BlastReport: A Perl Script to Facilitate the Use of Sequence Databases for Mapping and Clustering

Jeanette McClintick and Howard J. Edenberg
Indiana University School of Medicine, Indianapolis, IN, USA

There are many projects for which it is necessary to find or construct contigs or localized maps within a given chromosomal region. This can be very time consuming without some form of computer automation. We present here a method and a perl script that can be used to facilitate this using software freely available for most computers. The method uses blastcl3 from the National Center for Biotechnology Information (NCBI) to run Basic Local Alignment Search Tool (BLAST) searches of NCBI databases using text files containing multiple query sequences. The perl script BlastReport, available on our Web site, reads the BLAST output, reorganizes the results for ease of use and eliminates useless information.

There are several kinds of projects for which this is important. In genetic mapping experiments, chromosomal regions are initially defined by markers spaced at 5–10 cM intervals. It is then necessary to create detailed local maps of these regions to allow choice and ordering of additional markers for fine mapping and to select potential candidate genes for analysis. Markers for fine mapping are often chosen from different maps and have been placed with different levels of confidence. Ideally, marker maps and distances would be obtained from the complete genomic sequence, but that is not yet available for large regions of the human genome, and data for many other organisms is even more limited. One can, however, use currently available data to build local maps and identify genes and ESTs within the region. A second type of project involves assembling sequence data for the genomic region around a group of related genes and their associated regulatory elements.

The rapidly increasing availability of DNA sequence is an invaluable resource for these projects, but the very wealth of data makes searching through the databases and analyzing the results a time-consuming endeavor. The continual deposition of new sequences into the databases requires frequent repetition of searches, increasing the time required. Our suggested method and simple program can assist in these tasks.

To construct a map or contig starting from a marker, gene or EST, the sequence(s) can be searched against a database such as GenBank® (3) using the blastn program of the BLAST (1,2) tool available at NCBI (http://www.ncbi.nlm.nih.gov/blast/). Finished sequences are found in the nonredundant (nr) database. Unfinished or draft sequences are found in the unfinished high throughput genomic sequences (htgs) database. The movement of sequences from the htgs database to the nr database upon completion of a BAC sequence requires one to search both databases. When a significant alignment is found, the ends of this new sequence can be used to search for adjoining sequences. Alternatively, if searching reveals that the sequence is on a still unfinished BAC (in unordered pieces), sequences from the BAC can be used as queries for further searching. A map of the region can be built by this iterative process. Other information about a BAC clone may be obtained from the clone registry at NCBI (http://www.ncbi.nlm.nih.gov/genome/clone/).

There are, however, limitations that can be both frustrating and time consuming. One is the very redundant nature of the nr database. Another is the retrieval of false positives generated when there is a repetitive sequence within the query. These two problems can lead to a very large number of irrelevant alignments that must be examined to find the identical matches. An identical match, if it is short, may not be found if there are too many long near-alignments. These problems can be minimized by appropriate choice of search strategies and by the use of the simple program we have developed. Because the searches must be repeated frequently until a region is completely sequenced, the time saved can be substantial.

The Web-based BLAST tool available at NCBI for this process is excellent when searching a single sequence. It can, however, be very time consuming to search a large number of query sequences (e.g., a set of markers in a region). We report an alternative method using software that is available for free. The process uses blastcl3 (4) (available for Mac OS®, Microsoft® Windows™ and several UNIX® environments; ftp://ncbi.nlm.nih.gov/blast/network/netblast/), and a Perl script, BlastReport, available from our Web site (http://bmbmac.biochemistry.iupui.edu/Blastrep). [Perl is a scripting tool or language that is available on most UNIX machines, Mac OS and the Windows environment free of charge (http://www.Perl.com/pub/language/info/software.html). Information about Perl is available at http://www.Perl.com and http://www.Perl.org.] Blastcl3 eliminates much of the manual and repetitive work of the BLAST searches by using a text file containing multiple query sequences in FASTA format as input for BLAST searches. It runs locally and communicates to the BLAST server at NCBI. The results of these searches are written to a text file that has a format very similar to output of the browser-based BLAST search. Alternatively, the re-
Results can be returned in HTML format and viewed using a Web browser. These reports can, however, be very large and often contain alignments that are not useful. Judicious choice of parameters can reduce the clutter and increase the ability to detect meaningful alignments. Limiting the number of descriptions and alignments returned (parameters -v and -b, respectively) to about 10 each reduces the need to scan through many pages of alignments that are not likely to be meaningful. The output files can, nevertheless, be large and time consuming to review. BlastReport sorts through the BLAST output and formats it for rapid evaluation. A great improvement in signal-to-noise ratio for the BLAST searches can be obtained by setting the EXPECTATION parameter -e to limit the alignments returned. The default value of 10 is much too large for this process. Choosing a value for EXPECTATION is a balancing act between eliminating poor alignments or repeat elements and losing significant alignments of small markers or exons. The BLAST programs calculate an EXPECTATION value for each alignment. This calculated EXPECTATION gives an indication of how likely an alignment of this quality will occur by chance for a database of this size. BLAST uses the value of the EXPECTATION parameter to limit the alignments returned. Any alignment whose calculated EXPECTATION is larger than the supplied value is not returned. [See Reference 2 or the BLAST Course available on the NCBI BLAST Web site for more information on EXPECTATION.]

Some microsatellite markers and ESTs contain repetitive elements such as Alu sequences, which have many alignments that are highly similar but do not represent real positives. For example, marker D16S748, which contains an Alu sequence, when used with the EXPECTATION parameter set to 1.0, yielded a text file that was 300 pages long but contained only three useful alignments. BLAST, by default, filters out the low-complexity regions such as the tandem repeats found in a microsatellite marker. This has the effect of producing two local alignments for one marker, one for each end. The EXPECTATION parameter must be set to return these shorter alignments without returning near-alignments to repetitive elements. With the default low-complexity filter on, an EXPECTATION of 1.0-20 (entered as 1e-20) works well. Turning off the low-complexity filter usually gives one alignment instead of two, but a smaller EXPECTATION (e.g., 1e-35) is needed to eliminate alignments to the repeated segment. A difference in length between the tandem repeat in the marker and in the genomic sequence frequently occurs in these highly polymorphic markers, so there are usually gaps in the alignments. If the difference in repeat length is large, two local alignments may be produced, one on each side of the gap. For cDNA sequences, an EXPECTATION parameter

---

**Figure 1. Example of output from BRept_Full.** Query: microsatellite marker D5S461. DATABASE: hgs. EXPECTATION: 1 × 10^{-30}. Low complexity filter turned off. D5S461 had 1 alignment each to AC008391 and AC021572 and has two shorter alignments to AC016618. [Note that AC021572 is described as from chromosome 18; the exact match to this marker suggests that the chromosomal origin of AC021572 is misidentified. We used the BAC ends, AQ31730 and AQ37133, and five 500-700 nucleotide segments from this clone as search queries. Each of these aligned to one or more clone sequences annotated as chromosome 5, including AC008391, AC016618 and AC022431. Some of them also aligned to AC022944, labeled as chromosome 3.] The two short alignments with AC016618 occur because the query and the genomic segment differ in length of the dinucleotide repeat. There is also a repeat element in this marker between nucleotides 67 and 144 that is revealed in the alignment list by the presence of multiple, nonexact alignments to that region.
of $1 \times 10^{-30}$ usually works well, although sequences representing short exons may be missed. If one suspects that an exon has been missed (see below), one can rerun the search using that exon as a query.

As a general guideline for modifying our suggested values, we note that perfect matches of 100 nucleotides give an expectation approximately $10^{-47}$, 75 nucleotides about $10^{-33}$, and 61 nucleotides about $10^{-25}$. It is important to note that the quality of the sequence for most microsatellite markers is not as good as the genomic sequence currently being produced, so some mismatches are to be expected (Figure 1). A match with 98/110 identities gives an expectation of approximately $10^{-25}$. Both the length of the query sequence and the size of the database affect the expectation calculated for a particular local alignment. These values are based on the database size in August 2000; the expectation will increase tenfold as the database grows tenfold. This parameter may need to be adjusted for future growth so that alignments are not lost. The recommended expectation settings have been developed for searches of the hgs and nr databases; both are currently on the order of a billion bases. When using smaller databases, the calculated expectations will be smaller for similar alignments.

A second example of the variability of calculated expectation values involves cDNAs. In searching hgs with the reference sequence NM_000807 (GABRA2; 2189 bases), the alignment of exon 4 to AC015523 had an expectation of $3 \times 10^{-30}$ for an exact 71/71 match. When a 90-base sequence from NM_000807 that included exon 4 was used as the query sequence, the expectation was 9e-32. The decrease in expectation is due to the decrease in query string length. This indicates that the expectation parameter may need to be adjusted to avoid missing small exons when using full cDNA sequences as search sequences (also see below).

BlastReport reformats the initial blastcl3 output to create a report that is much easier to evaluate. BlastReport eliminates many of the alignments to repetitive elements such as Alus by inspection of the alignments to find regions of the query sequence that have many alignments. A number of these regions will be listed, but some or all of the alignments themselves will be eliminated. This can remove between 10 and 100 pages from the report.

BlastReport produces three files. BRept_Full provides a summary for each query sequence, followed by the alignments that are most likely to be biologically significant (Figure 1). BRept_Sum is a list of the regions of alignment only (Figure 2). BRept_Tab is a tab-delimited file containing query name, the name of the aligning sequence, the clone name if given, the first region of alignment and its expectation. BRept_Tab can be opened in a spreadsheet such as Microsoft Excel® to produce a table of alignments that can be sorted. Sorting by the aligning sequence (hit) makes it easy to see if that sequence contains multiple query sequences, such as two markers or a marker and a portion of a cDNA.

BlastReport prompts the user to specify whether markers or cDNAs were used for the search. Microsatellites and cDNA sequences differ in size and in the way that they align to genomic sequences. Microsatellite markers are 100–400 bp long and contain tandem repeats of 1–4 nucleotides that are 40–150 bp long. Since microsatellites are short, the entire sequence is usually found in one clone. On the other hand, cDNA sequences may have as many local alignments as they have exons, and the exons may be separated by large introns and be on different clones. The cDNA setting prints more data to adjust for the increased number of alignments expected for cDNAs. Individual exons or genomic segments should be analyzed using the marker setting. (See below for more on these limits.)

BRept_Full has an “eye-catcher” line for every query that has alignments (Figure 1A). This aids in scanning the report or searching using a text or word processor. The list of one-line descriptions with scores from the original BLAST output is next (Figure 1B), followed by a list of the top alignments (Figure 1C). For each alignment, the region of the alignment in the query sequence, the expectation, the ratio of matching residues to the total length of the local alignment and the alignment sequence identifier are given. This shows where alignments are found, how good they are and if there are repeat sequences in the query string (indicated by multiple matches to the same part of the query sequence). For each sequence producing a significant alignment, the regions of alignment are then shown (Figure 1D). For the cDNA setting, the alignments are presented in order of the query sequence rather than in order of the score (Figure 2). This helps the user to identify exons and to see if any are missing an alignment. If one region of a cDNA is missing, it might represent a small exon; we recommend that the search be repeated using the missing sequence with 5–10 flanking nucleotides. The most significant alignments are shown as they appear in the original BLAST report (Figure 1E). Within the BlastReport program there are easily modified parameters for marker and cDNA settings that limit the number of alignment regions, sequences with alignments and full align-

---

**Querystring for output from BRept_Sum.** NM_001043 is the reference sequence for cDNA SLC6A2, a noradrenaline transporter. Regions that align are sorted in sequence order along the query rather than in order of scores. This aids in the identification of exons in the cDNA. AC007602.3 is a working draft sequence containing six unordered pieces (deposited 6 April 2000). In this case, all 13 exons (spanning 2136 bp in the reference sequence) match sequences spanning 46006 bp within a 71312-bp contig.
ments printed in BRept_Full (Figure 1C, 1D, and 1E). The number of one-line descriptions (Figure 1B) is limited by the value of parameter -v used in the original blastcl3 run. See the program documentation at our Web site for full details on changing these values. Since the original blastcl3 output is not deleted by BlastReport, changes in parameters can be made and BlastReport rerun as needed. At the end of BRept_Full is a list of the query sequences that had no alignment in the original BLAST report, eliminating 40 lines of useless information for each of these.

The strategy presented for use of blastcl3 followed by BlastReport, with appropriate parameters, can greatly improve the efficiency with which multiple markers or other query sequences can be used to find matches in the databases. This strategy decreases the workload and increases the ability to evaluate the matches returned. The software, available free, will run on a small workstation or desktop computer; we use an iMac, but the software is also available for Windows and UNIX. This method does not require the downloading of NCBI data to a local computer along with the problem of retrieving new or updated sequences on a regular basis. These features make it an inexpensive alternative for researchers with limited computer resources and/or budgets to build local maps or find contiguous genomic regions.

ACKNOWLEDGMENTS

This research was supported by U.S. Public Health Service grant nos. AA06460 (to H.J.E.) and U10 AA08403 (The Collaborative Study on the Genetics of Alcoholism; H. Begleiter, PI) from the National Institute of Alcohol Abuse and Alcoholism.

REFERENCES


Received 15 June 2000; accepted 5 October 2000.

Address correspondence to:
Dr. Howard J. Edenberg
Dept. of Biochemistry and Molecular Biology
Indiana University School of Medicine
Indianapolis, IN 46202-5122, USA
e-mail: edenberg@iupui.edu