COPYRIGHT NOTICE

The FASTA program package

Introduction

This documentation describes the version 2.0x of the FASTA program package (see W. R. Pearson and D. J. Lipman (1988), "Improved Tools for Biological Sequence Analysis", PNAS 85:2444-2448, and W. R. Pearson (1990) "Rapid and Sensitive Sequence Comparison with FASTP and FASTA" Methods in Enzymology 183:63-98). Version 2.0 modifies version 1.8 to include explicit statistical estimates for similarity scores based on the extreme value distribution. In addition, FASTA protein alignments now use the Smith-Waterman algorithm with no limitation on gap size. FASTA and SSEARCH now use the BLOSUM50 matrix by default, with options to change gap penalties on the command line. Version 1.7 replaces rdf2 and rss with prdf and prss, which use the extreme-value distribution to calculate accurate probability estimates.

Although there are a large number of programs in this package, they belong to four groups:

Library search programs: FASTA, FASTX, TFASTA, TFASTX, SSEARCH

Local homology programs: LFASTA, PLFASTA, LALIGN, PLALIGN, FLALIGN

Statistical significance: PRDF, RELATE, PRSS, RANDSEQ

Global alignment: ALIGN

In addition, I have included several programs for protein sequence analysis, including a Kyte-Doolittle hydropathicity plotting program (GREASE, TGREASE), and a secondary structure prediction package (GARNIER).

The FASTA sequence comparison programs on this disk are improved versions of the FASTP program, originally described in Science (Lipman and Pearson, (1985) Science 227:1435-1441). We have made several improvements. First, the library search programs use a more sensitive method for the initial comparison of two sequences which allows the scores of several similar regions to be combined. As a result, the results of a library search are now given with three scores, initn (the new initial score which may include several similar regions), init1 (the old fastp initial score from the best initial region), and opt (the old fastp optimized score allowing gaps in a 32 residue wide band).

These programs have also been modified to become "universal" (hence FAST-A, for FASTA-All, as opposed to FAST-P (protein) or FAST-N (nucleotides)); by changing the environment variable SMATRIX, the programs can be used to search protein sequences, DNA sequences, or whatever you like. By default, FASTA, LFASTA, and the PRDF programs automatically recognize protein and DNA sequences. Sequences are first read as amino acids, and then converted to nucleotides if the sequence is greater than 85% A,C,G,T (the '-n' option can be used to indicate DNA sequences). TFASTA compares protein sequences to a translated DNA sequence. Alternative scoring matrices can also be used. In addition to the BLOSUM50 matrix for proteins, the PAM250 matrix or matrices based on simple identities or the genetic code can also be used for sequence comparisons or evaluation of significance. Several different protein sequence matrices have been included; instructions for constructing your own scoring matrix are included in the file FORMAT.DOC.

The remainder of this document is divided into three sections: (1) a brief history of the changes to the FASTA package; (2) A guide to installing the programs and databases; (3) A guide to using the FASTA programs. The programs are very easy to use, so if you are using them on a machine that is administered by someone else, you may want to skip to section (3) to learn how to use the programs, and then read section (1) to look at some of the more recent changes. If you are installing the programs on your own machine, you will need to read section (2) carefully.

1. Revision History

1.1. Changes with version 2.0u

Version 2.0u provides several major improvements over previous versions of FASTA (and SSEARCH). The most important is the incorporation of explicit statistical estimates and appropriate normalization of similarity scores. This improvement is discussed in more detail below in the section entitled Statistical Significance. In addition, all of the protein comparison programs now use the BLOSUM50 matrix, with gap penalties of -12, -2, by default. BLOSUM50 performs significantly better than the older PAM250 matrix. PAM250 can still be used with the command line option: -s 250. (DNA sequence comparisons use a more stringent gap penalty of -16, -4, which produces excellent statistical estimates when optimized scores are used. TFASTA uses -16, -4 as well.)

The quality of the fit of the extreme value distribution to the actual distribution of similarity scores is summarized with the Kolmogorov-Smirnov statistic. The acceptance limits for this statistic can be found in many statistics books. In general, values <0.10 (N=30) indicate excellent agreement between the actual and theoretical distributions. If this statistic is > 0.2, consider using a higher (more stringent) gap penalty, e.g. -16, -4 rather than -12, -2. The default scoring matrix for DNA has been changed to score +5 for an identity and -4 for a mismatch. These are the same scores used by BLASTN.

With explicit expectation calculations, the program now shows all scores and alignments with expectations less than 10.0 (with optimized scores, 2.0 without optimization) when the "-Q" (quiet) mode is used. The expectation threshold can be changed with the "-E" option.

Finally, the algorithm used to produce the final alignments of protein sequences is now a full Smith-Waterman, with unlimited gaps. (The older band-limited alignments are used for DNA sequences and TFASTA by default, because Smith-Waterman alignments are very slow for long sequences.) Both the optimized and Smith-Waterman scores are reported; if the Smith-Waterman score is higher, then additional gaps allowed a better alignment and similarity score to be calculated.

FASTA searches now optimize similarity scores by default (this slows searches about 2-fold (worst case) for ktup=2). Thus, the meaning of the "-o" option has been reversed; "-o" now turns off optimization and reports results sorted by "initn" scores. Optimization significantly improves the sensitivity of FASTA, so that it almost matches Smith-Waterman. With version 2.0, the default band width used for optimized calculations can be varied with the "-y" option. For proteins with ktup=2, a width of 16 (-y 16) is used; 16 is also used for DNA sequences. For proteins and ktup=1, a width of 32 is used. Searches that disable optimization with the "-o" option will work fine for sequences that share 25% or more identity in general, but to detect evolutionary relationships with 20% - 25% identity, the more sensitive default optimization is often required. Optimization is required for accurate statistical estimates with either protein or DNA sequences.

The FASTA package now includes FASTX, a program that compares a DNA sequence to a protein sequence database by translating the DNA sequence in three frames (the reverse frames are selected with the -i option) and aligning the three-frame translation with the sequences in the protein database. Alignment scores allow frameshifts so that a cDNA or EST sequence with insertion/deletion errors can be aligned with its homologues from beginning to end.

With release 20u6, there is also a TFASTX program, which is a replacement for TFASTA. TFASTA treats each of the six reading frames of a DNA library sequence as a different sequence; TFASTX compares a protein sequence against only two sequences from each DNA sequence - the forward and reverse orientation. For a given orientation, TFASTX calculates a similarity score for alignments that allow frameshifts, thus considering all possible reading frames.

Another new program is included - randseq - which will produce a randomly shuffled (uniform or local shuffle) from an input sequence. This randomly shuffled sequence can be used to evaluate the statistical estimates produced by FASTA, SSEARCH, or BLAST.

1.2. Changes with version 1.7

Version 1.7 has been released to provide the PRDF and PRSS programs for shuffling sequences and estimating accurately the probabilities of the unshuffled-sequence scores.

PRDF a version of RDF2 that uses calculates the probability of a similarity score more accurately by using a fit to an extreme value distribution. Code to fit the extreme value distribution parameters and the impetus to update RDF2 was provided by Phil Green, U. of Washington.

PRSS a version of PRDF that uses a rigorous Smith-Waterman calculation to score similarities

1.3. Changes with version 1.6

FASTA version 1.6 uses a new method for calculating optimal scores in a band (the optimization or last step in the FASTA algorithm). In addition, it uses a linear-space method for calculating the actual alignments. FASTA v1.6 package includes several new programs:

SSEARCH a program to search a sequence database using the rigorous Smith-Waterman algorithm (this program is about 100-fold slower than FASTA with ktup=2 (for proteins).

LALIGN A rigorous local sequence alignment program that will display the N-best local alignments (N=10 by default).

PLALIGN a version of lalign that plots the local alignments to a tektronix display.

FLALIGN a version of lalign that plots the local alignments to a GCG Figure file.

The LALIGN/PLALIGN/FLALIGN programs incorporate the "sim" algorithm described by Huang and Miller (1991) Adv. Appl. Math. 12:337-357. The SSEARCH and PRSS programs incorporate algorithms described by Huang, Hardison, and Miller (1990) CABIOS 6:373-381.

LFASTA and PLFASTA now calculate a different number of local similarities; they now behave more like LALIGN/PLALIGN. Since local alignments of identical sequences produce "mirror-image" alignments, lalign and lfasta consider only one-half of the potential alignments between sequences from identical file names. Thus

```
    lfasta mchu.aa mchu.aa
```

Displays only two alignments, with earlier versions of the program, it would have displayed five, including the identity alignment. PLFASTA does display five alignments; when two identical filenames are given, it draws the identity alignment, calculates the two unique local alignments, draws them, and draws their mirror images. LFASTA/PLFASTA and LALIGN/PLALIGN use the filenames, rather than the actual sequences, to determine whether sequences are identical; you can "trick" the programs into behaving the old way by putting the same sequence in two different files.

1.4. Changes with version 1.5

FASTA version 1.5 includes a number of substantial revisions to improve the performance and sensitivity of the program. It is now possible to tell the program to optimize all of the initn scores greater than a threshold. The threshold is set at the same value as the old FASTA cutoff score. Alternatively, you can tell FASTA to sort the results by the init1, rather than the initn, score by using the -1 option. FASTA -1 ... will report the results the way the older FASTP program did.

A new method has been provided for selecting libraries. In the past, one could enter the name of a sequence file to be searched or a single letter that would specify a library from the list included in the $FASTLIBS file. Now, you can specify a set of library files with a string of letters preceded by a '%'. Thus, if the FASTLIBS file has the lines:

```
    Genbank 70 primates$1P/seqlib/gbpri.seq 1
    Genbank 70 rodents$1R/seqlib/gbrod.seq 1
    Genbank 70 other mammals$1M/seqlib/gbmam.seq 1
    Genbank 70 vertebrates $1B/seqlib/gbvrt.seq 1
```

Then the string: "%PRMB" would tell FASTA to search the four libraries listed above. The %PRMB string can be entered either on the command line or when the program asks for a filename or library letter.

FASTA1.5 also provides additional flexibility for specifying the number of results and alignments to be displayed with the -Q (quiet) option. The -b number option allows you to specify the number of sequence scores to show when the search is finished. Thus

```
    FASTA -b 100 ...
```

tells the program to display the top 100 sequence scores. In the past, if you displayed 100 scores (in -Q mode), you would also have store 100 alignments. The -d option allows you to limit the number of alignments shown. FASTA -b 100 -d 20 would show 100 scores and 20 alignments.

Finally, FASTA can provide a complete list of all of the sequences and scores calculated to a file with the -r (results) option. FASTA -r results.out ... creates a file with a list of scores for every sequence in the library. The list is not sorted, and only includes those scores calculated during the initial scan of the library.

2. Installing the FASTA package

2.1. Installing the programs

2.1.1. Unix version

The FASTA distribution comes with several makefile's that can be used to compile the FASTA programs. Over the years, as ATT Unix System 5 and BSD unix have converged, these files have become very similar. To begin with, I recommend using the standard Makefile. There are two values in the makefile that should be checked against the values used on your system: the HZ value, which is the frequency in ticks per second used by the times() system call, this value can usually be found by running:

```
    grep HZ /usr/include/sys/*
```

and the functions available to return random numbers. If you have a rand48() function that returns a 32-bit random number, use it and use the lines:

```
    NRAND=nrand48
    RANFLG= -DRAND32
```

If not, you will need to use the rand() function call and determine whether it returns a 16-bit or a 32-bit value. These functions are used by PRDF and PRSS. If you have problems compiling the programs, you may want to examine the makefile.unx and makefile.sun files, to look for differences. I have tried to use very standard unix functions in these programs, and they have been successfully compiled, with very small changes to the Makefile, on Sun's (Sun OS 4.1), IBM RS/6000's (AIX), and MIPS machines (under the BSD environment).

2.1.2. IBM-PC/DOS version

For the IBM-PC/DOS version, the FASTA source code disk contains the complete source code to all of the programs on the other disks. The programs were compiled with Borland's Turbo 'C++', using Borland's MAKE utility. The graphics programs (PLFASTA, TGREASE) use the graphics device drivers supplied with the Turbo 'C' V2.0 package. Also included are the documentation files PROGRAMS.DOC and FORMAT.DOC. You do not need any of the files the source code disk to run the programs. The files on this disk are identical to the UNIX and VMS versions that run on larger machines. Also included is the code to compile ALIGN0.EXE. ALIGN0 is the same as ALIGN, but does not penalize for end-gaps.

If you have the DOS or Macintosh version of the FASTA package, to install the programs you should:

(1) Make a new directory (folder) for the FASTA programs. This need not be the same as the directory for your sequence databases.

(2) Copy the files from the FASTA source disk to the new directory.

(3) (DOS only) Edit your AUTOEXEC.BAT file to (a) modify your PATH command to include the FASTA directory and (b) add the line:

```
set FASTLIBS=c:\yourfastadirectory\fastgbs
```

On the Macintosh, you may need to edit the "environment" file and change the line that reads:

```
FASTLIBS=fastgbs
```

to indicate the full directory path for the fastgbs file, for example:

```
FASTLIBS=Q105:FASTA:fastgbs
```

(4) Finally, you will need to edit the fastgbs file. This is usually the most confusing part of the installation. An example of this file is shown below; to customize this file for your machine, you will need to change the file names from those provided in the fastgbs file to ones that reflect the directory names and file names you use on your machine. This is explained in more detail below. In addition, some entries in the fastgbs file refer to other files of file names. These files of file names (as opposed to actual database files) may also need to be edited.

2.2. Installing the libraries

2.2.1. The NBRF protein sequence library

The FASTA program package does not include any protein or DNA sequence libraries. You can obtain the PIR protein sequence database from:

National Biomedical Research Foundation
Georgetown University Medical Center
3900 Reservoir Rd, N.W.
Washington, D.C. 20007

In addition, this database is available via anonymous ftp from the host "ftp.bchs.uh.edu". It is available in two formats, VMS and CODATA format. The "VMS" format (library type 5 below) can be searched much faster, can be easily reformatted for use by the "BLAST" rapid searching program, and is compatible with the Genetics Computer Group package of programs. The CODATA format is used by the EUGENE/MBIR computing package from Baylor (library type 2).

2.2.2. The GENBANK DNA sequence library

FASTA, and TFASTA search sequences from the GENBANK "flatfile" (not ASN.1) DNA sequence library in the flat-file format distributed by the National Center for Biotechnology Information and the PIR format used by EBI/EMBL. CD-ROMs can be obtained from:

Genbank
National Center for Biotechnology Information
National Library of Medicine

National Institutes of Health
8600 Rockville Pike
Bethesda, MD 20894

The GenBank DNA sequence library is also available via anonymous FTP from ncbi.nlm.nih.gov.

2.2.3. The EBI/EMBL CD-ROM libraries

The European Bioinformatics Institute (EBI) is now distributing the EMBL CD-ROM that contains both the complete EMBL DNA sequence database (which should be essentially identical to the GenBank DNA sequence database) and the SWISS-PROT protein sequence database. SWISS-PROT is derived from the NBRF Protein sequence database with additions from the EBI/EMBL DNA sequence database. This CD-ROM is a "best-buy," since it provides both DNA and protein sequence libraries. It is available from:

European Bioinformatics Institute
Hinxton Genome Campus, Hinxton Hall
Hinxton, Cambridge CB10 1RQ,
United Kingdom
Tel: +44 1223 4944
Fax: +44 1223 494468
Email: DATALIB@ebi.ac.uk

In addition, the SWISS-PROT protein sequence database is available via anonymous FTP from ncbi.nlm.nih.gov.

2.3. Finding the libraries: FASTLIBS

FASTA and TFASTA use the environment variable FASTLIBS to find the protein and DNA sequence libraries. The FASTLIBS variable contains the name of a file that has the actual filenames of the libraries. The FASTGBS file on is an example of a file that can be referred to by FASTLIBS. To use the FASTGBS file, type:

```
    setenv FASTLIBS /usr/lib/fasta/fastgbs (BSD UNIX/csh)
    or
    export FASTLIBS=/usr/lib/fasta/fastgbs (SysV UNIX/ksh)
```

Then edit the FASTGBS file to indicate where the protein and DNA sequence libraries can be found. If you have a hard disk and your protein sequence library is kept in the file /usr/lib/aabank.lib and your Genbank DNA sequence library is kept in the directory: /usr/lib/genbank, then fastgbs might contain:

```
    NBRF Protein$0P/usr/lib/seq/aabank.lib 0
    SWISS PROT 10$0S/usr/lib/vmspir/swiss.seq 5
    GB Primate$1P@/usr/lib/genbank/gpri.nam
    GB Rodent$1R@/usr/lib/genbank/grod.nam
    GB Mammal$1M@/usr/lib/genbank/gmammal.nam
    ^   1    ^^^^       4                   ^     ^
              23                             (5)
```

The first line of this file says that there is a copy of the NBRF protein sequence database (which is a protein database) that can be selected by typing "P" on the command line or when the database menu is presented in the file /usr/lib/seq/aabank.lib.

Note that there are 4 or 5 fields in the lines in fastgbs. The first field is the description of the library which will be displayed by FASTA; it ends with a '$'. The second field (1 character), is a 0 if the library is a protein library and 1 if it is a DNA library. The third field (1 character) is the character to be typed to select the library.

The fourth field is the name of the library file. In the example above, the /usr/lib/seq/aabank.lib file contains the entire protein sequence library. However the DNA library file names are preceded by a '@', because these files (gpri.nam, grod.nam, gmammal.nam) do not contain the sequences; instead they contain the names of the files which contain the sequences. This is done because the GENBANK DNA database is broken down in to a large number of smaller files. In order to search the entire primate database, you must search more than a dozen files.

In addition, an optional fifth field can be used to specify the format of the library file. Alternatively, you can specify the library format in a file of file names (a file preceded by an '@'). This field must be separated from the file name by a space character (' ') from the filename. In the example above, the aabank.lib file is in Pearson/FASTA format, while the swiss.seq file is in PIR/VMS format (from the EMBL CD-ROM). Currently, FASTA can read the following formats:

0 Pearson/FASTA (>SEQID - comment/sequence)
1 Uncompressed Genbank (LOCUS/DEFINITION/ORIGIN)
2 NBRF CODATA (ENTRY/SEQUENCE)
3 EMBL/SWISS-PROT (ID/DE/SQ)
4 Intelligenetics (;comment/SEQID/sequence)
5 NBRF/PIR VMS (>P1;SEQID/comment/sequence)
6 GCG (version 8.0) Unix Protein and DNA (compressed)
11 NCBI Blast1.3.2 format (unix only)

In particular, this version will work with the EMBL and PIR VMS formats that are distributed on the EMBL CD-ROM. The latter format (PIR VMS) is much faster to search than EMBL format. This release also works with the protein and DNA database formats created for the BLASTP and BLASTN programs by SETDB and PRESSDB and with the new NCBI search format. If a library format is not specified, for example, because you are just comparing two sequences, Pearson/FASTA (format 0) is used by default. To change this default, you may set the LIBTYPE environment variable to a number. For example,

```
setenv LIBTYPE 1
```

would cause the program to use the GenBank LOCUS format by default for libraries (or the second sequence file), but the Pearson/FASTA format would still be used for the query sequence.

You can specify a group of library files by putting a '@'

symbol before a file that contains a list of file names to be searched. For example, if @gmam.nam is in the fastgbs file, the file "gmam.nam" might contain the lines:

```
    </usr/lib/genbank
    gbpri.seq 1
    gbrod.seq 1
    gbmam.seq 1
```

In this case, the line beginning with a '<' indicates the directory the files will be found in. The remaining lines name the actual sequence files. So the first sequence file to be searched would be:

```
    /usr/lib/genbank/gbpri.seq
```

The notation "<PIRNAQ:" might be used under the VAX/VMS operating system. Under UNIX, the trailing '/' is left off, so the library directory might be written as "</usr/seqlib".

With version 1.4 of the FASTA package, the FASTA and TFASTA programs can search a library composed of different files in different sequence formats. For example, you may wish to search the Genbank files (in GenBank flat file format) and the EMBL DNA sequence database on CD-ROM. To do this, you simply list the names and filetypes of the files to be searched in a file of filenames. For example, to search the mammalian portion of Genbank, the unannotated portion of Genbank, and the unannotated portion of the EMBL library, you could use the file:

```
    </usr/lib/DNA
    gbpri.seq 1
    #  (this '#' causes the program to display the size of the library)
    gbrod.seq 1
    gbmam.seq 1
    gbuna.seq 1
    unanno.seq 5
    #
```

You do not need to include library format numbers if you only use the Pearson/FASTA version of the PIR protein sequence library. If no library type is specified, the program assumes that type 0 is being used (unless you have set LIBTYPE).

Support for the old compressed GenBank files, which have not been distributed for more than four years, has been removed from programs in the FASTA package.

Test the setup by running FASTA. Enter the sequence file 'MUSPLFM.AA' when the program requests it (this file is included with the programs). The program should then ask you to select a protein sequence library. Alternatively, if you run the TFASTA program and use the MUSPLFM.AA query sequence, the program should show you a selection of DNA sequence libraries. Once the fastgbs file has been set up correctly, you can set FASTLIBS=fastgbs in your AUTOEXEC.BAT file, and you will not need to remember where the libraries are kept or how they are named.

FASTA and TFASTA must open a large number of files when searching and reporting the results of a GENBANK floppy disk format library search. You may have problems with the large number of files under DOS on IBM-PC's (Unix and VMS users will not have these problems). If you are going to search the GENBANK floppy disk format DNA sequence library under DOS, you should add the line:

```
    FILES=16
```

to your CONFIG.SYS file. (Typically this is already done for programs like Windows or WordPerfect.)

3. Using the FASTA Package

3.1. Overview

The FASTA sequence comparison programs all require similar information, the name of a query sequence file, a library file, and the ktup parameter. All of the programs can accept arguments on the command line, or they will prompt for the file names and ktup value.

To use FASTA, simply type:

```
    FASTA
    and you will be prompted for :
         the name of the test sequence file
         the name of the library file
         and whether you want ktup = 1 or 2. (or 1 to 6 for DNA sequences)

             ktup of 2 is about 5 times faster than ktup = 1.
             For  a  200  aa sequence against a 10,000,000 aa
             library, the program takes  about  30  min  with
             ktup = 2, 150 min with ktup = 1, on a 12 Mhz 286
             IBM-PC.
```

The program can also be run by typing

```
    FASTA test.aa /lib/bigfile.lib ktup (1 or 2)
```

Included with the package are the test files, MUSPLFM.AA, LCBO.AA, MCHU.AA and BOVPRL.SEQ. To check to make certain that everything is working, you can try:

```
    fasta musplfm.aa lcbo.aa
    and
    tfasta musplfm.aa bovprl.seq
```

To test the local similarity programs LFASTA and PLFASTA, try:

```
    lfasta mchu.aa mchu.aa
    and
    plfasta mchu.aa mchu.aa (use this only on an IBM-PC with graphics
    or on a Tektronix terminal under UNIX or VMS)
```

MCHU (calmodulin) has four duplicated calcium binding sites that are clearly detected by LFASTA. For a more complicated example, try MWRTC1.aa, myosin heavy chain.

3.2. Sequence files

The FASTA programs know about three kinds of sequence files (four under VMS): (1) plain sequence files that can only be used as query sequences or for LFASTA, PRDF, and ALIGN. (2) Standard library files. These are the same as plain sequence files, each sequence is preceded by a comment line with a '>' in the first column. (3) distributed sequence libraries (this is a broad class that includes the NBRF/PIR VMS and blocked ascii formats, Genbank flat-file format, EMBL flat-file format, and Intelligenetics format. All of the files that you create should be of type (1) or (2). Type (2) files (ones with a be used as query or library sequence files by all of the programs.

I have included several sample test files, *.AA. The first line may begin with a '>' or ';' followed by a comment. The text after ';' in other lines will be ignored. Spaces and tabs (and anything else that is not an amino-acid code) are ignored.

Library files should have the form:

```
>Sequence name and identifier
A F A S Y T .... actual sequence.
F S S       .... second line of sequence.
>Next sequence name and identifier
```

This is often referred to as "FASTA" or "Pearson" format. You can build your own library by concatenating several sequence files. Just be sure that each sequence is preceded by a line beginning with a '>' with a sequence name.

The test file should not have lines longer than 120 characters, and sequences entered with word processors should use a document mode, with normal carriage returns at the end of lines.

Program Summary

3.3. Sequence search programs

FASTA universal sequence comparison. Defaults to comparing protein sequences; if the sequences are > 85% A+C+G+T or the -n option is used, a DNA sequence is assumed.

FASTX Search a protein sequence library using amino acid sequence comparison to the forward three frames of a translated DNA query sequence. (The reverse frames are specified with the -i option.) Alignment scores allow frameshifts; the final alignment uses a Smith-Waterman type alignment routine (no limit on gaps) that allows frameshifts.

TFASTA Search DNA library for a protein sequence by translating the DNA sequence to protein in all six frames (three forward frames with the -3 command line option). TFASTA with ktup=2 is about as fast as a DNA FASTA with ktup=4, and is substantially more sensitive. (also reads the GENBANK library)

TFASTX Search DNA library for a protein sequence by translating the DNA sequence to protein in all six frames (three forward frames with the -3 command line option) calculating similarity scores that allow frameshifts. TFASTX produces an optimal Smith-Waterman alignment of the query and translated-library sequence.

SSEARCH Universal sequence comparison using the Smith-Waterman algorithm (T. F. Smith and M. S. Waterman (1981) J. Mol. Biol. 147:195-197). This program uses code developed by Huang and Miller (X. Huang, R. C. Hardison, W. Miller (1990) CABIOS 6:373-381) for calculating the local similarity score and code from the ALIGN program (see below) for calculating the local alignment. SSEARCH is about 50-times slower than FASTA with ktup=2 (for proteins).

ALIGN optimal global alignment of two sequences with no short-cuts. This program is a slightly modified version of one taken from E. Myers and W. Miller. The algorithm is described in E. Myers and W. Miller, "Optimal Alignments in Linear Space" (CABIOS (1988) 4:11-17).

3.4. Local similarity programs

LFASTA local similarity searches showing local alignments. The algorithm used to calculate the local alignment in a band has been improved (Chao, Pearson, and Miller, submitted).

PLFASTA local similarity searches with plot output (on the IBM, this program requires that the environment variable BGIDIR be set).

PCLFASTA (unix only) local similarity searches with plot output using pic commands.

LALIGN Calculates the N-best local alignments using a rigorous algorithm. (N=10 by default.) The algorithm was developed by Huang and Miller (X. Huang and W. Miller (1991) Adv. Appl. Math. 12:337-357), which is a linear-space version of an algorithm described by M. S. Waterman and M. Eggert (J. Mol. Biol. 197:723-728). Like SSEARCH, LALIGN is rigorous, but also very slow.

PLALIGN A version of LALIGN that plots its output to a screen or to a Tektronix terminal emulator.

3.5. Statistical Significance

With version 2.0 of the FASTA program distribution, FASTA, TFASTA, and SSEARCH now provide estimates of statistical significance for library searches. Work by Altschul, Arratia, Karlin, Mott, Waterman, and others (see Altschul et al. (1994) Nature Genetics 6:119 for an excellent review) suggests that local sequence similarity scores follow the extreme value distribution, so that P(s > x) = 1 - exp(-exp(-lambda(x-u)) where u = ln(Kmn)/lambda and m,m are the lengths of the query and library sequence. This formula can be rewritten as: 1 - exp(-Kmn exp(-lambda x), which shows that the average score for an unrelated library sequence increases with the logarithm of the length of the library sequence. FASTA and SSEARCH use simple linear regression against the the log of the library sequence length to calculate a normalized "z-score" with mean 50, regardless of library sequence length, and variance 10. These z-scores can then be used with the extreme value distribution and the poisson distribution (to account for the fact that each library sequence comparison is an independent test) to calculate the number of library sequences to obtain a score greater than or equal to the score obtained in the search. The original idea and routines to do the linear regression on library sequence length were provided Phil Green, U. Washington. This version of FASTA and SSEARCH uses a slightly different strategy for fitting the data than those originally provided by Dr. Green.

The expected number of sequences is plotted in the histogram using an "*". Since the parameters for the extreme value distribution are not calculated directly from the distribution of similarity scores, the pattern of "*'s" in the histogram gives a qualitative view of how well the statistical theory fits the similarity scores calculated by FASTA and SSEARCH. For FASTA, if optimized scores are calculated for each sequence in the database (the default), the agreement between the actual distribution of "z-scores" and the expected distribution based on the length dependence of the score and the extreme value distribution is usually very good. Likewise, the distribution of SSEARCH Smith-Waterman scores typically agrees closely with the actual distribution of "z-scores." The agreement with unoptimized scores, ktup=2, is often not very good, with too many high scoring sequences and too few low scoring sequences compared with the predicted relationship between sequence length and similarity score. In those cases, the expectation values may be overestimates.

The statistical routines assume that the library contains a large sample of unrelated sequences. If this is not the case, then the expectation values are meaningless. Likewise, if there are fewer than 20 sequences in the library, the statistical calculations are not done.

For protein searches, library sequences with E() values < 0.01 for searches of a 10,000 entry protein database are almost always homologous. Frequently sequences with E()-values from 1 - 10 are related as well. Remember, however, that these E() values also reflect differences between the amino acid composition of the query sequence and that of the "average" library sequence. Thus, when searches are done with query sequences with "biased" amino-acid composition, unrelated sequences may have "significant" scores because of sequence bias. The programs below, PRDF and PRSS, can address this problem by calculating similarity scores for random sequences with the same length and amino acid composition.

If optimization is not used ("-o"), E-values for DNA sequences overestimate the significance of the scores that are obtained and unrelated sequences frequently have E()-values < 0.0005. With optimization, the agreement between E()-value compares favorably with protein sequence comparison. This is in part due to the use of more stringent gap penalties for DNA sequence comparison, -16, -4 rather than -12, -2. With the latter penalties, many unrelated sequences appear to have significant similarity. Nevertheless, since protein sequence comparison is much more sensitive, DNA sequence comparison should not be used to identify sequences that encode protein. Even with ktup=6, optimization rarely increases run-times more than 50% with mRNA-size query sequences. Optimization should be used whenever possible.

Similar comments apply to TFASTA, where higher gap penalties (-16,-4) are required for accurate statistical estimates. Because TFASTA produces so many artificial "coding" sequences with atypical amino acid compositions, the statistical estimates with TFASTA are often over estimates. With optimized scores, ktup=1, and gap penalties of -16, -4, unrelated sequences will sometimes have E() values of 0.1. If initn scores are used, unrelated sequences may have have E() values < 0.01.

PRDF improved version of RDF program that includes accurate probability estimates for all three scoring methods (includes local or window shuffle routine)

PRSS A version of PRDF that uses the rigorous Smith-Waterman calculation used by SSEARCH.

RANDSEQ produces a randomly shuffled sequence from a query sequence.

RELATE significance program described by Dayhoff (Atlas of Protein Sequence and Structure, Vol. 5, Supplement 3). Each chunk of 25 residues in one sequence is compared to every 25 residue fragment of the second sequence. Sequences which are genuinely related will have a large number of scores greater than 3 standard deviations above the mean score of all of the comparisons.

3.6. Other analysis programs

AACOMP calculate the amino acid composition and molecular weight of a sequence.

BESTSCOR calculate the best self-comparison score.

GREASE Kyte-Doolittle hydropathicity profile

TGREASE graphic plot of Kyte-Doolittle profile

FROMGB convert from GenBank LOCUS format (also used by the IBI-Pustell programs) to Pearson/FASTA format.

GARNIER A secondary structure prediction program using the method of Garnier, Osgusthorpe, and Robson, J. Mol.

Biol., (1978) 120:97-120.

3.7. Options

These programs have a number of output options, which are invoked by the environment variables LINLEN, SHOWALL, and MARKX. Alternatively, these values can be controlled by command line options. The number of sequence residues per output line is now adjustable by setting the environment variable LINLEN, or the command line option -w. LINLEN is normally 60, to change it set LINLEN=80 before running the program or add -w 80 to the command line. LINLEN can be set up to 200. SHOWALL (-a) determines whether all, or just a portion, of the aligned sequences are displayed. Previously, FASTP would show the entire length of both sequences in an alignment while FASTN would only show the portions of the two sequences that overlapped. Now the default is to show only the overlap between the two sequences, to show complete sequences, set SHOWALL=1, or use the -a option on the command line.

The differences between the two aligned sequences can be highlighted in three different ways by changing the environment variable MARKX or the -m option. Normally (MARKX=0) the program uses ':' do denote identities and '.' to denote conservative replacements. If MARKX=1, the program will not mark identities; instead conservative replacements are denoted by a 'x' and non-conservative substitutions by a 'X'. If MARKX=2, the residues in the second sequence are only shown if they are different from the first. MARKX=3 displays the aligned library sequences without the query sequence; these can be used to build a primitive multiple alignment. MARKX=4 provides a graphical display of the boundaries of the alignments. Thus the five options are:

```
     MARKX=0      MARKX=1       MARKX=2       MARKX=3      MARKX=4

    MWRTCGPPYT   MWRTCGPPYT    MWRTCGPPYT                 MWRTCGPPYT
    ::..:: :::     xx  X       ..KS..Y...    MWKSCGYPYT   ----------
    MWKSCGYPYT   MWKSCGYPYT
```

(fasta20u4, Feb. 1996) In addition MARKX=10 is a new, parseable format for use with other programs. See the file"readme.v20u4" for a more complete description.

3.8. Command line options

It is now possible to specify several options on the command line, instead of using environment variables. The command line options are preceded by a dash; the following options are available:

-a   same as showall=1

-A   force Smith-Waterman alignments for DNA sequences and TFASA. By default, only FASTA protein sequence comparisons use Smith-Waterman alignments.

-b #   Number of sequence scores to be shown on output. In the absence of this option, fasta (and tfasta and ssearch) display all library sequences obtaining similarity scores with expectations less than 10.0 if optimized score are used, or 2.0 if they are not. The -b option can limit the display further, but it will not cause additional sequences to be displayed.

-c # Threshold score for optimization (OPTCUT). Set "-c 1" to optimize every sequence in a database. (This slows the program down about 5-fold).

-E # Limit the number of scores and alignments shown based on the expected number of scores. Used to override the expectation value of 10.0 used by default. When used with -Q, -E 2.0 will show all library sequences with scores with an expectation value <= 2.0.

-d # Number of alignments to be reported by default. (Used in conjunction with -Q). No longer necessary, see "-b" above.

-f Penalty for the first residue in a gap (-12 by default for proteins, -16 for DNA or for TFASTA).

-g Penalty for additional residues in a gap (-2 by default for proteins, -4 for DNA and TFASTA).

-h Penalty for frameshift (FASTX, TFASTX only).

-H Omit histogram.

-i Invert (reverse complement) the query sequence if it is DNA. For TFASTX, search the reverse complement of the library sequence only.

-k # Threshold for joining init1 segments to build an initn score (GAPCUT).

-l file Location of library menu file (FASTLIBS).

-L Display more information about the library sequence in the alignment.

-m # MARKX = # (0, 1, 2, 3, 4, 10)

-n Force the query sequence to be treated as a DNA sequence. This is particularly useful for query sequences that contain a large number of ambiguous residues, e.g. transcription factor binding sites.

-O Send copy of results to "filename." Helpful for environments without STDOUT.

-o Turn off default optimization of all scores greater than OPTCUT. Sort results by "initn" scores.

-Q,-q Quiet - does not prompt for any input. Writes scores and alignments to the terminal or standard output file.

-r file   Save a results summary line for every sequence in the sequence library. The summary line includes the sequence identifier, superfamily number (if available) position in the library, and the similarity scores calculated. This option can be used to evaluate the sensitivity and selectivity of different search strategies (see W. R. Pearson (1991) Genomics 11:635-650.)

-s file   SMATRIX is read from file. Several SMATRIX files are provided with the standard distribution. For protein sequences: codaa.mat - based on minimum mutation matrix; idnaa.mat - identity matrix; pam250.mat - the PAM250 matrix developed by Dayhoff et al (Atlas of Protein Sequence and Structure, vol. 5, suppl. 3, 1978); pam120.mat - a PAM120 matrix. The default scoring matrix is BLOSUM50, PAM250 is available with "-s 250", BLOSUM62 ("-s BL62") is also available.

-v   (LINEVAL) values used for line styles in plfasta

-w #   Line length (width) = number (<200)

-x   Specifies offsets for the beginning of the query and library sequence. For example, if you are comparing upstream regions for two genes, and the first sequence contains 500 nt of upstream sequence while the second contains 300 nt of upstream sequence, you might try:

```
    fasta -x "-500 -300" seq1.nt seq2.nt
```

If the -x option is not used, FASTA assumes numbering starts with 1. This option will not work properly with the translated library sequence with tfasta. (You should double check to be certain the negative numbering works properly.)

-y   Set the width of the band used for calculating "optimized" scores. For proteins and ktup=2, the width is 16. For proteins with ktup=1, the width is 32 by default. For DNA the width is 16.

-z   Turn off statistical calculations.

-1   sort output by init1 score (as FASTP used to do).

-3   (TFASTA, TFASTX only) translate only three forward frames

For example:

```
    fasta -w 80 -a seq1.aa seq.aa
```

would compare the sequence in seq1.aa to that in seq2.aa and display the results with 80 residues on an output line, showing all of the residues in both sequences. Be sure to enter the options before entering the file names, or just enter the options on the command line, and the program will prompt for the file names.

Not all of these options are appropriate for all of the programs. The options above are used by FASTA and TFASTA. RELATE uses the -s option, ALIGN uses the -w, -m, and -s options, and the PRDF program uses -c, -f, -k, and -s.

4. Environment variable summary

Environment variables allow you to set search parameters that will be used frequently when you run a program; for example, if you prefer to use the PAM250 scoring matrix, you might "set SMATRIX=250." Command line parameters, if used, always override environment variable settings. The following environment variables are used by this program:

AABANK the file name of the default sequence library.

FASTLIBS the location of the file which contains the list of library files to be searched.

GAPCUT threshold used for joining init1 regions in the second step of FASTA. Normally set based on sequence length and ktup.

LIBTYPE used to specify the format of the library sequence for FASTA and TFASTA.

LINLEN output line length - can go up to 200

LINEVAL used by plfasta to determine the relationship between line style and similarity score (-v). This should be a string of three numbers, e.g. "200 100 50"

MARKX symbol for denoting matches, mismatches. Note that this symbol is only used across the optimized local region; sequences that are outside this region are not marked.

OPTCUT Set the threshold to be used for optimization in a band around the best initial region. Normally the OPTCUT value is calculated from the length of the sequence and the ktup value (for a 200 residue sequence, it is about 28). If OPTCUT=1, every sequence in the database will be optimized. This is the most sensitive option.

PAMFACT This version of fasta uses a more sensitive method for identifying initial regions. Instead of using a constant factor (fact) for each match in a ktup, it uses the scoring matrix (PAM) scores. While this works well for protein sequences, it has not been as carefully tested for DNA sequences, so by default, this modification is used for proteins but not for DNA. Setting the PAMFACT environment variable to 1 forces the option on; PAMFACT=0 turns it off.

SHOWALL on output, show the complete sequence instead of just the overlap of the two aligned sequences.

SMATRIX alternative scoring matrix file.

TEKPLOT (IBM-PC only, Unix and VMS versions generate Tektronix graphics by default) Generate Tektronix output. Normally, PLFASTA and TGREASE plot graphs using the Turbo C graphics library. Unfortunately, often these plots cannot be printed out without special programs. However, if you set TEKPLOT=1, tektronix graphics commands will be used. Tektronix commands can be used together with the PLOTDEV program, available from Microplot Systems. They no lonter sell this program, but it can be downloaded from http://iquest.com/~microplt/index1.html. PLOTDEV also allows you to print out graphics on the screen.

As always, please inform me of bugs as soon as possible.

William R. Pearson
Department of Biochemistry
Box 440, Jordan Hall
U. of Virginia
Charlottesville, VA

wrp@virginia.EDU