

# Modern BLAST Programs

Jian Ma and Louxin Zhang

**Abstract** The Basic Local Alignment Search Tool (BLAST) is arguably the most widely used program in bioinformatics. By sacrificing sensitivity for speed, it makes sequence comparison practical on huge sequence databases currently available. The original version of BLAST was developed in 1990. Since then it has spawned a variant of specialized programs. This chapter surveys the development of BLAST and BLAST-like programs for homology search, alignment statistics that are used in assessment of reported matches in BLAST, and provides the reader with guidance to select appropriate programs and set proper parameters to match research requirements.

## 1 Introduction

The sequence structures of genes and proteins are conserved in nature. It is common to observe strong sequence similarity between a protein and its counterpart in another species that diverged hundreds of millions of years ago. Accordingly, the best method to identify the function of a new gene or protein is to find its sequence-related genes or proteins whose functions are already known.

The Basic Local Alignment Search Tool (BLAST) is a computer program for finding regions of local similarity between two DNA or protein sequences. It is designed for comparing a query sequence against a target database. It is a heuristic that finds short matches between query and database sequences and then attempts to start alignments from these 'seed hits'. By sacrificing sensitivity for speed, it makes sequence comparison practical on huge sequence databases currently available such as GenBank, which has over 80 million sequence records as of August 2008. In

---

Jian Ma  
University of California at Santa Cruz, e-mail: jianma@soe.ucsc.edu

Louxin Zhang  
National University of Singapore, e-mail: matzlx@nus.edu.sg

addition to generating local alignments, BLAST also provides statistical assessment of reported alignments. Because of these sensational features, BLAST becomes one of the most widely used bioinformatics tools.

BLAST analysis is often used to identify conserved sequence patterns and to establish functional or evolutionary relationship among proteins. It finds numerous applications in molecular biology, evolutionary biology and drug discovery.

The original version of BLAST [6] was developed by Altschul, Gish, Lipman, Miller and Myers in 1990. The improved version PSI-BLAST [7] was made available in 1997. Over the past 28 years, the original version has been customized into a set of specialized programs. These new variants of BLAST handle homology search on different types of databases. They were designed to find gapped local alignments and to detect weak signals in sequence alignment. Table 3 lists the popular BLAST programs together with their functions.

The rest of this chapter is divided into seven sections. Section 2 describes the available BLAST programs and other BLAST-like programs. Sections 3 and 4 present the algorithmic and statistical aspects of BLAST, respectively. Section 5 describes two practical examples of using BLAST. Through these examples, we examine the biological and statistical information output from BLAST. Section 6 addresses three advanced issues of using BLAST homology search. Section 7 lists some exercises for the reader to master BLAST programs. Finally, we summarize the most relevant and useful references on BLAST for further reading in Section 8.

## 2 Available Implementations

From a user point of view, based on different purposes, a BLAST search generally involves three important parts: input, database searched against, and a particular BLAST program.

On NCBI BLAST website, the available databases can be categorized into protein databases and nucleotide databases. Frequently used databases are summarized in Table 1 and Table 2. In addition, NCBI also provides specialized BLAST databases, e.g. genome databases for different species, trace databases, as well as various databases for model organisms.

**Table 1** Main protein sequence databases for BLAST

Database	Description
nr	Non-redundant collections from GenBank CDS translations, PDB, SwissProt, PIR, and PRF
month	The nr updates in the last 30 days
refseq	Protein sequences from RefSeq project
swissprot	SWISS-PROT protein sequence database
pdb	Sequences from the 3-dimensional structure records in PDB

**Table 2** Main nucleotide sequence databases for BLAST

Database	Description
nr	All sequences in GenBank, EMBL, DDBJ, PDB except EST, STS, GSS, etc.
month	The nr updates in the last 30 days
refseq_mrna	mRNA sequences from RefSeq Project
refseq_genomic	Genomic sequences from RefSeq Project
est	Sequences in GenBank, EMBL, and DDBJ from EST division
gss	Genome Survey Sequence

A family of BLAST programs have been developed since its original version was launched in 1990. The difference mainly comes from the input type and the databases that the input is searched against. For example, BLASTN is useful to identify an unknown nucleotide sequence or to search homologous genomic sequences in different organisms. The major BLAST programs on NCBI website are summarized in Table 3. These programs can be used via a web interface (<http://www.ncbi.nlm.nih.gov/blast>) or as stand-alone tools.

Apart from the set of programs on NCBI BLAST server, there are other BLAST-like homology search programs and web servers. Table 4 lists a few widely-used tools. On WU-BLAST server, BLAST programs available on NCBI are also available, but all the programs were implemented differently. WU-BLAST also includes other tools developed in Warren Gish's lab.

FASTA is a sequence similarity search program first developed by Lipman and Pearson in NCBI. Its sequence format, called the FASTA format, has been widely adopted for sequence comparison. It uses a multiple-step approach to aligning the query and target sequences. It first finds runs of *ktup* or more identities, which are called word matches. Here *ktup* is a program parameter used for controlling the sensitivity and speed of the program. From these identified word matches, it determines a band in which good alignments likely locate and then calculates the optimal alignment in the band using the dynamic programming method.

Sequence Search and Alignment by Hashing Algorithm (SSAHA) is developed to search large DNA database efficiently. The essential idea is to preprocess the sequences in a database by breaking them into consecutive *k*-tuples of *k* contiguous bases and then using a hash table to store them. Therefore, searching for a query sequence in the database is done by obtaining from the hash table the 'hits' for each *k*-tuple in the query sequence and then performing a sort on the results.

Sim4 employs a BLAST-based approach. It first determines the maximal scoring gap-free segments and then extends these segments into the adjacent regions greedily. It can be downloaded from Webb Miller's lab and installed in a standalone workstation. It can also be run through the web server <http://pbil.univ-lyon1.fr/sim4.php>.

BLAT is an alignment tool like BLAST, and it is extremely efficient, developed by Jim Kent. On DNA sequences, BLAT works by keeping an index of an entire genome, consisting of all non-overlapping 11-mers, which makes BLAT quickly find sequences of 95% and greater similarity of length 40 bases or more. However,

**Table 3** Major BLAST programs on NCBI website

Program	Description
BLASTN	Search a nucleotide sequence against a nucleotide sequence database
BLASTP	Search an amino acid sequence against a protein sequence database
BLASTX	Search a nucleotide sequence translated in all reading frames against a protein sequence database
TBLASTN	Search a protein sequence against a nucleotide sequence database dynamically translated in all reading frames
TBLASTX	Search the six-frame translations of a nucleotide sequence against the six-frame translations of a nucleotide sequence database
MEGABLAST [29]	Find long alignments between very similar sequences more efficiently
PSI-BLAST [7]	Find members of a protein family or build a custom position-specific score matrix
PHI-BLAST [28]	Find proteins similar to the query around a given pattern

it is less sensitive to more divergent or short sequence alignments. On protein sequences, BLAT uses 4-mers, rapidly finding protein sequences of 80% and greater similarity to the query of length longer than 20 amino acids. However, it is far less sensitive than BLAST and PSI-BLAST at NCBI.

### 3 Algorithm Description

Alignment is a way of arranging two DNA or protein sequences to identify regions of similarity that are conserved among species. Each aligned sequence appears as a row within a matrix. Gaps are inserted between the residues of each sequence so that identical or similar bases in different sequences are aligned in successive positions. Each gap spans one or more columns within the alignment matrix. The score of an alignment is calculated by summing the rewarding scores for match columns

**Table 4** Other BLAST-like programs

Program	Description	URL	Refs
WU-BLAST	Washington University BLAST	<a href="http://blast.wustl.edu/">http://blast.wustl.edu/</a>	[17]
FASTA	Homology search against Protein or DNA databases	<a href="http://fasta.bioch.virginia.edu/">http://fasta.bioch.virginia.edu/</a>	[26]
SSAHA	Fast matching and alignment of DNA sequences	<a href="http://www.sanger.ac.uk/">http://www.sanger.ac.uk/</a>	[24]
Sim4	Homology search of an expressed DNA sequence (EST, cDNA, mRNA) with a genomic sequence	<a href="http://www.bx.psu.edu/miller_lab/">http://www.bx.psu.edu/miller_lab/</a>	[15]
BLAT	BLAST-Like Alignment Tool	<a href="http://genome.ucsc.edu/">http://genome.ucsc.edu/</a>	[21]

that contain the same bases and the penalty scores for gaps and mismatch columns that contain different bases. A scoring scheme specifies the scores for matches and mismatches, which form the scoring matrix, and the scores for gaps, called the gap cost. There are two types of alignments for sequence comparison. Given a scoring scheme, calculating a global alignment is a kind of global optimization that ‘forces’ the alignment to span the entire length of two query sequences, whereas local alignments just identify regions of similarity within two sequences.

The original version of BLAST finds good ungapped local alignments between the query and database sequences [6]. Accordingly, it is also called ungapped BLAST. Database sequences are usually called target sequences. To speed up the homology search process, BLAST employs a filtration strategy: It first scans the database for length- $w$  word matches of alignment score at least  $T$  between the query and target sequences and then extends each match in both ends to generate local alignment (in the sequences) whose alignment score is larger than a threshold  $S$ , which is often called high-scoring segment pairs (HSPs). BLAST outputs a list of HSPs together with E-value that measures how frequent such HSPs would occur by chance.

### 3.1 Phase 1: Scan the Database for Match Hits

Consider a set of parameters  $w$ ,  $T$  and  $S$ . A sequence of length  $w$  is called a  $w$ -mer. For a query sequence, a  $w$ -mer is called a neighborhood sequence if it forms a match of alignment score at least  $T$  with some  $w$ -mer in the query sequence. We illustrate this concept using a DNA query sequence.

Consider query sequence  $Q$ : GCATTGACCC and parameters  $w = 8, T = 6$ . Under a simple scoring scheme by which matches and mismatches score 1 and -1 respectively, the neighborhood sequences that match 8-mer GCATTGAC in the query sequence are all 1-mismatch 8-mers:

.CATTGAC, G.ATTGAC, GC.TTGAC, GCA.TGAC,  
GCAT.GAC, GCATT.AC, GCATTG.C, GCATTGA.,

where ‘.’ stands for any letter of A, G, C, and T. Similarly, the set of neighborhood sequences also include the following sequences:

.ATTGACC, C.TTGACC, CA.TGACC, CAT.GACC,  
CATT.ACC, CATTG.CC, CATTGA.C, CATTGAC.,  
.TTGACCC, A.TGACCC, AT.GACCC, ATT.ACCC,  
ATTG.CCC, ATTGA.CC, ATTGAC.C, ATTGACC.,

which match 8-mers CATTGACC or ATTGACCC.

The set of neighborhood sequences is efficiently constructed from the query sequence since there are at most  $4^w$  neighborhood sequences. Having the set of neighborhood sequences, the next task is to check whether each neighborhood sequence occurs in the target sequence or not. Such an occurrence of a neighborhood sequence

is called a seed hit. For example, for target sequence

$T$ : ATAGCATGGACTTGACCCCGGCATTGTCATCG,

the 8-mer GCATTGAC hits  $T$  at positions 4 and 21, whereas the 8-mer ATTGACCC hits  $T$  at position 11. Here seed hits are not perfect. As the matter of fact, BLAST programs use perfect hits for DNA sequence search and imperfect hits whose score is higher than a threshold. All the hits can be identified using an efficient data structure such as hash table, suffix tree, or suffix array. The reader is referred to the book [12] of Chao and Zhang for implementation details.

The sensitivity and speed of BLAST search are closely related to the match size  $w$ . When  $w$  is big, the BLAST search is fast but has low sensitivity in the sense that it may miss short homologous sequences. In contrast, when  $w$  is small, it is slower, but has high sensitivity. The  $w$  is set by default to 11 and 3 for BLASTN and BLASTP, respectively. To achieve the optimal balance between sensitivity and speed, the discontinuous MEGABLAST finds  $l$ -mer pairs that match in  $w$  discontinuous positions specified by a fixed pattern. Such a pattern is called a spaced seed. For example, one default spaced seed used for searching non-coding sequences is  $111 * 1 * 11 * * 1 * 11 * 111$ . When such a spaced seed is used, two 18-mers match if they have identical nucleotides in the positions indicated by the 1s: 1, 2, 3, 5, 7, 8, 11, 13, 14, 16, 17, 18. It is first observed by Ma, Tromp and Li that optimized spaced seed significantly improves homology search sensitivity [23].

### 3.2 Phase 2: Hit extension

In the second phase, ungapped BLAST extends each ‘seed’ hit in both ends to generate a HSP and outputs this HSP if its alignment score is  $S$  or greater. At each end, the extension includes aligned pairs in successive positions, with corresponding increments to the alignment score. It continues until the alignment score drops more than  $X$  below the maximum score that has attained up to that position.

It was observed that ungapped BLAST consumes more than 90% of the running time in hit extension. It was also observed that an HSP usually contains multiple hits that are close to one another. Accordingly, Gapped BLAST uses double hits to trigger hit extension to generate high-scoring gapped local alignments. It starts the extension process only if there are two non-overlapping hits within  $D_g$  positions, where the subscript  $g$  indicates that it is a parameter for Gapped BLAST. These adjacent non-overlapping hits can be detected if all hit positions are maintained.

In Gapped BLAST, gap extension is done by using the dynamic programming approach. Since the approach takes quadratic time, the extension process is much slower than ungapped one. Here two more ideas are employed in order to handle gap extension more efficiently. One idea is only to extend those HSPs that have alignment score  $S_g$  or greater. The threshold  $S_g$  is determined in such a way that only one gap extension is invoked on average in per 50 database sequences. Another idea for handling the extension is to restrict gapped extension to those positions in which

the optimal local alignment score drops no more than  $X_g$  below the maximum local alignment score attained up to the position.

## 4 BLAST Statistics

An important feature of BLAST is that it rank-orders the reported HSPs by E-values. For a local alignment of score  $s$ , An E-value of  $10^{-5}$  is often used as a cutoff for BLAST homology search. It means that with a collection of random query sequences, only once in a hundred thousand of instances would an alignment with that score or greater occur by chance. The smaller the E-value, the greater the belief that the aligned sequences are homologous.

The E-values for HSPs in BLAST printout are calculated based on the seminal work of Karlin and Altschul on the distribution of optimal ungapped local alignment scores [19]. Both theoretical and empirical studies suggest that the distributions of optimal local alignment scores  $s$  with or without gaps are accurately described by an extreme value distribution.

Assume that we search a query sequence  $Q$  against a database. Let  $l_Q$  be the length of  $Q$ . For each database sequence  $T$ , the mean number  $E_T$  of HSPs with score  $s$  or greater occurring in the comparison of  $Q$  and  $T$  is

$$E_T = K(l_Q - \bar{l}(s))(l_T - \bar{l}(s))e^{-\lambda s}, \quad (1)$$

where  $K$  and  $\lambda$  are constants independent of  $T$  and  $\bar{l}(s)$  is the length adjustment.  $K$  and  $\lambda$  are the two parameters of the extreme value distribution of optimal local alignment scores. Their values are efficiently calculated from the letter composition of the database sequences and the scoring scheme used for the search. The values of  $K$  and  $\lambda$  are listed in BLAST search printout.

The length adjustment  $\bar{l}(s)$  is equal to the mean length of HSPs with score  $s$  or greater. It is used to eliminate the ‘edge’ effect of the fact that optimal local alignment locates unlikely at the end of both query and target sequences. Let  $N$  and  $M$  be the numbers of sequences and letters in the database. The current BLASTP (version 2.2.18) calculates the length adjustment  $\bar{l}(s)$  for score  $s$  as an integer-valued approximation to the unique root of the following functional equation

$$x = \alpha \frac{\ln(K(l_Q - x)(M - Nx))}{\lambda} + \beta. \quad (2)$$

For ungapped alignment,  $\alpha = \lambda/H$ , and  $\beta = 0$ , where  $H$  is the relative entropy of the scoring matrix used for the database search. For gapped alignment, the values of  $\alpha$  and  $\beta$  depend on scoring matrix and affine gap cost. Take BLOSUM62 as an example. We have that  $\alpha = 1.90$  and  $\beta = -29.70$  for the affine gap cost in which the gap opening and extension costs are 11 and 1 respectively.

We define the effective size of search space as

$$\text{eff-searchSP} = \sum_{T \in \mathcal{D}} l_T - N\bar{l}(s). \quad (3)$$

According to the linearity property of means, the expected number of high-scoring alignments with score  $s$  or greater found in the entire database is

$$\text{E-value} = \sum_{T \in \mathcal{D}} E_T = K \times (l_Q - \bar{l}(s)) \times \text{eff-searchSP} \times e^{-\lambda s}. \quad (4)$$

When two sequences are aligned, insertions and deletions can break a long alignment into several parts. If this is the case, focusing on the single highest-scoring segment could lose useful information. As an option, one may consider the scores of the multiple highest-scoring segments.

Assessing multiple highest-scoring segments is more involved than it might first appear. Suppose, for example, comparison X reports two highest scores 88 and 68, whereas comparison Y reports 79 and 75. One can say that Y is not better than X, because its high score is lower than that of X. But neither is X considered better, because the second high score of X is lower than that of Y. The natural way to rank all the possible results is to consider the sum of the alignment scores of the HSPs as suggested by Karlin and Altschul [20]. This sum is now called the Karlin-Altschul sum statistic.

In the earlier version of BLAST, the Karlin-Altschul sum statistic was only used for ungapped alignments as an alternative to performing gapped alignment. Now, it is applied to any HSP. The Karlin-Altschul sum statistics is too involved to be described here due to the space limit.

Finally, we must warn that formulas for P-value and E-value in BLAST are evolving. The above calculations are used in the current version of BLAST (version 2.2). They are different from what were used in the earlier versions. The length adjustment was calculated as the product of  $\lambda$  and the raw score divided by  $H$  in the earlier version. Accordingly, they might be modified again in future.

## 5 Examples

### 5.1 A BLASTP Search Example

As an example of using BLASTP, we will consider the capsid protein of the West Nile Virus (WNV)<sup>1</sup>. This virus mainly infects birds, but occasionally infects humans through the bite of an infected mosquito. The WNV is a positive-sense, single strand of RNA, having about 11,000 nucleotides. There are 7 non-structural proteins and 3 structural proteins in the RNA. The capsid protein of the WNV has sequence [10]:

MSKKPGGPGK SRAVNMLKRG MPRVLSLIGL KRAML SLIDG KGP IRFVLAL LAFFRFTAIA

<sup>1</sup> This example first appeared in the article of Casey [11].

PTRAVLDRWR GVNKQTAMKH LLSFKKELGT LTSAINRRSS KQKKR

whose accession id is YP\_001527877. We compare this sequence against the non-redundant GenBank by using BLASTP available at the NCBI server with default setting. A BLAST printout contains (a) the information on the program, (b) a set of local alignments together with its statistical scores, and (c) a set of parameters used for the statistical analysis. A partial printout from our search follows:

#### BLASTP 2.2.18+

...

**Database:** All non-redundant GenBank CDS translations+PDB+SwissProt+PIR+PRF excluding environmental samples from WGS projects  
7,036,788 sequences; 2,431,208,758 total letters

...

**Query=** gi|158516889|ref|YP\_001527878.1| capsid protein [West Nile virus]  
Length=105

...

#### Alignments

>gb|ABD67759.1| polyprotein precursor [West Nile virus]  
Length=3433

Score = 203 bits (517), Expect = 3e-51, Method: Composition-based stats.  
Identities = 100/100 (100%), Positives = 100/100 (100%), Gaps = 0/100 (0%)

Query 1 MSKKPGGPGKSRVAVNMLKRGMPRVLSLIGLKRAMLSDIDGKPIRFVLALLAFFRFTAIA 60  
MSKKPGGPGKSRVAVNMLKRGMPRVLSLIGLKRAMLSDIDGKPIRFVLALLAFFRFTAIA  
Sbjct 1 MSKKPGGPGKSRVAVNMLKRGMPRVLSLIGLKRAMLSDIDGKPIRFVLALLAFFRFTAIA 60

Query 61 PTRAVLDRWRGVNKQTAMKHLLSFKKELGTLTSAINRRSS 100  
PTRAVLDRWRGVNKQTAMKHLLSFKKELGTLTSAINRRSS  
Sbjct 61 PTRAVLDRWRGVNKQTAMKHLLSFKKELGTLTSAINRRSS 100

...

>gb|ACA28703.1| polyprotein [Japanese encephalitis virus]  
Length=3432

Score = 164 bits (414), Expect = 2e-39, Method: Composition-based stats.  
Identities = 71/105 (67%), Positives = 90/105 (85%), Gaps = 0/105 (0%)

Query 1 MSKKPGGPGKSRVAVNMLKRGMPRVLSLIGLKRAMLSDIDGKPIRFVLALLAFFRFTAIA 60  
M+KKPGGPGK+RA+NMLKRG+PRV L+G+KR ++SL+DG+GP+RFVLAL+ FF+FTA+A  
Sbjct 1 MTKKPGGPGKNRAINMLKRGLPRVPLVGVKRVVMSLLDGRGPVRFVLALITFFKFTALA 60

Query 61 PTRAVLDRWRGVNKQTAMKHLLSFKKELGTLTSAINRRSSKQKKR 105  
PT+A+L RWR V K AMKHL SFK+ELGTL A+N+R KQ KR  
Sbjct 61 PTKALLGRWRAVEKSVAMKHLTSFKRELGTLDVAVNKRGGKQNKR 105

...

**Database:** All non-redundant GenBank CDS translations+PDB+SwissProt+PIR+PRF excluding environmental samples from WGS projects  
Posted date: Sep 9, 2008 5:57 PM  
Number of letters in database: -1,863,758,534  
Number of sequences in database: 7,036,788

```

Lambda      K      H
  0.324      0.137  0.389
Gapped
Lambda      K      H
  0.267      0.0410  0.140
Matrix: BLOSUM62
Gap Penalties: Existence: 11, Extension: 1
Number of Sequences: 7036788
...
Length of query: 105
Length of database: 2431208758
Length adjustment: 111
...

```

The second local alignment in the printout shows that the capsid protein of the WNV has a significant similarity with a domain region of the Japanese encephalitis virus. Our BLAST search reveals correctly the fact that the Japanese encephalitis virus and the WNV share similar proteins in their protein coats.

The statistical analysis associated with each alignment in the printout is done as follows. As shown in the printout, the query sequence has 105 letters; the target database contains 7,036,788 sequences and 2,431,208,758 letters; and the length adjustment displayed in the printout is 73. Since the local alignment involving the Japanese encephalitis virus is gapped, the following values are used in the calculation of the E-value:

$$\lambda = 0.267, \quad K = 0.041.$$

The raw score of the alignment is 414 and hence its bit score is

$$\frac{\lambda \times S_{raw} - \ln(K)}{\ln(2)} = \frac{0.267 \times 414 - \ln(0.041)}{\ln(2)} = 164.080851,$$

which agrees with 164 in the printout [16]. By (4), the E-value is

$$0.041 \times (105 - 73) \times (2,431,208,758 - 7,036,788 \times 73) \times e^{-0.267 \times 414},$$

which is  $2.481035e-39$ , in agreement with the printout value  $2e-39$ .

## 5.2 A PSI-BLAST Search Example

The Position-Specific Iterated (PSI)-BLAST was designed to identify subtle homologous protein relationships that might be missed by other BLAST programs [7]. It searches a protein database iteratively. At each iteration step, PSI-BLAST generates a profile, or a position specific scoring matrix (PSSM), based on a multiple alignment of the identified high scoring hits to a given query sequence. The PSSM is calculated by considering position-specific scores for each position in the alignment. Highly conserved positions receive high scores, whereas weakly conserved positions receive low scores. The profile is then used to perform subsequent round of BLAST search. The strategy is to use the results of each iteration to refine the profile progressively. When such a profile is used to search a database, it can often

**Fig. 1** The webpage for launching a PSI-BLAST search.

detect distantly homologous, in structure or function, relationships between proteins.

We illustrate how to operate PSI-BLAST by searching part (the first 300 bp) of a putative zinc finger protein (XP\_656065.1) in *Entamoeba histolytica* against the non-redundant protein sequence database as an example. We run the online version of PSI-BLAST available at <http://ncbi.nlm.nih.gov/BLAST>. The search is done in the following steps:

1. Paste the query sequence into the query box of the PSI-BLAST Web page and choose the searched database. Here, we used the accession id XP\_656065.1 of the putative zinc finger protein (see Figure 5.1).
2. Set the algorithm parameters. If one chooses to use the default parameters, this step is skipped. In our example, we changed the maximum number of target sequences from 500 to 1000, the expected threshold from 10 to 4, and the PSI-BLAST threshold from 0.005 to 0.05.
3. Format to get the results. We ticked the box next to the BLAST button for the results to be retrieved in a new webpage. The hits will be displayed into two sections. The hits with E-value smaller than the threshold  $s$ , 0.05 in our search, are listed first; those with E-value larger than  $s$  but smaller than the expected threshold, 4 in our case, are listed further down the page. The hits listed in the first section will be used in forming the profile that will be used in the next iteration step.
4. Click repeatedly the 'run PSI-BLAST iteration' button until the user decides to stop the search process or the search result cannot be improved. By clicking the 'Taxonomy reports' link on the top of the result window, one can view the distribution of the hits and decide to stop the search or not.

Hit list size

[Distance tree of results](#) **NEW**

---

**Sequences with E-value BETTER than threshold**

Sequences producing significant alignments:

			Score (Bits)	E Value		
<b>NEW</b>	<input checked="" type="checkbox"/>	<a href="#">ref XP_656065.1 </a>	zinc finger protein, putative [Entamoeba his...	<a href="#">609</a>	7e-173	<b>G</b>
<b>NEW</b>	<input checked="" type="checkbox"/>	<a href="#">ref XP_001738912.1 </a>	A Kinase anchor protein, putative [Entamo...	<a href="#">585</a>	1e-165	<b>G</b>
<b>NEW</b>	<input checked="" type="checkbox"/>	<a href="#">ref XP_629445.1 </a>	FYVE-type Zn finger-containing protein [Dict...	<a href="#">47.8</a>	0.001	<b>G</b>

---

**Sequences with E-value WORSE than threshold**

<input type="checkbox"/>	<a href="#">ref XP_001021993.1 </a>	arrestin domain protein [Tetrahymena ther...	<a href="#">37.0</a>	1.6	<b>UG</b>
<input type="checkbox"/>	<a href="#">gb EDN64421.1 </a>	conserved protein [Saccharomyces cerevisiae YJ...	<a href="#">36.6</a>	2.7	
<input type="checkbox"/>	<a href="#">ref ZP_02164687.1 </a>	beta-mannosidase precursor [Hoeftlea photot...	<a href="#">36.2</a>	3.0	
<input type="checkbox"/>	<a href="#">ref NP_013742.1 </a>	Protein involved in G1 cell cycle arrest in ...	<a href="#">36.2</a>	3.0	<b>G</b>
<input type="checkbox"/>	<a href="#">ref XP_002121784.1 </a>	PREDICTED: hypothetical protein [Ciona in...	<a href="#">35.8</a>	3.8	
<input type="checkbox"/>	<a href="#">ref XP_963615.1 </a>	hypothetical protein NCU06774 [Neurospora cr...	<a href="#">35.8</a>	3.8	<b>G</b>
<input type="checkbox"/>	<a href="#">ref XP_002110570.1 </a>	hypothetical protein TRIADDRAFT_23340 [Tr...	<a href="#">35.8</a>	4.0	<b>G</b>

**Fig. 2** The initial PSI-BLAST search results.

We obtained three significant hits (shown in Figure 5.2) in the initial search and surprisingly nine more significant hits (Figure 5.2) after the first iteration. The following several iterations generated even more significant hits.

PSI-BLAST is a powerful tool. Many important but subtle relationships that previously were detectable only by structural comparison can now be uncovered by a simple PSI-BLAST search. However, the user must use it with caution. The false relationship can be easily amplified by iteration. As a result, different queries that belong to the same family of proteins can perform differently in searches against the same database. It is recommended that the user run PSI-BLAST search with different query sequences to obtain reliable homologous relationship.

## 6 Advanced Topics

NCBI web interface provides biologists an easy access to BLAST homology search against different databases. It has a simple search form on which a dozen of default values can be overwritten and displays aligned sequences together with significant analysis. But, using BLAST effectively requires knowledge of alignment statistics and insights on the algorithmic details of the program.

Sequences with E-value BETTER than threshold						
Sequences producing significant alignments:				Score	E	
				(Bits)	Value	
<input checked="" type="checkbox"/>	<a href="#">ref XP_656065.1 </a>	zinc finger protein, putative [Entamoeba his...	<a href="#">.547</a>	4e-154	<a href="#">G</a>	
<input checked="" type="checkbox"/>	<a href="#">ref XP_001738912.1 </a>	A kinase anchor protein, putative [Entamo...	<a href="#">.546</a>	6e-154	<a href="#">G</a>	
<input checked="" type="checkbox"/>	<a href="#">ref XP_629445.1 </a>	FYVE-type Zn finger-containing protein [Dict...	<a href="#">.265</a>	3e-69	<a href="#">G</a>	
NEW	<input checked="" type="checkbox"/>	<a href="#">ref XP_001021993.1 </a>	arrestin domain protein [Tetrahymena ther...	<a href="#">51.2</a>	9e-05 <a href="#">U</a> <a href="#">G</a>	
NEW	<input checked="" type="checkbox"/>	<a href="#">ref XP_001021994.1 </a>	arrestin domain protein [Tetrahymena ther...	<a href="#">48.1</a>	9e-04 <a href="#">G</a>	
NEW	<input checked="" type="checkbox"/>	<a href="#">ref XP_001436511.1 </a>	hypothetical protein GSPATT00037550001 [P...	<a href="#">45.4</a>	0.005 <a href="#">G</a>	
NEW	<input checked="" type="checkbox"/>	<a href="#">ref XP_001453022.1 </a>	hypothetical protein GSPATT00019342001 [P...	<a href="#">45.0</a>	0.008 <a href="#">G</a>	
NEW	<input checked="" type="checkbox"/>	<a href="#">ref XP_001450786.1 </a>	hypothetical protein GSPATT00039538001 [P...	<a href="#">44.3</a>	0.012 <a href="#">G</a>	
NEW	<input checked="" type="checkbox"/>	<a href="#">ref XP_001952503.1 </a>	PREDICTED: similar to arrestin homolog [A...	<a href="#">43.1</a>	0.023 <a href="#">U</a> <a href="#">G</a>	
NEW	<input checked="" type="checkbox"/>	<a href="#">ref NP_001098299.1 </a>	arrestin [Oryzias latipes] >dbj BAA21719...	<a href="#">43.1</a>	0.024 <a href="#">U</a> <a href="#">G</a>	
NEW	<input checked="" type="checkbox"/>	<a href="#">ref XP_001447043.1 </a>	hypothetical protein GSPATT00014576001 [P...	<a href="#">42.4</a>	0.045 <a href="#">G</a>	
NEW	<input checked="" type="checkbox"/>	<a href="#">gb AAK84368.1 AF393635.1</a>	visual arrestin [Loligo pealeii]	<a href="#">42.4</a>	0.047	
Run PSI-Blast iteration 3						
Sequences with E-value WORSE than threshold						
<input type="checkbox"/>	<a href="#">ref XP_001431708.1 </a>	hypothetical protein GSPATT00006163001 [P...	<a href="#">42.0</a>	0.059	<a href="#">G</a>	
<input type="checkbox"/>	<a href="#">ref XP_001423611.1 </a>	hypothetical protein GSPATT00004307001 [P...	<a href="#">42.0</a>	0.061	<a href="#">G</a>	
<input type="checkbox"/>	<a href="#">ref XP_002116059.1 </a>	hypothetical protein TRIADDDRAFT_60106 [Tr...	<a href="#">42.0</a>	0.066	<a href="#">G</a>	

Fig. 3 The result after the first iteration of PSI-BLAST search.

## 6.1 Scoring Matrices

The statistical significance of HSPs listed in BLAST printout is calculated mainly based on internal scoring matrix. For protein sequence comparison, BLOSUM or PAM matrices are usually used; for DNA sequence comparison, similar but simpler substitution matrices are used. Although these scoring matrices are derived in different approach, they take essentially the same log-odds form: the score for aligning bases  $a$  and  $b$  is basically the logarithm of the ratio of the probability that  $a$  and  $b$  are aligned in homologous sequences to the probability that we expect to observe  $a$  and  $b$  aligned in random sequences with the same background letter composition.

The choice of scoring matrix for homology search can have a profound effect on the search output. Choosing PAM120 will generate one HSP, whereas choosing BLOSUM62 will generate another. In other words, scoring matrices detect different classes of alignments. For example, in searching a protein database containing 10,000,000 letters, the length range of the local alignments that PAM120 can detect is roughly from 15 to 50 (see [1]). Accordingly, choosing PAM120 may miss short but strong or long but weak alignments.

To obtain good result, one should choose proper scoring matrix rather than just using the default matrix listed on the web interface. In PAM matrix naming system, higher numbers denote larger evolutionary distance. Hence, PAM120 is generally more appropriate than PAM30 for finding remotely-related homologous sequences. In contrast, larger numbers in BLOSUM matrices denote higher sequence similarity

and so BLOSUM45 is generally better than BLOSUM62 for studying sequence relationship among divergent species.

## 6.2 Gap Penalties

Another important issue of BLAST search is to choose gap cost. It is rare for two gene sequences to align perfectly with one another. Gapped BLAST introduces gaps between residues to bring up matches in the following positions. The way Gapped BLAST treats gaps can significantly affect its output. The user is allowed to set choose different costs for gap opening and gap extension in a BLAST webpage.

It is observed that the optimal local alignments do not likely contain gaps of more than 1 residues [5] if the gap extension cost is relatively large. Hence, it is not rewarding to use any gap extension cost that is too close to the gap opening one.

On the other hand, if the gap extension cost is too small, the optimal local alignment scores might not follow the extreme value distribution. If this happens, the statistical analysis done by BLAST will be no longer meaningful.

## 6.3 Should DNA or Protein Sequence Be Used?

One natural question often asked by a BLAST user is: Should I compare gene sequences or the corresponding protein sequences? This can be answered by the following analysis.

Synonymous mutations are nucleotide substitutions that do not result in a change to the amino acids sequence of a protein. Evolutionary study suggests that there tend to be approximately 1.5 synonymous point mutations for every nonsynonymous point mutation. Because each codon has 3 nucleotides, each protein PAM translates into roughly  $\frac{1+1.5}{3} \approx 0.8$  PAMs in DNA level.

The substitution scores in the scoring matrix are implicitly log-odds scores. By multiplying a constant factor, a scoring matrix is normalized to the logarithms of odds-ratios to base 2. The alignment score obtained with such a normalized scoring matrix is called bit score and considered as bit information. In the alignment of two proteins that have diverged by 120 PAMs, each residue carries on average 0.98-bit information, whereas in the alignment of two DNA sequences that are diverged at 96 (or  $120 \times 0.8$ ) PAMs, every three residues (a codon) carry only about 0.62-bit information [27]. Hence, at this evolutionary distance, 37% of the information available in protein comparison will be lost in DNA sequence comparison.

In a nutshell, protein sequence comparison is generally more sensitive than that of DNA sequences.

## 7 Exercises

Here we list 7 exercise problems for the reader to gain mastery of BLAST and BLAST-like programs.

1. Which of the following in the BLAST output provides an estimate of the false positive of the BLAST search, E-value, Score, or Identity?
2. What are the major advantages of Gapped-BLAST over BLASTP?
3. Use one of the BLAST programs to determine the frame shift of the following sequence.

```
ATGAGAGTGAAGGAGAAATATCAGCACTTGTGGAGATGGGGCACCATGCTCCTTGGGTTGT
TGATGATCCGTAGTGCTGCAGACCAATTGTGGGTCACAGTCTATTATGGGGTACCTGTGTG
GAAAGAAGCAACCACCACTCCATTTTGTGCATCAGATGCTAAAGCATATGATACAGAGGTA
CATAATGTTTGGGCCACACACGCCTGTGTACCCACAGACCCCAACCCACAAGAAGTAGTAT
TGGCAAATGTGGCAGAAAATTTTAACATGTG
```

4. Use BLASTP to align protein sequence P23749 and P16235 with different matrices, PAM30, BLOSUM45, and BLOSUM80. Which matrix gave the best alignment score?
5. Higher eukaryotic genomes contain large amounts of repetitive DNA. The most abundant interspersed repeat in the human genome is the Alu element. Alus tend to occur near genes, within the introns of genes, or in the regions between genes. In some cases, their presence and absence can fairly accurately show the intron-exon structure of a gene. Demonstrate this by performing a nucleotide-nucleotide BLAST search against the Alu database (alu\_repeats) with the genomic sequence of the human Von Hippel Lindau syndrome gene (Accession AF010238). Note that the exons appear in the BLAST graphic as places where the Alu elements do not align.
6. A PSI-BLAST search is most useful when you want to do:
  - a. Extend a database search to find additional proteins,
  - b. Extend a database search to find additional DNA sequences,
  - c. Find the mouse ortholog of a human protein, or
  - d. Use a pattern to extend a protein search.

The human fragile histidine triad protein (FHIT, Accession P49789) is structurally related to galactose-1-phosphate uridylyltransferases. However, this relationship is not apparent in an ordinary BLAST search. Perform a protein-protein BLAST search against the SWISS-PROT database with P49789 and search your results for galactose-1-phosphate uridylyltransferases. Now use PSI-BLAST to verify the relationship between these two protein families.

7. On UCSC Genome Browser (<http://genome.ucsc.edu/>), find the protein sequence for rat leptin. BLAT this sequence against the human genome to find the human homology. Look for SNPs in the coding region of this gene. Are there any?

## 8 Further Reading

Here we just like to point out the most relevant and useful references on the topics covered in this chapter for the reader to consult. Sequence alignment has been extensively studied by biologists, computer scientists, and mathematicians in the past four decades. There is a large body of literature on this subject matter. For general treatment of sequence alignment, we refer the reader to the survey papers of Batzoglou [9] and Altschul et al. [3] and the book of Chao and Zhang [12].

For further consultation on how to use BLAST, we refer the reader to the online tutorial on the BLAST server at NCBI or the book of Korf, Yandell and Bedell [22].

In 1990, Karlin and Altschul published their seminal work [19] on the distribution of optimal ungapped local alignment scores. Later, Altschul and Gish [5] and Pearson [25] investigated empirically the distribution of optimal gapped local alignment scores. Karlin-Altschul statistics of local alignment scores are surveyed by Pearson and Wood in [8] and Karlin in [18] and covered in the books of Chao and Zhang [12] and Ewens and Grant [14].

For the general theory of scoring matrices, the reader is referred to the papers of Altschul [1, 2] and Eddy [13]. The information on parameters and formulas used for statistical analysis in BLAST can be found in the paper [4] of Altschul et al. and the note (<ftp://ftp.ncbi.nlm.nih.gov/blast/documents/developer/scoring.pdf>) of Gertz [16].

**Acknowledgements** LX Zhang was partially supported by Singapore ARF grant R146-000-109-112. He would also like to thank Stephen Altschul for useful communication (through e-mail) on scoring matrices and alignment statistics and Kun-Mao Chao, Wayne Matten and Scott D. McGinnis for discussion of the implementation issues of BLAST programs.

## References

1. Altschul, S.F.: Amino acid substitution matrices from an information theoretic perspective. *Journal of Molecular Biology* **219**(3), 555–565 (1991)
2. Altschul, S.F.: A protein alignment scoring system sensitive at all evolutionary distances. *Journal of Molecular Evolution* **36**, 290–300 (1993)
3. Altschul, S.F., Boguski, M.S., Gish, W., Wootton, J.C.: Issues in searching molecular sequence databases. *Nature Genetics* **6**, 119–129 (1994)
4. Altschul, S.F., Bundschuh, R., Olsen, R., Hwa, T.: The estimation of statistical parameters for local alignment score distributions. *Nucleic Acids Research* **29**, 351–361 (2001)
5. Altschul, S.F., Gish, W.: Local alignment statistics. *Methods in Enzymology* **266**(2), 460–480 (1996)

6. Altschul, S.F., Gish, W., Miller, W., Myers, E.W., Lipman, D.J.: Basic local alignment search tool. *Journal of Molecular Biology* **215**(3), 403–410 (1990)
7. Altschul, S.F., Madden, T.L., Schaffer, A.A., Zhang, J.H., Zhang, Z., Miller, W., Lipman, D.J.: Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research* **25**(17), 3389–3402 (1997)
8. Balding, D., Bishop, M., Cannings, C. (eds.): *Handbook of Statistical Genetics*, chap. 2, pp. 39–65. John Wiley & Sons (2003)
9. Batzoglou, S.: The many faces of sequence alignment. *Briefings in Bioinformatics* **6**(1), 6–22 (2005)
10. Borisevich, V., Seregin, A., Nistler, R., Mutabazi, D., Yamshchikov, V.: Biological properties of chimeric West Nile viruses. *Virology* **349**(2), 371–381 (2006)
11. Casey, R.M.: Blast sequences aid in genomics and proteomics. <http://www.b-eye-network.com/print/1730> (2005)
12. Chao, K.M., Zhang, L.: *Sequence Comparison: Theory and Methods*. Springer (2008)
13. Eddy, S.R.: Where did the BLOSUM62 alignment score matrix come from? *Nature Biotechnology* **22**(8), 1035–1036 (2004)
14. Ewens, W.J., Grant, G.R.: *Statistical Methods in Bioinformatics: An Introduction*. Springer-Verlag (2001)
15. Florea, L., Hartzell, G., Zhang, Z., Rubin, G., Miller, W.: A computer program for aligning a cDNA sequence with a genomic DNA sequence. *Genome Research* **8**(9), 967–974 (1998)
16. Gertz, E.M.: Blast scoring parameters (2005)
17. Gish, W.: personal communication
18. Karlin, S.: Statistical signals in bioinformatics. *Proc Nat'l Acad Sci USA* **102**, 13,355–13,362 (2005)
19. Karlin, S., Altschul, S.F.: Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc Nat'l Acad Sci USA* **87**, 2264–2268 (1990)
20. Karlin, S., Altschul, S.F.: Applications and statistics for multiple high-scoring segments in molecular sequences. *Proc Nat'l Acad Sci USA* **90**, 5783–5877 (1993)
21. Kent, W.J.: BLAT-The BLAST-like alignment tool. *Genome Research* **12**(4), 656–664 (2002)
22. Korf, I., Yandell, M., Bedell, J.: BLAST. O'Reilly Media, Inc. (2003)
23. Ma, B., Tromp, J., Li, M.: PatternHunter-faster and more sensitive homology search. *Bioinformatics* **18**, 440–445 (2005)
24. Ning, Z., Cox, A.J., Mullikin, J.C.: SSAHA: A fast search method for large DNA databases. *Genome Research* **11**(10), 1725–1729 (2001)
25. Pearson, W.R.: Empirical statistical estimates for sequence similarity searches. *Journal of Molecular Biology* **276**(1), 71–84 (1998)
26. Pearson, W.R., Lipman, D.J.: Improved tools for biological sequence comparison. *Proceedings of the National Academy of Sciences* **85**(8), 2444–2448 (1988)
27. States, D.J., Gish, W., Altschul, S.F.: Improved sensitivity of nucleic acid databases searches using application-specific scoring matrices. *Methods* **3**(1), 61–71 (1991)
28. Zhang, Z., Schaffer, A.A., Miller, W., Madden, T.L., Lipman, D.J., Koonin, E.V., Altschul, S.F.: Protein sequence similarity searches using patterns as seeds. *Nucleic Acids Research* **26**(17), 3986–3990 (1998)
29. Zhang, Z., Schwartz, S., Wagner, L., Miller, W.: A greedy algorithm for aligning DNA sequences. *Journal of Computational Biology* **7**(1-2), 203–214 (2000)



# Index

## A

Alignment 5  
  gapped 6, 8  
  global 5  
  local 5  
  score 5  
  ungapped 7, 8

## B

Bit score 14  
BLAST and BLAST-like programs 1, 3  
  BLASTN 3  
  BLASTP 3  
  BLASTX 3  
  BLAT 4  
  Gapped BLAST 6  
  MEGABLAST 3  
  PHI-BLAST 3  
  PSI-BLAST 3  
  TBLASTN 3  
  TBLASTX 3  
  ungapped BLAST 5  
  WU-BLAST 4  
BLOSUM matrices 13  
  BLOSUM45 14, 15  
  BLOSUM62 7, 13, 14  
  BLOSUM80 15

## D

Databases  
  est 2  
  gss 2  
  month 2  
  nr 2

  pdb 2  
  refseq 2

## E

E-value 8  
Effective size of search space 7, 8

## F

FASTA 3, 4  
  sequence format 3  
Filtration strategy 5

## G

Gap cost 5  
  extension 14  
  opening 14

## H

High-scoring segment pairs (HSPs) 5

## K

Karlin-Altschul sum statistic 8

## L

Length adjustment 7

## P

PAM matrices 13  
  PAM120 13  
  PAM30 13, 15  
Point mutation  
  nonsynonymous 14  
  synonymous 14

**S**

Scoring matrix 5  
Scoring scheme 5  
Seed hit 6  
Sim4 4

Spaced seed 6  
SSAHA 4  
Substitution 14  
matrices 13  
score 14