefix to the sequence description (Option 10), or both. Modifying the sequence name is useful as it generates unique names for each mutant sequence.

For each mutational process, the composition of the mutated sequence(s) is automatically analysed and the results placed into an output file that includes the same compositional attributes of the starting sequence. Metrics measures in the mutated sequences include sequence divergence from the original sequence, mono- and dinucleotide frequencies and codon pair bias.

The next menu allows one of four different mutational programs to be selected:

1) *Global Sequence Divergence.* This allows mutations to be introduced into a sequence to achieve a specified degree of sequence divergence, a defined ratio of synonymous to non-synonymous mutations (dN/dS ratio), transition / transversion ratio and the tolerance of multiple substitutions ("Substitution Method"). If "Fixed" is selected, mutations will be added until the degree of specified divergence is achieved (if possible). If "Evolutionary Distance" is selected, multiple substitutions are allowed, and the degree of divergence achieved will be less than specified (although potentially reconstructable using the appropriate evolutionary model when calculating sequence divergence.

A final option is to specify special contexts, *ie.* allowing the immediate environment of the mutation site to influence mutation rates and bases substituted. As an example, a much higher rate to C->T transitions can be specified when followed by a G, mimicking the mutational effect of methylation in mammalian genomic DNA.

The next box allows the composition of the mutated sequences to be specified. It is possible not constrain this, to constrain it to match the composition of the native sequences (sensible default choice) or to impose a different composition ("Target Frequencies"). If selected, the target base frequencies at each codon positions can be entered with default values representing the base composition of native sequences.

3) *Mononucleotide frequency changes*. This allows sequences to be mutated to achieve a specific target frequency of each base. The program will introduce mutations into a sequence until the frequencies of the four bases match the specified values within the tolerance value provided (%). Note that only three base frequencies need to be entered as the fourth follows from these. This mutational process can be performed under constraints; selecting "Tick for coding sequences" ensures that the all introduced substitutions retain identical protein coding of the sequence. The "Invariant dinucleotide" option allows one of more dinucleotides to fixed in frequency and position in the sequences as it is mutated.

3) *Dinucleotide frequency changes*. This mutates sequences to achieve a specified target frequency of an individual dinucleotide. All frequencies are expressed as Observed / Expected values calculated from the frequency obtained in a random sequence with the same mononucleotide frequencies. This mutational process can also be performed under several constraints, including retaining protein coding ("Tick for coding sequences"), native mononucleotide frequencies and the option to fix frequencies of other dinucleotides.

4) *Change codon pair bias*. This allows the generation of sequences with a specified codon pair bias (CPB) score or the minimum or maximum possible values. (For an explanation of CPB, see the <CPB SECTION> in Composition Scan for further details.) Briefly, CPB is a composite metric that describes the overrepresentation or underrepresentation of codon pairs compared with frequencies based on the frequencies of individual codon and amino acid pairs in a reference organism (Gutman & Hatfield, 1989). By default, CPB is based codon pair usage in human mRNA sequences using the supplied table in the file "CPB_Bias.TXT" (in the "SSE_v1.4 Userfiles" directory). Alternative codon pair frequency tables based on other organisms can be used (see the <CPB SECTION> in Composition Scan for further details).

Mutating a sequence to achieve a specified CPB score retains the protein coding of the sequence. Optionally, the mutational process can be constrained so as to preserve frequencies of one or more dinucleotides in the sequence. A manuscript describing the purpose of such a procedure is in preparation. <BACK TO RESEARCH OPTIONS>.

Program	Scope	Output file
SCRAMBLE SEQUENCES	A, S, I	

The program selects individual or groups of sequences and scrambles their sequence order using a variety of algorithms that preserve different sequence ordering features, including the biased dinucleotide frequencies in the native sequence, and the encoded amino acid in coding sequences. These algorithms are equivalent to those described in Folding Energy Scan (see above)

Sequence randomisation can be carried out on a restricted sequence range (options 1 and 2 on the "Scramble parameters" menu, and the number series used for randomisation initiated from a user-defined seed. The number of copies to be generated can be selected (line 4), and the option to scramble individual sequences within an alignment or coordinate sequence exchanges between sets of selected sequences. The latter is of value in preserving the phylogeny of the scrambled sequence dataset (required in some circumstances) but is restrictive in that constraints on sequence change have been followed for every sequence. A parallel screen of reverse complemented sequences can be selected by ticking the box on line 6, and generation of a series of sequence fragments across an alignment (Option 7). Sequences can be tagged according to the tag of the native sequence from which it was generated (Option 8). Finally, the scrambled sequences can be incorporated into the sequence alignment at the end of the run, or exported to file in FASTA format, along with the native sequences that generated them (Option 10). The sequence description line of scrambled sequences generated by the program is updated with information on the randomisation method used, the copy number and the length of the sequence fragment scrambled (excluding gaps and unknown bases) and the degree of sequence divergence from the native sequence (if Option 9 is ticked).

If scrambled sequences are selected for export, then a further menu ("Export File Location") appears to allow the file location and name of the exported sequences to be specified (and stored as default; option 6). The sequences may optionally be prefixed with the tag name or number of the native sequence used to generate them. This may be convenient if sequences generated from different original native sequences need to be separately identified at a later stage in processing or analysis.

The next menu ("Select Scrambling Method") allows each of several different scrambling methods to be selected. These are described in the help section for the Folding Energy Scan. Subsequent menus are similarly described in the earlier section. <BACK TO RESEARCH OPTIONS>.

REFERENCES AND CITATIONS

A) References

Cochrane, A., Searle, B., Hardie, A., Robertson, R., Delahooke, T., Cameron, S., Tedder, R. S., Dusheiko, G. M., De, L., X & Simmonds, P. (2002). A genetic analysis of hepatitis C virus transmission between injection drug users. *J. Infect. Dis.* **186**, 1212-1221.

Chenna R, Sugawara H, Koike T, Lopez R, Gibson TJ et al. Multiple sequence alignment with the Clustal series of programs. *Nucleic Acids Res* 2003;31(13):3497-3500.

Coleman, J. R., Papamichail, D., Skiena, S., Futcher, B., Wimmer, E. & Mueller, S. (2008). Virus attenuation by genome-scale changes in codon pair bias. *Science* **320**, 1784-1787.

Edgar RC. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res* 2004;32(5):1792-1797.

Gutman, G. A. & Hatfield, G. W. (1989). Nonrandom utilization of codon pairs in Escherichia coli. *Proc Natl Acad Sci U S A* **86**, 3699-3703.

Mueller, S., Coleman, J. R., Papamichail, D., Ward, C. B., Nimnual, A., Futcher, B., Skiena, S. & Wimmer, E. (2010). Live attenuated influenza virus vaccines by computer-aided rational design. *Nat Biotechnol* **28**, 723-726.

Hopp, T. P. & Woods, K. R. (1983). A computer program for predicting protein antigenic determinants. *Mol. Immunol.* **20**, 483-489.

Kyte, J. & Doolittle, R. F. (1982). A simple method for displaying the hydropathic character of a protein. *J. Mol. Biol.* **157**, 105-132.

Li, W.-H. & Graur, D. (1991). Fundamentals of molecular evolution.

Rivas, E. & Eddy, S. R. (2000) . Secondary structure alone is generally not statistically significant for the detection of noncoding RNAs. *Bioinformatics* **16**, 583-605.

Robinson, D.F. & Foulds L.R. (1981). Comparison of phylogenetic trees. *Mathematical Biosciences* **53:** 131-147

Salminen M.O., Carr J.K., Burke D.S. & McCutchan FE (1995). Identification of breakpoints in intergenotypic recombinants of HIV type 1 by bootscanning. *AIDS Res Hum Retroviruses* **11**:1423-1425**.**

Sharp, P. M. & Li, W. H. (1986). An evolutionary perspective on synonymous codon usage in unicellular organisms. *J. Mol. Evol.* **24**, 28-38.

Simmonds, P. & Smith, D. B. (1999). Structural constraints on RNA virus evolution. *J. Virol.* **73**, 5787-5794.

Simmonds, P., Tuplin, A. & Evans, D. J. (2004). Detection of genome-scale ordered RNA structure (GORS) in genomes of positive-stranded RNA viruses: Implications for virus evolution and host persistence. *RNA* **10**, 1337-1351.

Tulloch, F., N. J. Atkinson, D. J. Evans, M. D. Ryan, and P. Simmonds. 2014. RNA virus attenuation by codon pair deoptimisation is an artefact of increases in CpG/UpA dinucleotide frequencies. eLife **3**: e04531.

Tuplin, A., Evans, D. J. & Simmonds, P. (2004). Detailed mapping of RNA secondary structures in core and NS5B coding region sequences of hepatitis C virus by RNAse cleavage and novel bioinformatic prediction methods. *J. Gen. Virol.* **85**, 3037-3047.

Tuplin, A., Wood, J., Evans, D. J., Patel, A. H. & Simmonds, P. (2002). Thermodynamic and phylogenetic prediction of RNA secondary structures in the coding region of hepatitis C virus. *RNA* **8**, 824-841.

Wang, T. H., Donaldson, Y. K., Brettle, R. P., Bell, J. E. & Simmonds, P. (2001). Identification of shared populations of human immunodeficiency virus type 1 infecting microglia and tissue macrophages outside the central nervous system. *J. Virol.* **75**, 11686-99.

Wright, F. (1990). The 'effective number of codons' used in a gene. *Gene* **87**, 23-29.

Yang, Z. (1999). Phylogenetic analysis by maximum likelihood (PAML) Version 2.0.

B) Citations: Please cite the following paper if referring to the sequence editor in a publication:

Simmonds, P. (2012). SSE: A nucleotide and amino acid sequence analysis platform. *BMC Research Notes.* **5**: 50. (http://www.biomedcentral.com/1756-0500/5/50)

The Association Index method was first published in the following paper:

Wang, T. H., Donaldson, Y. K., Brettle, R. P., Bell, J. E. & Simmonds, P. (2001). Identification of shared populations of human immunodeficiency virus type 1 infecting microglia and tissue macrophages outside the central nervous system. *J Virol* **75**, 11686-99.

The Grouping Scan method was first published in the following paper:

Simmonds, P. and S. Midgley. 2005. Recombination in the genesis and evolution of hepatitis B virus genotypes. *J. Virol.* **79**: 15467-15476.

Sequence scrambling methods were described in:

Simmonds, P., Tuplin, A. & Evans, D. J. (2004). Detection of genome-scale ordered RNA structure (GORS) in genomes of positive-stranded RNA viruses: Implications for virus evolution and host persistence. *RNA* **10**, 1337-1351.

Tuplin, A., Wood, J., Evans, D. J., Patel, A. H. & Simmonds, P. (2002). Thermodynamic and phylogenetic prediction of RNA secondary structures in the coding region of hepatitis C virus. *RNA* **8**, 824-841.

Two applications of the TreeOrder scan are first described in:

Simmonds, P. and S. Midgley. 2005. Recombination in the genesis and evolution of hepatitis B virus genotypes. J. *Virol*. **79**: 15467-15476.

Simmonds, P. and J. Welch. 2006. Frequency and dynamics of recombination within different species of human enteroviruses. *J. Virol.* **80**: 483-493.

The expanded Composition analysis program was used for large scale analysis of codon usage in:

Simmonds, P. 2006. Recombination and selections in the evolution of picornaviruses and other positive-stranded RNA viruses.. *J. Virol*. **80**: 11124-11140.

<BACK TO DOCUMENT INDEX>

CONDITIONS OF USE

Author:
Peter Simmonds
University of Oxford
Peter.Simmonds@ndm.ox.ac.uk

<BACK TO DOCUMENT INDEX>