Similarity searching using BLAST

Article *in* Methods in Molecular Biology · February 2009

DOI: 10.1007/978-1-59745-251-9_1 · Source: PubMed

Keith A. Crandall
George Washington University
869 PUBLICATIONS **67,326** CITATIONS

Chapter 1

Similarity Searching Using BLAST

Kit J. Menlove, Mark Clement, and Keith A. Crandall

Abstract

Similarity searches are an essential component of most bioinformatic applications. They form the bases of structural motif identification, gene identification, and insights into functional associations. With the rapid increase in the available genetic data through a wide variety of databases, similarity searches are an essential tool for accessing these data in an informative and productive way. In this chapter, we provide an overview of similarity searching approaches, related databases, and parameter options to achieve the best results for a variety of applications. We then provide a worked example and some notes for consideration.

Key words: BLAST, sequence alignment, similarity search.

1. Introduction

1.1. An Introduction to Nucleotide Databases

Perhaps the central goal of genetics is to articulate the associations of phenotypes of interest with their underlying genetic components and then to understand the relationship between genetic variation and variation in the phenotype. This goal has been buoyed by the tremendous increase in our ability to obtain molecular genetic data, across both populations and species. As methods of gathering information about various aspects of biological macromolecules arose, biological information became abundant and the need to consolidate and make this information accessible became increasingly apparent. In the early 1960s, Margaret Dayhoff and colleagues at the National Biomedical Research Foundation (NBRF) began collecting information on protein sequences and structure into a volume entitled *Atlas of Protein Sequence and Structure (1)*. Since that beginning, databases have been an important and essential part of biological and biochemical research.

David Posada (ed.), *Bioinformatics for DNA Sequence Analysis,* Methods in Molecular Biology 537
© Humana Press, a part of Springer Science+Business Media, LLC 2009
DOI 10.1007/978-1-59745-251-9_1

By 1972, the size of the Atlas had become unwieldy, so Dr. Dayhoff, a pioneer of bioinformatics, developed a database infrastructure into which this information could be funneled. Though nucleotide information was included in the Atlas as early as 1966 *(2)*, its bulk was comprised of amino acid sequences with structural annotation.

1.2. International Nucleotide Sequence Database Collaboration: DDBJ, EMBL, and GenBank

It was not until 1982 that databases were developed with the express purpose of storing nucleotide sequences by the European Molecular Biology Laboratory (EMBL: http://www.embl.org/) in Europe and the National Institutes of Health (NIH – NCBI: http://www.ncbi.nlm.nih.gov/) in North America. Japan followed suit with the creation of their DNA Databank (DDBJ: http://www.ddbj.nig.ac.jp/) in 1986. A sizeable amount of sharing naturally occurred between these three databases and the Genome Sequence Database, also in North America, a condition that led to their coalition in 1988 under the title International Nucleotide Sequence Database Collaboration (INSDC). They still remain very distinct entities, but in the 1988 meeting, they established policies to govern the formatting of and stewardship over the sequences each receives. Their current policies include unrestricted access and use of all data records, proper citation of data originators, and the responsibilities of submitters to verify the validity of the data and their right to submit it. The INSDC currently contains approximately 80 billion base pairs (bp) (not including whole-genome shotgun sequences) and nearly 80 million sequence entries. Including shotgun sequences (HTGS), it passed the 100-gigabase mark on August 22, 2005, and contains approximately 200 billion bp as of September 2007. For more than 10 years, the amount of data in these databases doubled approximately every 18 months. This expansion has begun to level off as our capacity for high-throughput sequencing is gradually reaching a maximum. The next redoubling of the data is expected to occur in approximately 4 years (**Fig. 1.1**).

1.3. Other Nucleotide Sequence Databases

Since the first nucleotide databases were initiated by EMBL and NIH (now held by NCBI), many DNA databases have been formed to cater to the needs of specialized research groups. The 2007 Database issue of *Nucleic Acid Research* contained 109 nucleotide sequence databases that met the standards required to be included in its listing *(3)*. These databases are typically developed to include ancillary data associated with the genetic data, such as patient or specimen information, including clinical information, images, downstream analyses. Many do not meet the standards of "quality, quantity and originality of data as well as the quality of the web interface" that are required to be considered for the issue *(4)*. Even more are privately held to permit access of costly data to a select few. All in all, the number of DNA databases is astounding and steadily increasing as we find new, powerful ways to gather, store, and utilize the pieces that comprise the puzzle of life.

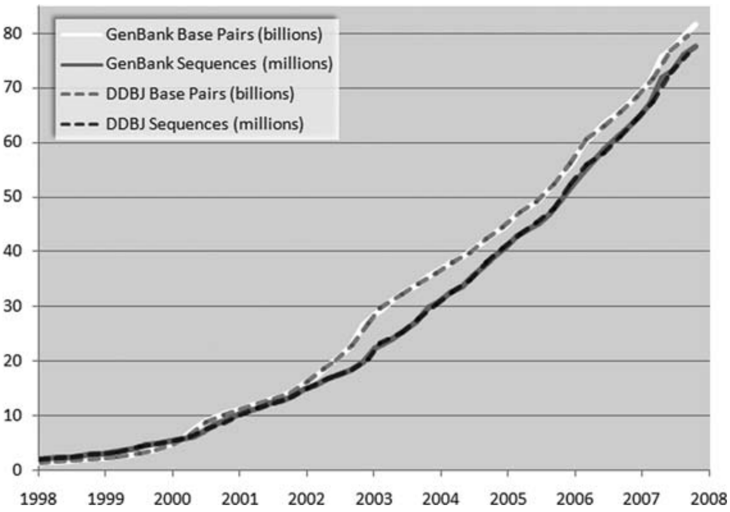


Fig. 1.1. Growth of GenBank and DDBJ genetic databases over the past 10 years. The INSDC databases have grown, over the past 10 years, approximately 168-fold in total number of base pairs. While in the past the number of entries in INSDC databases doubled approximately every 2 years, a simple second-order polynomial regression (R^2=0.9995) of the data over the past 10 years indicates that the next redoubling will take a little over 4 years. This graph does not include HTG data.

2. Program Usage

2.1. Database File Formats

One of the largest sources of diversity among DNA databases lies in their file formats. While great efforts have been made to standardize file formats, the various types and purposes of sequence information and annotation entreat customized file types.

2.1.1. FASTA Format

First used with Pearson and Lipman's FASTA program for sequence comparison *(5)*, the FASTA file format is the simplest of the widely used formats available through the INSDC. It is composed of a definition or description line followed by the sequence. The definition line begins with a greater-than symbol (>) and marks the beginning of each new entry. The information following the greater-than symbol varies according to its source. Generally, an identifier follows (**Table 1.1**), after which optional description words may be included. If the sequence is retrieved through NCBI's databases, a GI number precedes the identifier. Though it is recommended that the definition line be no greater than 80 characters, various types and levels of information are often included. The definition line is followed by the DNA sequence itself, in single or multi-line format. Nucleotides are represented by their standard IUB/IUPAC codes, including ambiguity codes (**Table 1.2**).

Table 1.1
FASTA File sequence identifiers. Information from the NCBI Handbook *(25)*

Database name	Identifier syntax
GenBank	gb|*accession.version*
EMBL	emb|*accession.version*
DDBJ	dbj|*accession.version*
NCBI RefSeq	ref|*accession.version*
PDB	pdb|*entry*|*chain*
Patents	pat|*country*|*number*
NBRF PIR	pir||*entry*
SWISS-PROT	sp|*accession*|*entry*
Protein Research Foundation	prf|*name*
GenInfo Backbone Id	bbs|*number*
General database identifier	gnl|*database*|*identifier*
Local Sequence identifier	lcl|*identifier*

Table 1.2
IUB/IUPAC nucleotide and ambiguity codes

A	**a**denosine	M	A or C (a**m**ino)	V	A, C, or G
C	**c**ytidine	K	G or T (**k**eto)	H	A, C, or T
G	**g**uanine	R	A or G (pu**r**ine)	D	A, G, or T
T	**t**hymidine	Y	C or T (p**y**rimidine)	B	C, G, or T
U	**u**ridine	S	A or T (**s**trong)	–	Gap of indeterminate length
		W	C or G (**w**eak)	N	A, C, G, or T (a**n**y or unk**n**own)

2.1.2. Flat File Format

GenBank, EMBL, and DDBJ each have their own flat file format, but contain basically the same information. They are all based upon the Feature Table, which can be found at http://www.ncbi.nlm.nih.gov/collab/FT. For references to these file types, see *(6–9)*.

2.1.3. Accession Numbers, Version Numbers, Locus Names, Database Identifiers, etc.

The standard for identifying a nucleotide sequence record is by an *accession.version* system where the *accession number* is an identifier of two letters followed by six digits and the *version* is an incremental number indicating the number of changes that have been made to the sequence since it was first submitted. Locus names (*see* **Note 1**) are older, less standardized identifiers whose original purpose was to group entries with similar sequences *(10)*. The original locus format was intended to hold information about the organism and other common group characteristics (such as gene product). That ten-character format is no longer able to hold such information for the large number and variety of sequences now available, so the locus has become yet another unique identifier often set to be the same value as the accession number. Database identifiers are simply two- or three-character strings that serve to indicate which database originally received and stored the information. The database identifier is the first value listed in the FASTA identifier syntax (**Table 1.1**).

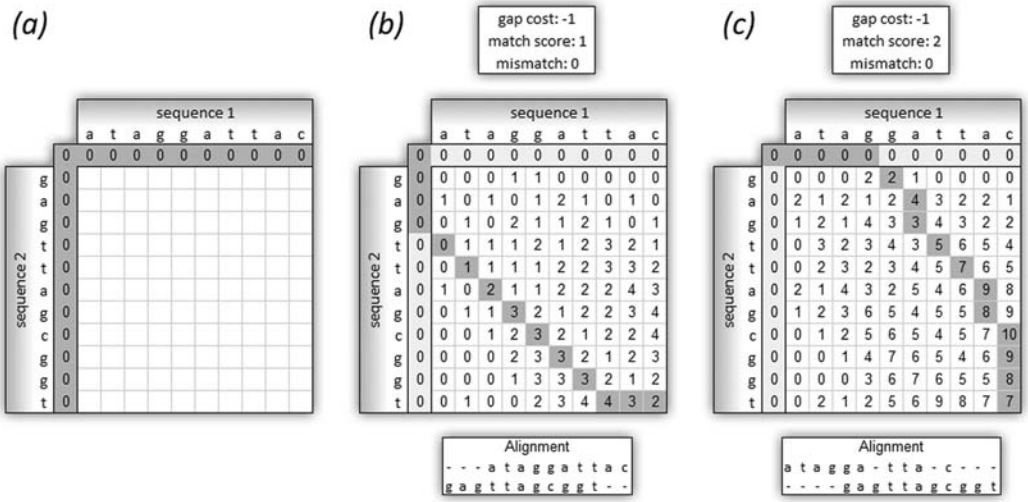
When a sequence is first submitted to GenBank, it is submitted with several defined features associated with the sequence. Some include CDS (coding sequence), RBS (ribosome binding site), rep_origin (origin of replication), and tRNA (mature transfer RNA) information. A translation of protein coding nucleotide sequences into amino acids is provided as part of the features section. Likewise, labeling of different open reading frames, introns, etc., are all part of the table of features. A list of features and their descriptions, formats, and conventions that were agreed upon by INSDC can be found in the Feature Table (*see* **Section 2.1.2**).

2.2. Smith–Waterman and Dynamic Programming

In 1970, Needleman and Wunsch adapted the idea of dynamic programming to the difficult problem of global sequence alignment *(11)*. In 1981, Smith and Waterman adapted this algorithm to local alignments *(12)*. A global alignment attempts to align two sequences throughout their entire length, whereas a local alignment aligns regions of two sequences where high similarity is observed. Both methods involve initializing, scoring, and tracing a matrix where the rows and columns correspond to the bases or residues of the two sequences being aligned (**Fig. 1.2**). In the local alignment case, the first row and the first column are filled with zeroes. The remaining cells are filled with a metric value recursively derived from neighboring values:

$$\max\begin{cases} 0 \\ \text{left neighbor } + \text{gap penalty} \\ \text{top neighbor } + \text{ gap penatly} \\ \text{top-left neighbor} + \text{ match/mismatch score} \end{cases}$$

If the current cell corresponds to a match (identical bases), the match score is added to the value from the diagonal neighbor, otherwise the mismatch score is used. The gap penalty and mismatch scores are generally zero or a small, negative number while the match score is a positive number, larger in magnitude. This


(a)
(b)
(c)
sequence 1
sequence 2

(a)

		a	t	a	g	g	a	t	t	a	c
	0	0	0	0	0	0	0	0	0	0	0
g	0										
a	0										
g	0										
t	0										
t	0										
a	0										
g	0										
c	0										
g	0										
g	0										
t	0										

(b)

gap cost: -1
match score: 1
mismatch: 0

		a	t	a	g	g	a	t	t	a	c
	0	0	0	0	0	0	0	0	0	0	0
g	0	0	0	0	1	1	0	0	0	0	0
a	0	1	0	1	0	1	2	1	0	1	0
g	0	0	1	0	2	1	1	2	1	0	1
t	0	0	1	1	1	2	1	2	3	2	1
t	0	0	1	1	1	1	2	2	3	3	2
a	0	1	0	2	1	1	2	2	2	4	3
g	0	0	1	1	3	2	1	2	2	3	4
c	0	0	0	1	2	3	2	1	2	2	4
g	0	0	0	0	2	3	3	2	1	2	3
g	0	0	0	0	1	3	3	3	2	1	2
t	0	0	1	0	0	2	3	4	4	3	2

Alignment
- - - a t a g g a t t a c
g a g t t a g c g g t - -

(c)

gap cost: -1
match score: 2
mismatch: 0

		a	t	a	g	g	a	t	t	a	c
	0	0	0	0	0	0	0	0	0	0	0
g	0	0	0	0	2	2	1	0	0	0	0
a	0	2	1	2	1	2	4	3	2	2	1
g	0	1	2	1	4	3	3	4	3	2	2
t	0	0	3	2	3	4	3	5	6	5	4
t	0	0	2	3	2	3	4	5	7	6	5
a	0	2	1	4	3	2	5	4	6	9	8
g	0	1	2	3	6	5	4	5	5	8	9
c	0	0	1	2	5	6	5	4	5	7	10
g	0	0	0	1	4	7	6	5	4	6	9
g	0	0	0	0	3	6	7	6	5	5	8
t	0	0	2	1	2	5	6	9	8	7	7

Alignment
a t a g g a - t t a - c - - -
- - - - g a g t t a g c g g t

Fig. 1.2. Smith-Waterman local alignment example. (**A**) shows an empty matrix, initialized for a Smith-Waterman alignment. (**B**) and (**C**) are alignments calculated using the specified scoring parameters. The alignment produced in (**B**) is drastically different from that in (**C**), though they only differ slightly in their scoring parameters, one using a match score of 1 and the other 2.

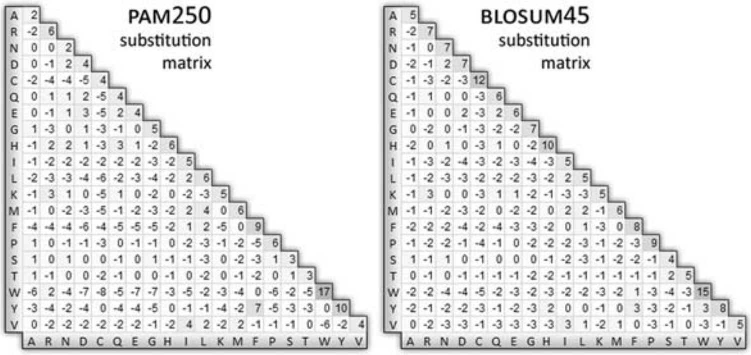
method is used recursively, starting from the upper left corner of the matrix and proceeding to the lower right corner. **Figure 1.2b** and **c** shows matrices from two different sets of gap and match scores.

To find a local alignment, one simply finds the largest number in the matrix and traces a path back until a zero is reached, each step moving to a cell that was responsible for the current cell's value. While this method is robust and is guaranteed to give the best alignment(s) for a given set of scores and penalties, it is important to note that often multiple paths and therefore multiple alignments are possible for any given matrix when these parameters are used. As an example, **b** and **c** of **Fig. 1.2** only differ slightly in their gap and match scores, but produce very different alignments. In addition, the set of scores and penalties used dramatically affect the alignment, and finding the optimal set is neither trivial nor deterministic. Weight matrices for protein-coding sequences were developed in the late 1970s in an attempt to overcome these challenges.

2.3. Weighting/Models

2.3.1. PAM Matrices

To increase the specificity of alignment algorithms and provide a means to evaluate their statistical significance, it was necessary to implement a meaningful scoring scheme for nucleotide and amino acid substitutions. This was especially true when dealing with protein (or protein-coding) sequences. In 1978, Dayhoff et al. developed the first scoring or weighting matrices created from substitutions that have been observed during evolutionary history *(13)*. These substitutions, since they have been allowed or accepted by natural



PAM250 substitution matrix

A	2																			
R	-2	6																		
N	0	0	2																	
D	0	-1	2	4																
C	-2	-4	-4	-5	4															
Q	0	1	1	2	-5	4														
E	0	-1	1	3	-5	2	4													
G	1	-3	0	1	-3	-1	0	5												
H	-1	2	2	1	-3	3	1	-2	6											
I	-1	-2	-2	-2	-2	-2	-2	-3	-2	5										
L	-2	-3	-3	-4	-6	-2	-3	-4	-2	2	6									
K	-1	3	1	0	-5	1	0	-2	0	-2	-3	5								
M	-1	0	-2	-3	-5	-1	-2	-3	-2	2	4	0	6							
F	-4	-4	-4	-6	-4	-5	-5	-5	-2	1	2	-5	0	9						
P	1	0	-1	-1	-3	0	-1	-1	0	-2	-3	-1	-2	-5	6					
S	1	0	1	0	0	-1	0	1	-1	-1	-3	0	-2	-3	1	3				
T	1	-1	0	0	-2	-1	0	0	-1	0	-2	0	-1	-2	0	1	3			
W	-6	2	-4	-7	-8	-5	-7	-7	-3	-5	-2	-3	-4	0	-6	-2	-5	17		
Y	-3	-4	-2	-4	0	-4	-4	-5	0	-1	-1	-4	-2	7	-5	-3	-3	0	10	
V	0	-2	-2	-2	-2	-2	-2	-1	-2	4	2	-2	2	-1	-1	-1	0	-6	-2	4
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V

BLOSUM45 substitution matrix

A	5																			
R	-2	7																		
N	-1	0	7																	
D	-2	-1	2	7																
C	-1	-3	-2	-3	12															
Q	-1	1	0	0	-3	6														
E	-1	0	0	2	-3	2	6													
G	0	-2	0	-1	-3	-2	-2	7												
H	-2	0	1	0	-3	1	0	-2	10											
I	-1	-3	-2	-4	-3	-2	-3	-4	-3	5										
L	-1	-2	-3	-3	-2	-2	-2	-3	-2	2	5									
K	-1	3	0	0	-3	1	1	-2	-1	-3	-3	5								
M	-1	-1	-2	-3	-2	0	-2	-2	0	2	2	-1	6							
F	-2	-2	-2	-4	-2	-4	-3	-3	-2	0	1	-3	0	8						
P	-1	-2	-2	-1	-4	-1	0	-2	-2	-2	-3	-1	-2	-3	9					
S	1	-1	1	0	-1	0	0	0	-1	-2	-3	-1	-2	-2	-1	4				
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-1	-1	2	5			
W	-2	-2	-4	-4	-5	-2	-3	-2	-3	-2	-2	-3	-2	1	-3	-4	-3	15		
Y	-2	-1	-2	-2	-3	-1	-2	-3	2	0	0	-1	0	3	-3	-2	-1	3	8	
V	0	-2	-3	-3	-1	-3	-3	-3	-3	3	1	-2	1	0	-3	-1	0	-3	-1	5
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V

Fig. 1.3. PAM250 and BLOSUM45 substitution matrices.

selection, are called accepted point mutations (PAM). For Dayhoff's PAM matrices, groups of proteins with 85% or more sequence similarity were analyzed and their 1,571 substitutions were cataloged. Each cell of a PAM matrix corresponds to the frequency in substitutions per 100 residues between two given amino acids. This frequency is referred to as one PAM unit. Back in the 1970s, when they were created, however, there was a limited number and variety of protein sequences available, so they are biased toward small, globular proteins. It is also important to note that each PAM matrix corresponds to a specific evolutionary distance and that each is simply an extrapolation of the original. For example, a PAM250 (**Fig. 1.3**) matrix is constructed by multiplying the PAM1 matrix by itself 250 times and is viewed as a typical scoring matrix for proteins that have been separated by 250 million years of evolution.

2.3.2. BLOSUM Matrices

To overcome some of the drawbacks of PAM matrices, Henikoff and Henikoff developed the BLOSUM matrices in 1992 *(14)*. These matrices were based on the BLOCKS database, which organizes proteins into blocks, where each block, defined by an alignment of motifs, corresponds to a family. Whereas the original PAM matrix was calculated with proteins with at least 85% identity, BLOSUM matrices are each calculated separately using conserved motifs at or below a specific evolutionary distance. This diversity of matrices coupled with being based on larger datasets makes the BLOSUM matrices more robust at detecting similarity at greater evolutionary distances and more accurate, in many cases, at performing local similarity searches *(15)*.

2.3.3. Choosing a Matrix

When choosing a matrix, it is important to consider the alternatives. Do not simply choose the default setting without some initial consideration. In general, finding similarity at increasing divergence corresponds to increasing PAM matrices (PAM1, PAM40, PAM120, etc.) and decreasing BLOSUM matrices (BLOSUM90,

Table 1.3
Suggested uses for common substitution matrices. The matrices highlighted in bold are available through NCBI's BLAST web interface. BLOSUM62 has been shown to provide the best results in BLAST searches overall due to its ability to detect large ranges of similarity. Nevertheless, the other matrices have their strengths. For example, if your goal is to only detect sequences of high similarity to infer homology within a species, the PAM30, BLOSUM90, and PAM70 matrices would provide the best results. This table was adapted from results obtained by David Wheeler *(16)*

Alignment size	Best at detecting	Similarity (%)	PAM	BLOSUM
Short	Similarity within a **species**	75–90	**PAM30**	BLOSUM95
"	Similarity within a **genus**	60–75	**PAM70**	BLOSUM85
Medium	Similarity within a **family**	50–60	PAM120	**BLOSUM80**
"	The **largest range** of similarity	40–50	PAM160	**BLOSUM62**
Long	Similarity within a **class**	30–40	PAM250	**BLOSUM45**
"	Similarity within the **twilight zone**	20–30		BLOSUM30

BLOSUM80, BLOSUM62, etc.) *(16)*. PAM matrices are strong at detecting high similarity due to their use of evolutionary information. However, as evolutionary distance increases, BLOSUM matrices are more sensitive and accurate than their PAM counterparts. **Table 1.3** includes a list of suggested uses.

2.4. BLAST Programs

Nucleotide–nucleotide searches are beneficial because no information is lost in the alignment. When a codon is translated from nucleotides to amino acid, approximately 69% of the complexity is lost (64 possible nucleotide combinations mapped to 20 amino acids). In contrast, however, the true physical relationship between two coding sequences is best captured in the translated view. Matrices that take into account physical properties, such as PAM and BLOSUM, can be used to add power to the search. Additionally, in a nucleotide search, there are only four possible character states compared to 20 in an amino acid search. Thus the probability of a match due to chance versus a match due to common ancestry (identify in state versus identical by descent) is high.

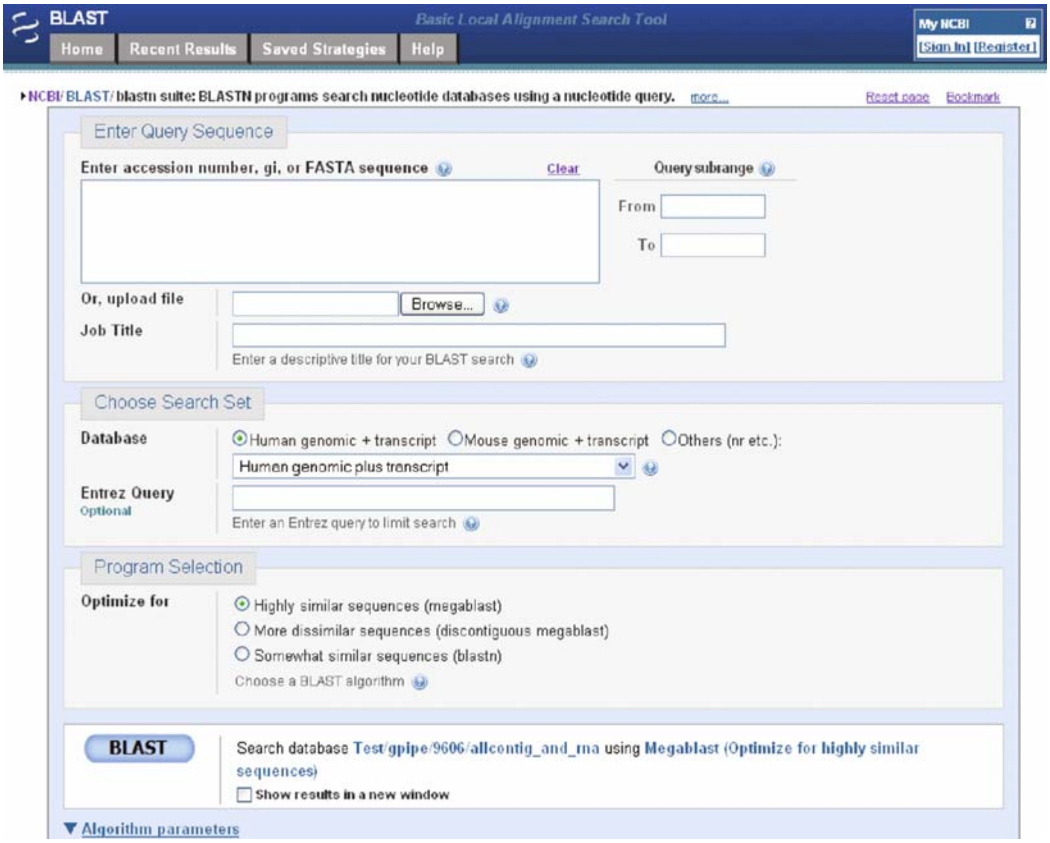
The Basic Local Alignment and Search Tools (BLAST) are the most widely used and among the most accurate in detecting sequence similarity *(17)*(*see* **Note 2**). The standard BLAST programs are Nucleotide BLAST (blastn), Protein BLAST (blastp), blastx, tblastn, and tblastx. Others have also been developed to meet specific needs. When choosing a BLAST program, it is important to choose the correct one for your question of interest. Some of the most common mistakes in similarity searching come from misunderstandings of these different applications.

- **Nucleotide blast**: compares a nucleotide query against a nucleotide sequence database
- **Protein blast**: compares a protein query against a protein sequence database
- **blastx**: compares a nucleotide query translated in all six reading frames against a protein database
- **tblastn**: compares a protein query against a nucleotide sequence database dynamically translated in all six reading frames
- **tblastx**: compares a nucleotide query in all six reading frames against a nucleotide sequence database in all six reading frames

The BLAST algorithm is an heuristic program, one that is not guaranteed to return the best result. It is, however, quite accurate. BLAST works by first making a look-up table of all the "words" and "neighboring words" of the query sequence. Words are short subsequences of length *W* and neighboring words are words that are highly accepted in the scoring matrix sense, determined by a threshold *T*. The database is then scanned for the words and neighboring words. Once a match is found, extensions with and without gaps are initiated there both upstream and downstream. The extension continues, adding gap existence (initiation) and extension penalties, and match and mismatch scores as appropriate as in the Smith-Waterman algorithm until a score threshold *S* is reached. Reaching this mark flags the sequence for output. The extension then continues until the score drops by a value *X* from the maximum, at which point the extension stops and the alignment is trimmed back to the point where the maximum score was hit. Understanding this algorithm is important for users if they are to select optimal parameters for BLAST. The interaction between the parameters *T*, *W*, *S*, *X*, and the scoring matrix allows the user to find a balance between sensitivity and specificity, alter the running time, and tweak the accuracy of the algorithm. The interactions among these variables will be discussed in **Section 2.8**.

2.5. Query Sequence

Query sequences may be entered by uploading a file or entering one manually in the text box provided (**Fig. 1.4**). The upload option accepts files containing a single sequence, multiple sequences in FASTA format, or a list of valid sequence identifiers (accession numbers, GI numbers, etc.). In contrast to previous versions of BLAST on the NCBI website, the current version allows the user to specify a descriptive job title. This allows the user to track any adjustments or versions of a search as well as its purpose and query information. This is especially important when sequence identifiers are not included in the uploaded file.



▸ NCBI/ BLAST/ blastn suite: BLASTN programs search nucleotide databases using a nucleotide query. more...

Enter Query Sequence

Enter accession number, gi, or FASTA sequence

Query subrange

From

To

Or, upload file

Job Title

Enter a descriptive title for your BLAST search

Choose Search Set

Database: ● Human genomic + transcript ○ Mouse genomic + transcript ○ Others (nr etc.):

Human genomic plus transcript

Entrez Query Optional

Enter an Entrez query to limit search

Program Selection

Optimize for

● Highly similar sequences (megablast)

○ More dissimilar sequences (discontiguous megablast)

○ Somewhat similar sequences (blastn)

Choose a BLAST algorithm

Search database Test/gpipe/9606/allcontig_and_rna using Megablast (Optimize for highly similar sequences)

□ Show results in a new window

Fig. 1.4. NCBI nucleotide BLAST interface.

2.6. Search Set

2.6.1. Databases

When choosing a database, it is important to understand their purpose, content, and limitations. The list of nucleotide databases is divided into *Genomic plus Transcript* and *Other Databases* sections. Some of the databases, composed of reference sequences, come from the RefSeq database, a highly curated, all-inclusive, non-redundant set of INSDC (EMBL + GenBank + DDBJ) DNA, mRNA, and protein entries. RefSeq sequences have accession numbers of the form AA_######, where AA is one of the following combination of letters (**Table 1.4**) and ###### is a unique number representing the sequence.

A description of the nucleotide databases is included below. A list of protein databases accessible through BLAST's web interface can be found at http://www.ncbi.nlm.nih.gov/BLAST/blastcgihelp.shtml.

- **Human genomic plus transcript**: contains all human genomic and RNA sequences.
- **Mouse genomic plus transcript**: contains all mouse genomic and RNA sequences.

Table 1.4
RefSeq categories

Experimentally determined and curated		Genome annotation (computational predictions from DNA)	
NC	**C**omplete genomic molecules		
NG	Incomplete **g**enomic region		
NM	**m**RNA	XM	Model **m**RNA
NR	**R**NA (non-coding)		
NP	**P**rotein	XP	Model **p**rotein

- **Nucleotide collection (nr/nt)**: contains INSDC + RefSeq nucleotides + PDB sequences, not including EST, STS, GSS, or unfinished HGT sequences. The nucleotide collection is the most comprehensive set of nucleotide sequences available through BLAST.
- **Reference mRNA sequences (refseq_rna)**: contains the non-redundant RefSeq mRNA sequences.
- **Reference genomic sequences (refseq_genomic)**: contains the non-redundant RefSeq genomic sequences.
- **Expressed sequence tags (est)**: contains short, single reads from mRNA sequencing (via cDNA). These cDNA sequences represent the mRNA in a cell at a particular moment in a particular tissue.
- **Non-human, non-mouse ESTs (est_others)**: the previous database with human and mouse sequences removed.
- **Genomic survey sequences (gss)**: contains random genomic sequences obtained from single-pass genome surveys, cosmids, BACs, YACs, and other survey methods. Their quality varies.
- **High-throughput genomic sequences (HTGS)**: contains sequences obtained from high-throughput genome centers. Sequences in this database contain a phase number, 0 being the initial phase and 3 being the finished phase. Once finished, the sequences move to the appropriate division in their respective database.
- **Patent sequences (pat)**: contains sequences from the patent offices at each of the INSDC organizations.
- **Protein data bank (pdb)**: the nucleotide sequences from the Brookhaven Protein Data Bank managed by the Research Collaboratory for Structural Bioinformatics (http://www.rcsb.org/pdb).

- **Human ALU repeat elements (alu_repeats)**: contains a set of ALU repeat elements that can be used to mask repeat elements from query sequences. ALU sequences are regions subject to cleavage by Alu restriction endonucleases, around 300 bp long, and estimated to constitute about 10% of the human genome *(18)*.
- **Sequence tagged sites (dbsts)**: a collection of unique sequences used in PCR and genome mapping that identify a particular region of a genome.
- **Whole-genome shotgun reads (wgs)**: contains large-scale shotgun sequences, mostly unassembled and non-annotated.
- **Environmental samples (env_nt)**: contains sets of whole-genome shotgun reads from many sampled organisms, each set from a particular location of interest. These sets allow researchers to look into the genetic diversity existing at a particular location and environment.

2.6.2. Organism

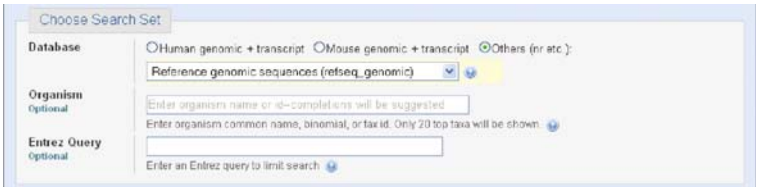
The organism box allows the user to specify a particular organism to search. It automatically suggests organisms when you begin typing. This option is not available when Genomic plus Transcript databases are selected (**Fig. 1.5**).

2.6.3. Entrez Queries

Entrez queries provide a way to limit your search to a specific type of organism or molecule. It is an efficient way to filter unwanted results by excluding organisms or defining sequence length criteria. In addition, Entrez queries allow the user to find sequences submitted by a particular author, from a particular journal, with a particular property or feature key, or submitted or modified within a specific date range. For help with Entrez queries, see the Entrez Help document at http://www.ncbi.nlm.nih.gov/entrez/query/static/help/helpdoc.html.

2.7. BLAST Search Parameters

In addition to entering a query sequence, choosing a search set, and selecting a program, several additional parameters are available, which allow you to fine-tune your search to your needs. These parameters are available by clicking the "Algorithm parameters" link at the bottom of the BLAST page (**Fig. 1.6**) (*see* **Notes 3 and 4**).



Choose Search Set

Database: ○Human genomic + transcript ○Mouse genomic + transcript ●Others (nr etc.):

Reference genomic sequences (refseq_genomic)

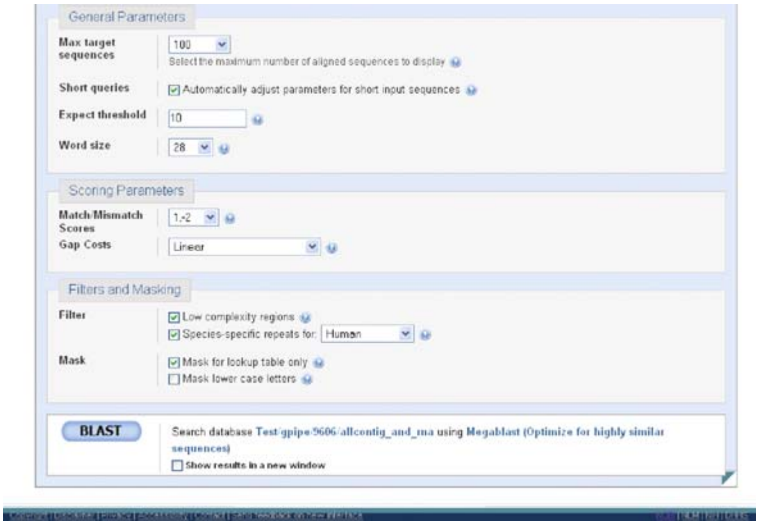
Organism Optional: Enter organism name or id–completions will be suggested

Enter organism common name, binomial, or tax id. Only 20 top taxa will be shown.

Entrez Query Optional:

Enter an Entrez query to limit search

Fig. 1.5. NCBI nucleotide BLAST algorithm parameters.



General Parameters

Max target sequences: 100 ▾
Select the maximum number of aligned sequences to display

Short queries: ☑ Automatically adjust parameters for short input sequences

Expect threshold: 10

Word size: 28 ▾

Scoring Parameters

Match/Mismatch Scores: 1,-2 ▾

Gap Costs: Linear ▾

Filters and Masking

Filter: ☑ Low complexity regions
☑ Species-specific repeats for: Human ▾

Mask: ☑ Mask for lookup table only
□ Mask lower case letters

Search database Test/gpipe/9606/allcontig_and_rna using Megablast (Optimize for highly similar sequences)

□ Show results in a new window

Fig. 1.6. Organism selection when searching a multi-organism database.

2.7.1. Max Target Sequences

The maximum target sequences parameter allows you to select the number of sequences you would like displayed in your results. Lower numbers do not reduce the search time, but do reduce the time to send the results back. This is generally only an issue over a slow connection.

2.7.2. Short Queries

When using short queries (of length 30 or less), the parameters must be adjusted or you will not receive statistically significant results. Checking the “short queries” box automatically adjusts the parameters to return valid responses for a short query sequence.

2.7.3. Expect Threshold

The expect threshold limits the results displayed to those with an *E*-value lower than it. This value corresponds to the number of sequence matches that are expected to be found merely by chance.

2.7.4. Word Size

The word size, *W*, as discussed earlier determines the length of the words and neighboring words used as initial search queries. Increasing the word size generally results in fewer extension initializations, increasing the speed of the BLAST search but decreasing its sensitivity.

2.7.5. Scoring Parameters

The scoring parameters of a nucleotide search are the match and mismatch scores and gap costs. In protein searches, the match and mismatch scores are indicated by a scoring matrix (*see* **Section 2.3**). A limited set of suggested match and mismatch scores are available from the dropdown menu on NCBI’s BLAST search form. Increasing the ratio in the following fashion (match, mismatch): (1,–1) → (4,–5) → (2,–3) → (1,–2) → (1,–3) → (1,–4) prevents mismatched nucleotides from aligning, increasing the

Table 1.5
Suggested scoring parameters for nucleotide–nucleotideBLAST searches. When performing a nucleotide–nucleotide BLAST search, these general guidelines may be used to choose a match/mismatch score based upon the degree of conservation you expect to see in your results. If you are searching for sequences with a high degree of similarity (i.e., within a species), the default parameters of (match +1, mismatch –2) would be appropriate. If, however, you are searching for sequences between very distant organisms (a worm and a mouse, for example), a smaller ratio would be more appropriate (for example, –1). Information provided by NCBI *(26)*

Match/mismatch ratio	Similarity (%)
0.33 (1/–3)	99
–0.5 (1/–2)	95
–1 (1/–1)	75

number of gaps, but decreasing mismatches. The greater divergence you expect in sequences you are looking for, the larger the ratio you should choose. NCBI has provided the guidelines found in **Table 1.5**. Additionally, decreasing the gap existence and extension penalties will increase gap incidence.

2.7.6. Filters

The low complexity regions filter removes regions of the sequence with low complexity, preventing those segments from producing statistically significant but uninformative results. The DUST program by Tatusov and Lipman (unpublished) is used for nucleotide BLAST searches. Often, when a search takes much longer than expected, the query contains a low-complexity region that is being matched with many similar but unrelated sequences. It is important to note, however, that turning this filter on may remove some interesting and informative matches from the results. In nucleotide searches, it is also possible to remove species-specific repeats by checking the "Species-specific repeats for:" box and selecting the appropriate species. This prevents repeats that are common in a particular species from producing false-positives with other parts of its own or closely related genomes.

2.7.7. Masks

The "Mask for lookup table only" option allows the user to mask the low-complexity regions (regions of biased composition including homopolymeric runs, short-period repeats, etc.) during the seeding stage, where words and neighboring words are scanned, but unmask them during the extension phases. This prevents the *E*-values from being affected in biologically interesting results while preventing regions of low complexity from slowing the search down and introducing uninteresting results.

The "Mask lower case letters" option gives the user the option to annotate his or her sequence by using lower case letters where masking is desired.

2.8. Interpreting the Results

By default, BLAST results contain five basic sections: a summary of your input (query and parameters), a graphical overview of the top results, a table of sequences producing significant alignments, the best 100 alignments, and result statistics. The number of hits shown in the graphical overview as well as the number of alignments, among other options, may be changed by clicking "Reformat these results" at the top of the results page or by clicking "Formatting options" on the Formatting Results page (the page that appears after you click BLAST and before the results appear).

In the third section, the results table contains eight columns: accession, description, max score, total score, query coverage, E-value, max ident, and links. The *Accession* number provides a link to detailed information about the sequence. The *description* provides information about the species and the kind of sample the hit was generated from. The *max score* provides a metric for how good the best local alignment is. The *total score* indicates how similar the sequence is to the query, accounting for all local alignments between the two sequences. If the max score is greater than the total score, then more than one local alignment was found between the two sequences. Higher scores are correlated with more similar sequences. Both of these scores, reported in bits, are calculated from a formula that takes into account matches (or similar residues, if doing a protein search) and mismatch penalties along with gap insertion penalties. Bit scores are normalized so that they can be directly compared even though the alignments between different sequences may be of different lengths. The expectation value or *E-value* provides an estimate of how likely it is that this alignment occurred by random chance. An E-value of 2e–02 indicates that similarity found in the alignment has a 2 in 100 chance of occurring by chance. The lower the E-value, the more significant the score. An appropriate cutoff E-value depends on the users' goals. The *max identity* field shows the percentage of the query sequence that was identical to the database hit. The *links* field provides links to UniGene, the Gene Expression Omnibus, Entrez Gene, Entrez's Related Structures (for protein sequences), and the Map Viewer (for genomic sequences).

2.9. Future of Similarity Searching

Since both PAM and BLOSUM matrices are experimentally derived from a limited set of sequences in a database that was available at the time they were created, they will almost certainly not provide optimal values for searches with new sequence families. Current research is being performed to determine which chemical properties are changing in a sequence in order to provide a magnitude of change that is independent of scoring matrices.

Current techniques to find promoter regions are severely lacking in accuracy *(19)*. Techniques will arise in the future that may improve current methods by using BLAST-like algorithms to assess the similarity of a sequence to known promoter elements, thus helping to identify it as a promoter.

3. Examples

This section will provide three examples of common BLAST uses: a nucleotide–nucleotide BLAST, a position-specific iterated BLAST, and a blastx.

3.1. Nucleotide–Nucleotide BLAST for Allele Finding

Here we present an example of using BLAST to search for the known alleles of a given nucleotide sequence. This approach can be used to answer the question: what are the known variants of my gene of interest (within its species)? Our example will be to find all known variants of a Tp53 nucleotide sequence (accession number AF151353) from a mouse. While this sequence does code for a protein, non-coding sequences would work just as well using this approach.

We will start by going to the BLAST homepage at http://www.ncbi.nlm.nih.gov/BLAST/ and selecting *nucleotide blast*. In the “Enter Query Sequence” box, we type the accession number: AF151353. You will notice that the “Job Title” box automatically fills in a title for you “AF151353:*Mus musculus* tumor suppressor p53...”. If we were to paste a sequence instead of an accession number or GI, we would want to enter a job title to help us keep track of our results. Under “Choose Search Set,” we select the “Nucleotide collection (nr/nt)” database, since it is the most comprehensive database (remember that nr is no longer non-redundant). For a complete search, we should also perform a search on the “Expressed sequence tags (est)” database. In the Organism box, we choose type “mouse” and select “mouse (taxid:10090),” which corresponds to *Mus musculus*, the house mouse. Since we are searching for alleles, we select “Highly similar sequences (megablast)” in the “Program Selection” box.

Next, let us change the algorithm parameters. Click “Algorithm parameters” to display them. Since the sequence is 1,409 bp in length, we deselect the “Automatically adjust parameters for short input sequences” box. Since we expect that the p53 protein is a well-conserved protein (due to its critical function), we set the expect threshold to a low value. Let us choose 1e-8. For a word size, we are not concerned about speed in this case, so the number of extensions performed is not a concern. Let us select a word size of 20 to make sure we do not miss any matches (although in this case a larger word size should not make much difference). As for the scoring parameters, we choose the largest ratio, corresponding to the greatest identity: “1,–4.” Since this is a protein-coding sequence, we do not expect repeats to be a factor, so we leave the Filters and Masking section at the default settings.

The results indicate that 108 hits were found on the query sequence. Looking at the graphical alignment (**Fig. 1.7**), we notice that only about 2/3 of them span a good portion of the query. When we scroll down to the gene descriptions, most of the last fourth are pseudogenes (partial sequence) (**Fig. 1.8**), which may offer insight into different alleles and their corresponding phenotypes, but which were not sequenced experimentally. Performing a search on the EST database with the same parameters results in 101 additional hits.

3.2. PSI-BLAST for Distant Homology Searching

When searching for distantly related sequences, two BLAST options are available. One is the standard nucleotide–nucleotide BLAST with discontiguous BLAST, a method very similar to Ma

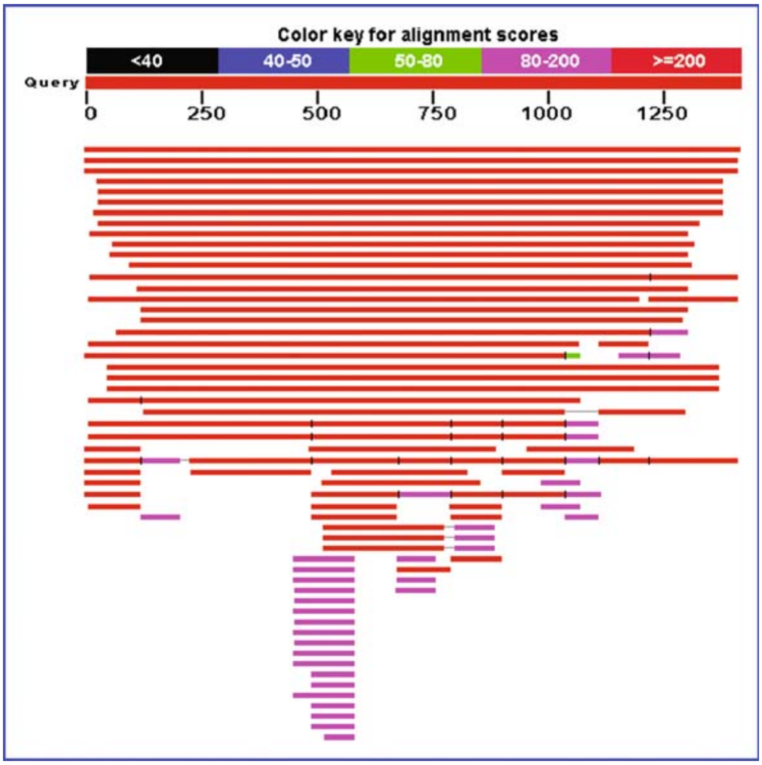
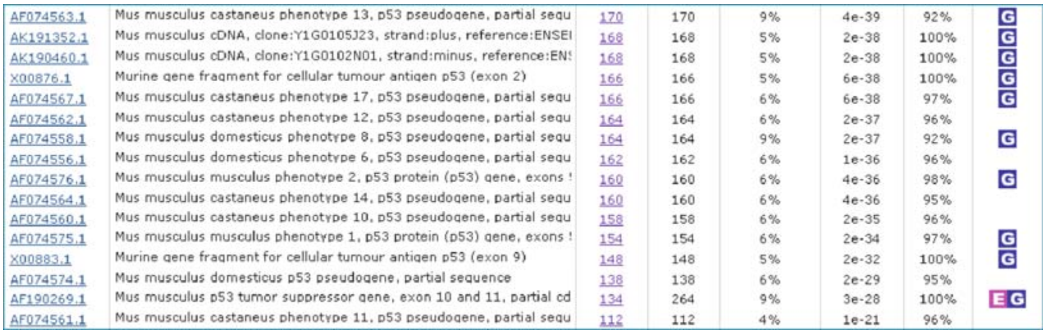


Fig. 1.7. Graphical distribution of top 100 BLAST hits.



AF074563.1	Mus musculus castaneus phenotype 13, p53 pseudogene, partial sequ	170	170	9%	4e-39	92%	G
AK191352.1	Mus musculus cDNA, clone:Y1G0105J23, strand:plus, reference:ENSEI	168	168	5%	2e-38	100%	G
AK190460.1	Mus musculus cDNA, clone:Y1G0102N01, strand:minus, reference:EN!	168	168	5%	2e-38	100%	G
X00876.1	Murine gene fragment for cellular tumour antigen p53 (exon 2)	166	166	5%	6e-38	100%	G
AF074567.1	Mus musculus castaneus phenotype 17, p53 pseudogene, partial sequ	166	166	6%	6e-38	97%	G
AF074562.1	Mus musculus castaneus phenotype 12, p53 pseudogene, partial sequ	164	164	6%	2e-37	96%	
AF074558.1	Mus musculus domesticus phenotype 8, p53 pseudogene, partial sequ	164	164	9%	2e-37	92%	G
AF074556.1	Mus musculus domesticus phenotype 6, p53 pseudogene, partial sequ	162	162	6%	1e-36	96%	
AF074576.1	Mus musculus musculus phenotype 2, p53 protein (p53) gene, exons !	160	160	6%	4e-36	98%	G
AF074564.1	Mus musculus castaneus phenotype 14, p53 pseudogene, partial sequ	160	160	6%	4e-36	95%	
AF074560.1	Mus musculus castaneus phenotype 10, p53 pseudogene, partial sequ	158	158	6%	2e-35	96%	
AF074575.1	Mus musculus musculus phenotype 1, p53 protein (p53) gene, exons !	154	154	6%	2e-34	97%	G
X00883.1	Murine gene fragment for cellular tumour antigen p53 (exon 9)	148	148	5%	2e-32	100%	G
AF074574.1	Mus musculus domesticus p53 pseudogene, partial sequence	138	138	6%	2e-29	95%	
AF190269.1	Mus musculus p53 tumor suppressor gene, exon 10 and 11, partial cd	134	264	9%	3e-28	100%	E G
AF074561.1	Mus musculus castaneus phenotype 11, p53 pseudogene, partial sequ	112	112	4%	1e-21	96%	

Fig. 1.8. Last 16 sequences producing significant alignments from a mouse p53 gene Nucleotide BLAST search. Nineteen of the last 26 reported sequences are pseudogenes.

et al.'s work *(20)*, selected as the program. The other is to use a more sensitive approach, PSI-BLAST, which performs an iterative search on a protein sequence query. Though the second approach will only work if you are dealing with protein-coding sequences, it is more sensitive and accurate than the first.

In this example, we will search for relatives of the cytochrome *b* gene of the Durango night lizard (*Xantusia extorris*). We start by selecting *protein blast* from the BLAST home page and entering the accession number, ABY48155, into the query box. If your sequence is not available as a protein sequence, you will need to translate it. This can easily be done using a program such as MEGA *(21)*, available at http://www.megasoftware.net, or an online tool such as the JustBio Translator (http://www.justbio.com/translator/) or the ExPASy Translate Tool (http://www.expasy.org/tools/dna.html).

Once again, the "Job Title" box is filled with "ABY48155: cytochrome b [*Xantusia extorris*]." We will choose the "Reference proteins (refseq_protein)" database, which is more highly curated and non-redundant (per gene) than the default nr database. We do not specify an organism because we want results from any and all related organisms. For the algorithm, we select PSI-BLAST due to its ability to detect more distantly related sequences. We hope to include as many sequences as possible in our iterations, so we choose 1,000 as the max target sequences. We can, once again, remove the "Automatically adjust parameters for short input sequences" check, since our sequence is sufficiently long (380 amino acids). Since we wish to detect all related sequences, we keep the expect threshold at its default of 10. While decreasing it may remove false-positives, it may also prevent some significant results from being returned. Since we do not have a particular scope in mind (within the genus or family, for example), we will use the BLOSUM62 matrix due to its ability to detect homology over large ranges of similarity.

The first iteration results in 1,000 hits on the query sequence, all of which cover at least 93% of the query sequence and have an E-value of 10^{-126} or less. We leave all of the sequences selected and press the “Run PSI-Blast iteration 2” button. The second iteration likewise returns 1,000 hits, but this time they have E-values less than 10^{-99} and cover at least 65% of the query sequence (all but six cover 90% or more). We uncheck the last hit, Bi4p [*Saccharomyces cerevisiae*], since we are unsure of its homology, and iterate one last time.

At this point, it would be helpful to view the taxonomy report of the results. You can do so by clicking “Taxonomy Reports” near the bottom of the first section of the BLAST report. You will notice that we have a good selection of organisms, ranging from bony fishes to Proteobacteria. While this list would need to be narrowed to produce a good taxonomy, it would be a good starting point if you wish to perform a broad phylogenetic reconstruction. To perform a search of more closely related sequences, you would likely perform a standard blastp (protein–protein BLAST) instead of a PSI-BLAST and use the PAM 70 or PAM 30 matrix.

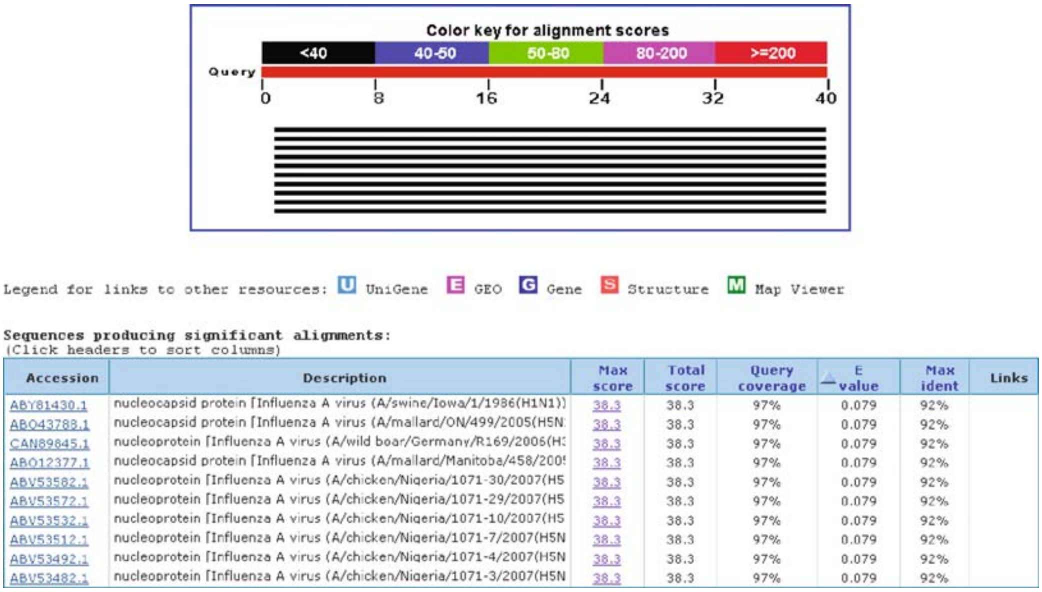
3.3. Blastx for EST Identification

What if you have a nucleotide sequence such as an expressed sequence tag and wish to know if it codes for a known protein? You can search the nucleotide database or take the more direct approach of blastx. Blastx allows you to search the protein database using a nucleotide query, which it first translates into all six reading frames. In this example, we will perform a blastx on the following sequence:

TCTCTATAGTTATGGTGTTCTGAATCAGCCTTCCCTCATA

Since the sequence is only 40 bp long, we need to be careful with our parameters. We start by selecting blastx from the BLAST homepage. We then enter the sequence into the query box and enter a relevant job title, such as “EST blastx Search 1.” We will search the “Non-redundant protein sequences (nr)” database, since it has the largest number of annotated nucleotide sequences. Under “Algorithm parameters,” we need to choose an appropriate expect threshold and matrix. If we choose too low an expect threshold, we might not find anything. Likewise, if we choose the wrong matrix, we may not obtain significant results due to the short length of our sequence. We will choose 10 (the default) as our expect threshold and PAM70 as our matrix, since it corresponds to finding similarity at or below the family/genus level. Since we do not know what our sequence is, we want to filter regions of low complexity to ensure that if our sequence contains such regions, they will not return deceptively significant results.

Our search produces a large number (more than 1,000) of results with an E-value of 0.079 (**Fig. 1.9**). If we were to use the PAM70 matrix, essentially the same results would be obtained, but each with an E-value of 3.0. Since all of the 2,117 results are different entries of the nucleocapsid protein of the Influenza A



Legend for links to other resources: U UniGene E GEO G Gene S Structure M Map Viewer

Sequences producing significant alignments:
(Click headers to sort columns)

Accession	Description	Max score	Total score	Query coverage	E value	Max ident	Links
ABY81430.1	nucleocapsid protein [Influenza A virus (A/swine/Iowa/1/1986(H1N1))	38.3	38.3	97%	0.079	92%	
ABO43788.1	nucleocapsid protein [Influenza A virus (A/mallard/ON/499/2005(H5N	38.3	38.3	97%	0.079	92%	
CAN89845.1	nucleoprotein [Influenza A virus (A/wild boar/Germany/R169/2006(H	38.3	38.3	97%	0.079	92%	
ABO12377.1	nucleocapsid protein [Influenza A virus (A/mallard/Manitoba/458/200	38.3	38.3	97%	0.079	92%	
ABV53582.1	nucleoprotein [Influenza A virus (A/chicken/Nigeria/1071-30/2007(H5	38.3	38.3	97%	0.079	92%	
ABV53572.1	nucleoprotein [Influenza A virus (A/chicken/Nigeria/1071-29/2007(H5	38.3	38.3	97%	0.079	92%	
ABV53532.1	nucleoprotein [Influenza A virus (A/chicken/Nigeria/1071-10/2007(H5	38.3	38.3	97%	0.079	92%	
ABV53512.1	nucleoprotein [Influenza A virus (A/chicken/Nigeria/1071-7/2007(H5N	38.3	38.3	97%	0.079	92%	
ABV53492.1	nucleoprotein [Influenza A virus (A/chicken/Nigeria/1071-4/2007(H5N	38.3	38.3	97%	0.079	92%	
ABV53482.1	nucleoprotein [Influenza A virus (A/chicken/Nigeria/1071-3/2007(H5N	38.3	38.3	97%	0.079	92%	

Fig. 1.9. Blastx results showing E-values of 0.079 for the top ten<10> hits, all of which are nucleocapsid proteins or nucleoproteins.

virus, we can be somewhat confident that our protein is related, especially if we had any prior knowledge that would support our findings.

4. Notes

1. One of the options NCBI provides from their homepage is to search across their databases using an identifier (accession number, sequence identification number, Locus ID, etc.). This option can be rather straightforward if you are using an identifier unique to a particular sequence; however, if you are searching for a locus across organisms or individuals, you may need to pay close attention to the search terms you are using. For example, since the Cytochrome b/b6 subunit is known by the terms “Cytochrome b,” “Cytochrome b6,” “cyt-b,” “cytb,” “cyb,” “COB,” “COB1,” “cyb6,” “petB,” “mtcyb,” and “mt-cyb” in a search for all possible homologs of this subunit, it is necessary to search for all of its names and abbreviations used in the organisms of interest. Since research groups studying different organisms create their own unique locus names for the same gene, it is important to use all of them in your search. IHOP (www.ihop-net.org) is an excellent resource for protein names *(22)*. In addition, you will want to perform a BLAST search to make sure you have everything!

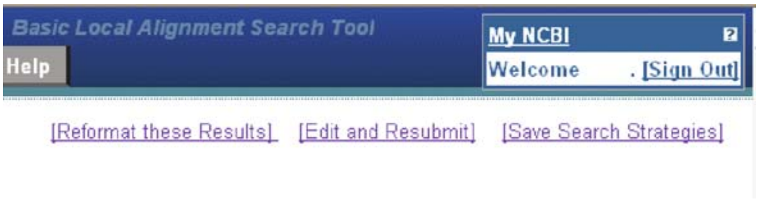


Fig. 1.10. Save search strategies.

2. In addition to the BLAST program provided by NCBI, other BLAST programs exist, which have improved the BLAST algorithm in various ways. Dr. Warren Gish at Washington University in St. Louis has developed WU-BLAST, the first BLAST algorithm that allowed gaped alignments with statistics *(23)*. It boasts speed, accuracy, and flexibility, taking on even the largest jobs. Another program, FSA-BLAST (Faster Search Algorithm), was developed to implement recently published improvements to the original BLAST algorithm *(24)*. It promises to be twice as fast as NCBI's and just as accurate. WU-BLAST is free for academic and non-profit use and FSA-BLAST is an open source under the BSD license agreement.
3. My NCBI is a tool that allows you to customize your preferences, save searches, and set up automatic searches that send results via e-mail. If you find yourself performing the same searches (or even similar searches) repeatedly, you may want to take advantage of this option! To register, go to the NCBI home page and click the "My NCBI" link under "Hot Spots." Once you have registered and signed in, a new option will be available to you on all BLAST and Entrez searches (**Fig. 1.10**).
4. To save a BLAST search strategy, simply click the "Save Search Strategies" link on the results page. This will add the search to your "Saved Strategies" page, which is available through a tab on the top of each page in the BLAST website when you are logged in to My NCBI. Doing so will not save your results, but it will save your query and all parameters you specified for your search so you can run it later to retrieve updated results.

References

1. Dayhoff, M. O., Eck, R. V., Chang, M. A., and Sochard, M. R. (1965) Atlas of Protein Sequence and Structure, National Biomedical Research Foundation, Silver Spring, MD.
2. Hersh, R. T. (1967) Reviews. *Syst Zool* **16,** 262–63.
3. Galperin, M. Y. (2007) The molecular biology database collection: 2007 update. *Nucleic Acids Res* **35,** D3–D4.
4. Batemen, A. (2007) Editorial. *Nucleic Acids Res* **35,** D1–2.
5. Pearson, W. R., and Lipman, D. J. (1988) Improved tools for biological sequence comparison. *Proc Natl Acad Sci USA* **85,** 2444–48.
6. León, D., and Markel, S. (2003) *Sequence Analysis in a Nutshell*, O'Reilly & Associates, Inc., Sebastopol, CA.

7. Sample GenBank Record [Internet]. National Library of Medicine, Bethesda, MD; [modified October 23, 2006; cited November 24, 2007]. Available from: http://www.ncbi.nlm.nih.gov/Sitemap/samplerecord.html
8. EMBL Nuleotide Sequence Database User Manual [Internet]. The European Bioinformatics Institute, Cambridge, United Kingdom; [modified June 7, 2007; cited November 24, 2007]. Available from: http://www.ebi.ac.uk/embl/Documentation/User_manual/usrman.html
9. Explanation of DDBJ flat file Format [Internet]. DNA Data Bank of Japan, Mishima, Shizuoka, Japan; [modified August 7, 2007; cited November 24, 2007]. Available from: http://www.ddbj.nig.ac.jp/sub/ref10-e.html
10. NCBI-GenBank Flat File Release 162.0 [Internet]. National Library of Medicine; [modified October 15, 2007; cited November 20, 2007]. Available from: ftp://ftp.ncbi.nih.gov/genbank/gbrel.txt
11. Needleman, S. B., and Wunsch, C. D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol* **48,** 443–53.
12. Smith, T. F., and Waterman, M. S. (1981) Identification of common molecular subsequences. *J Mol Biol* **147,** 195–97.
13. Dayhoff, M. O., Schwartz, R. M., and Orcutt, B. C. (1978) *Atlas of Protein Sequence and Structure* (Foundation, N. B. R., Ed.), Vol. 5, pp. 345–58, National Biomedical Research Foundation., Silver Spring, MD.
14. Henikoff, S., and Henikoff, J. G. (1992) Amino Acid Substitution Matrices from Protein Blocks. *Proc Natl Acad Sci USA* **89,** 10915–19.
15. Baxevanis, A. D., and Ouellette, B. F. F. (2005) Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins, John Wiley & Sons, Inc., Hoboken, New Jersey.
16. Wheeler, D. G. (2003) Selecting the right protein scoring matrix. *Curr Proto Bioinformat* 3.5.1–6.
17. Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990) Basic local alignment search tool. *J Mol Biol* **215,** 403–10.
18. Roy-Engel, A. M., Carroll, M. L., Vogel, E., Garber, R. K., Nguyen, S. V., Salem, A. H., Batzer, M. A., and Deininger, P. L. (2001) Alu insertion polymorphisms for the study of human genomic diversity. *Genetics* **159,** 279–90.
19. Tompa, M., Li, N., Bailey, T. L., Church, G. M., De Moor, B., Eskin, E., Favorov, A. V., Frith, M. C., Fu, Y. T., Kent, W. J., Makeev, V. J., Mironov, A. A., Noble, W. S., Pavesi, G., Pesole, G., Regnier, M., Simonis, N., Sinha, S., Thijs, G., van Helden, J., Vandenbogaert, M., Weng, Z. P., Workman, C., Ye, C., and Zhu, Z. (2005) Assessing computational tools for the discovery of transcription factor binding sites. *Nat Biotechnol* **23,** 137–44.
20. Ma, B., Tromp, J., and Li, M. (2002) PatternHunter: faster and more sensitive homology search. *Bioinformatics* **18,** 440–5.
21. Tamura, K., Dudley, J., Nei, M., and Kumar, S. (2007) MEGA4: molecular evolutionary genetics analysis (MEGA) software version 4.0. *Mol Biol Evol* **24,** 1596–9.
22. Hoffmann, R., and Valencia, A. (2004) A gene network for navigating the literature. *Nat Genet* **36,** 664–64.
23. Gish, W. (1996–2004) WU BLAST 2.0 [Internet]. Saint Louis, MO; [modified March 22, 2006; cited January 3, 2008]. Available from: http://blast.wustl.edu
24. Cameron, M., Williams, H. E., Bernstein, Y., and Cannane, A. (2004–2006) FSA BLAST [Internet]. [modified March 8, 2006; cited January 3, 2008]. Available from: http://www.fsa-blast.org
25. Madden, T. (2002) The BLAST Sequence Analysis Tool [Internet]. National Library of Medicine, Bethesda, MD; [modified August 13, 2003; cited January 4, 2008]. Available from: http://www.ncbi.nlm.nih.gov/books/bv.fcgi?rid=handbook
26. Web BLAST page options [Internet]. National Library of Medicine, Bethesda, MD; [cited January 4, 2008]. Available from: http://www.ncbi.nlm.nih.gov/BLAST/blastcgihelp.shtml#Reward-penalty