

Patent searches for genetic sequences: How to retrieve relevant records from patented sequence databases

Results from current homology-based search tools, designed to locate biologically relevant sequences, present a level of uncertainty from an intellectual property standpoint.

Guillaume Dufresne, László Takács*, Hendrik C. Heus, Jean-Jacques Codani, and Manuel Duval

Once a naturally occurring or recombinant DNA or polypeptide sequence has been characterized and assigned a function, how is its uniqueness currently assessed from an intellectual property standpoint? Databases of patented sequences, such as the public database PAT¹ (the patent sequences division of GenBank) and the proprietary database GENESEQ² (provided by Derwent Thomson Scientific), are updated repositories that can be used to run patent searches. Apart from ancillary annotation, patented sequence databases comprise the same type of information—DNA and polypeptide sequences—as primary DNA and protein databases¹ such as EMBL, GenBank, and DDBJ. Being sequentially derived from the latter type of database, their formats are similar. Thus, database search tools such as BLAST³ and FASTA⁴, which are designed to locate biologically relevant sequences, are also usually used to assess the intellectual property content⁵ of genes.

Patents are often granted on both the invention of a specific genetic sequence and a set of sequences sharing a given degree of sequence identity with the invention. Patent searches run with homology-based search tools such as BLAST and FASTA may fail to spot relevant database records, leading to inaccurate evaluation of intellectual property rights. In this article, we discuss an approximate string matching algorithm as a solution for a legally meaningful way of searching for a genetic sequence.

What kind of information do BLAST and FASTA retrieve?

BLAST and FASTA are designed to retrieve putative homologs of a given query

Guillaume Dufresne is in the Patent Department at Pfizer Legal and László Takács and Manuel Duval are in the Genomics and Bioinformatics Group at Pfizer Global Research Development, Fresnes Laboratoires, 3–9 rue de la Loge, 94265 Fresnes, France; Hendrik C. Heus and Jean-Jacques Codani are at Gene-IT SA, 147 Avenue Paul Doumer, 92500 Rueil Malmaison, France. *Corresponding author (laszlo.takacs@pfizer.com).

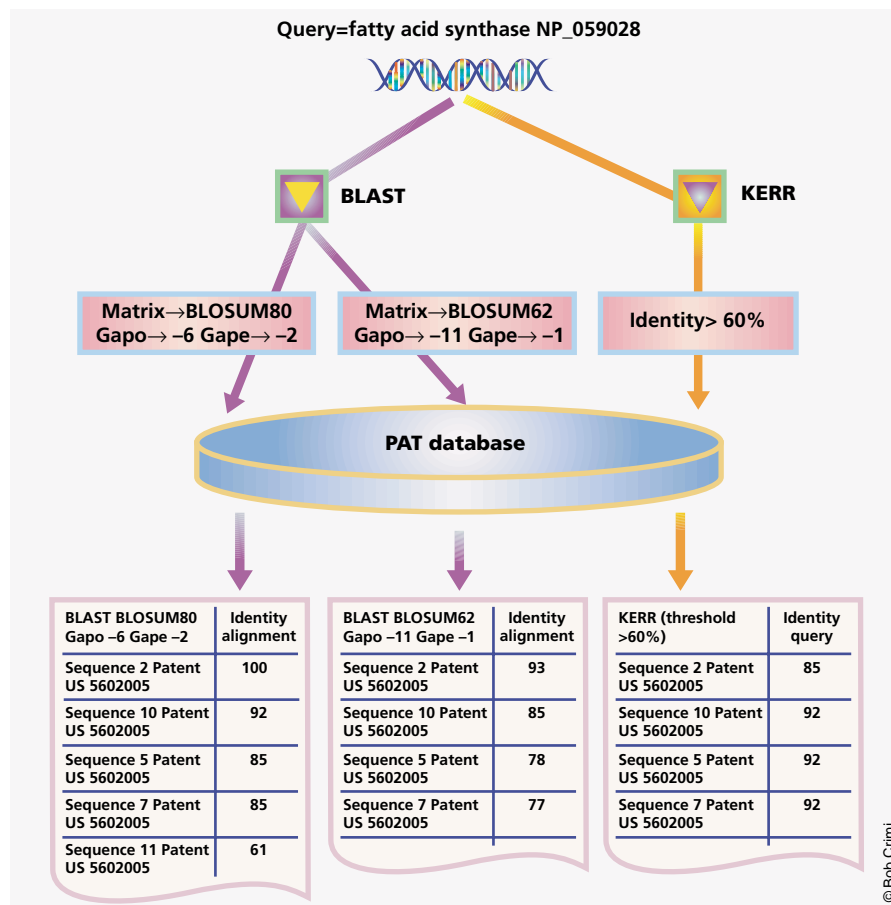


Figure 1. Comparison of KERR and two possible BLAST outcomes. KERR retrieves only database entries meeting the unique criterion of a minimum percent identity (for example, 60%) with the query. In contrast, BLAST requires specifying values for parameters that influence the results. BLAST sorts database entries by scores measuring their degree of homology to the query.

sequence. Accordingly, these tools rely on the concept of homology: two genes are said to be homologous if they have evolved from a common ancestral gene⁶. Both BLAST and FASTA look for evidence of homology between a given pair of sequences using a set of parameters and a scoring system that convey biological information. These configurable parameters, which can be tuned by the investigator, include the gap opening penalty value, substitution matrix, cut-off scores, and gap extension penalty. BLAST search results for

the same query against the same database may differ depending on the parameters chosen (Fig. 1; the same holds true for FASTA). This is because the search algorithms are based on an internally computed score sensitive to parametric changes. Alignments are computed with percent-identity scores only after the best hits are identified by the scoring function.

BLAST performs a local alignment: that is, it optimizes the alignment with any fraction of the query giving the best score. Consequently, in producing alignments with

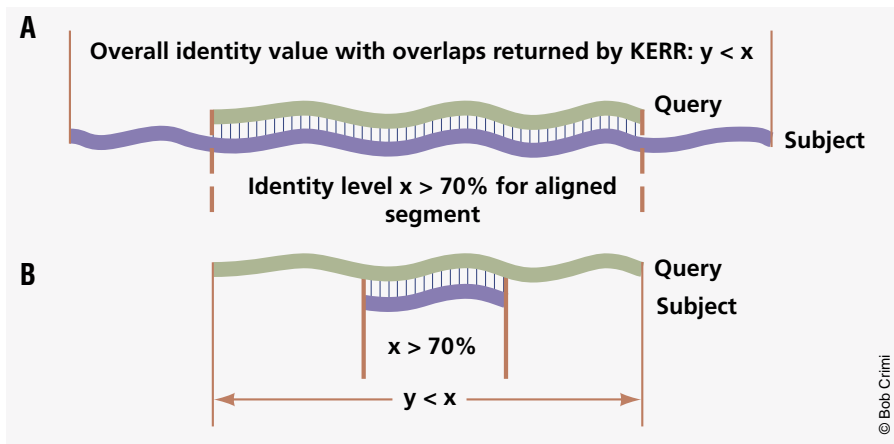


Figure 2. Schematic representation of two possible KERR outcomes. In the first, KERR retrieves all database occurrences that align globally with the query and have a percent identity above the threshold value (for example, $>60\%$). In the second, KERR returns a second identity value, y , taking into account overlaps for either the query or the subject. Gaps in the aligned segment are permitted, as long as the identity value is above the threshold.

percent-identity values, it uses only a portion of the query. The user has no way of specifying the fraction of the query to be aligned. Thus, the alignment results in Figure 1 show, for the same query, 93% identity with one set of parameters and 100% identity with another set. Both BLAST and FASTA rely on the same homology-based search paradigm, and the hits they return depend upon a specific set of parameters.

Scope of gene patents

Taking into account that derivatives of the specific sequence of the invention can confer the same claimed function, patent protection is usually sought for a collection of closely linked sequences that are usually not specifically disclosed in the patent. Typically, the link is based on a given percent-identity threshold⁷. All nearly identical sequences are claimed, in addition to the sequence originally identified. Thus, databases of filed and issued gene patents are populated with sequence records to which an implicit domain of connected sequences is assigned, even though the sequence databases themselves contain only the actual sequences disclosed in the patents. Therefore, searching databases of patented gene sequences means retrieving occurrences in the database that are either identical to the query or related to it by virtue of a minimum identity level, for example, 80%.

An analogous problem exists in information retrieval, where appropriate information in online databases is identified through approximate string matching. String searching consists of finding one or more occurrences of a string of length m in a text⁸. Approximate string matching involves searching a textual database for strings that are similar but not necessarily exactly identical to a given pattern string⁹. The National

Institute of Standards and Technology defines string matching as “searching for approximate (e.g., up to a predefined number of symbol mismatches, insertions, and deletions) occurrences of a pattern string in a text string”. The use of the term “approximate” underlines the fact that a perfect match may not be achievable and that imperfections such as missing and extraneous symbols have to be considered. The primary principle of approximate string matching is the concept of string edit distance, a measure for quantifying the similarity between two strings. Software solutions have been proposed to implement efficient approximate string matching^{8,10}, including KERR¹¹, developed by Gene-IT.

Approximate string matching

As stated previously, gene patents often claim both a specifically disclosed sequence embodiment and a set of related sequences. The elements of this set are defined by their level of identity with the prototypical sequence. In other words, a given sequence is an element of the claimed set if it shares an identity level with the original sequence above the defined threshold. The KERR approximate-string-matching algorithm provides a relevant alternative to BLAST as a sequence-database search method capable of meeting legal requirements.

Given two sequences, the KERR algorithm performs a pairwise alignment to return the best fit between the two sequences. The “best fit” aligns the largest number of residues of the shorter sequence with those of the longer sequence. In contrast to BLAST, the query sequence is indivisible for the outcome of the search—that is, KERR does not perform a local alignment. All database records that align with the query sequence with an iden-

tity greater than the input value threshold are retrieved (Fig. 1). KERR retrieves any database records that (i) contain the whole query or (ii) are contained in the whole query, on condition that percent identity with respect to the shortest sequence is greater than the threshold set (Fig. 2). This means that KERR does not miss database records much longer or shorter than the query, provided that percent identity over the alignment length is above the threshold value. KERR returns percent-identity values with respect to the subject (target) sequence when it is longer than the query sequence (Fig. 2A), as well as with respect to the query sequence (Fig. 2B).

This functionality is intended to assist in decision making for deciphering specific cases. As an example, splice variants of a given gene may be of very different sizes¹², which may confer distinct properties. A KERR query of one splice variant will return other variants with two different forms of percent identity, one on the aligned region and one over the length of the longest variant sequence. Both pieces of data are relevant to the investigator for deciphering the status of the invention in light of patent claims.

KERR is based on a metric (the “edit metric”) that does not reflect the evolutionary process through which two genes can diverge. The edit metric is neutral with regard to any differences or identities assessed between two sequences. KERR finds the minimum number of differences between the query and all database records and returns all matches with respect to a given identity threshold (for example, all subject sequences having more than 80% identity with the whole query). No assumptions are made as to which aligned residues or gaps are more costly in terms of phylogenetic or functional relationships. Though inappropriate for biomedical research purposes, this procedure is appropriate from an intellectual property standpoint, as it mirrors the way in which gene-sequence patents are claimed.

Comparison of KERR and BLAST in the context of patent searches

KERR is invoked with only three input parameters: input sequence data (query), target sequence database(s), and identity threshold. The latter value directs KERR to retrieve only those database subjects matching the query with an identity above the specified threshold. KERR looks for the best fit with the query sequence from beginning to end. In contrast, BLAST seeks the best local alignments to return significant matches in terms of homology. Whereas BLAST finds a subset of sequences in a database that are putative homologs of the query, KERR retrieves the

subset of sequences related to the query in terms of identity only; it does not assign different weights to aligned residues. Any mismatches or gaps are weighted equally as errors and counted as such in finding the optimal alignment. Hence, unlike BLAST, the only search argument used by KERR is the filter threshold targeting the desired record set. Thus, KERR will always return the same results for a given query.

Figure 1 compares two possible BLAST results with the unique data set retrieved by KERR, composed of hits conforming to the identity threshold. In this example, although BLAST retrieves the same hits as KERR, the output data differ significantly between the two search engines. BLAST returns sequence 2 from patent US 5602005 with either 100% or 93% identity, depending on the choice of search parameters. If sequence 5 from patent US 5602005 is claimed with 80% identity, BLAST may not allow one to decipher whether the query falls within the patent, whereas KERR does. Consider a further example, that of someone wishing to find any patent rights potentially covering a short peptide sequence of six amino acids. A BLAST search against GENESSEQ yields no hits when the default parameters are used. Relevant

records are eventually retrieved, but only after various attempts at tuning the BLAST parameters, such as cut-off score. In contrast, KERR produces hits, and delivers the percent identity to the whole subject sequence. KERR does not appear to miss patents and returns values that aid decision making. The problem underlined here is that a very simple question such as “Find any occurrence in the database that is at least 70% identical to my query sequence” cannot be answered by a tool that returns different outcomes depending on how the search is set up.

Conclusions

The advent of genomics has created a logistical challenge with regard to patent searches for patent offices, industry, and academia. To conduct these searches, databases of patented gene sequences have to be interrogated using a search engine capable of retrieving all instances that reach the identity level potentially covered in the patent. To perform this task effectively, such a search engine has to employ an algorithm that compares sequences solely on the basis of identity criteria. Although BLAST and FASTA are powerful tools for answering biologically oriented queries, they present an element of uncer-

tainty when one is looking strictly for sequence identity, as it is often the case in gene patents. We believe it necessary to agree on a more efficient tool for running patent searches, and propose KERR to this end.

1. Ouellette, B.F. & Boguski, M.S. Database divisions and homology search files: a guide for the perplexed. *Genome Res.* **10**, 952–955 (1997).
2. Derwent GENESSEQ (<http://www.derwent.com/genesseq/>).
3. Altschul, S.F. *et al.* Basic local alignment search tool. *J. Mol. Biol.* **215**, 403–410 (1990).
4. Pearson, W. Using the FASTA program to search protein and DNA sequence databases. *Methods Mol. Biol.* **25**, 365–389 (1994).
5. CAS Registry BLAST Similarity Searching via STN Reference Guide (<http://www.cas.org/ONLINE/QRGUIDES/blast.pdf>) (2002).
6. Altschul, S.F. *et al.* Issues in searching molecular sequence databases. *Nat. Genet.* **6**, 119–129 (1994).
7. US Patent and Trademark Office, Manual of Patent Examining Procedure (MPEP) Edn. 8, August, 2001, Chapter 8, Section 803.04, Restriction—Nucleotide (<http://patents.ame.nd.edu/mpep/8/803.04.html>).
8. Baeza-Yates, R. & Navarro, G. Faster approximate string matching. *Algorithmica* **23**, 127–158 (1999).
9. Navarro, G. A guided tour to approximate string matching. *ACM Comput. Surv.*, **33**, 31–88 (2001).
10. Landau, G.M. *et al.* An efficient string matching algorithm with k differences for nucleotide and amino acid sequences. *Nucleic Acids Res.* **14**, 31–46 (1986).
11. Gene-IT. Biofacet User Manual (2002).
12. Neufeld, G. *et al.* Similarities and differences between the vascular endothelial growth factor (VEGF) splice variants. *Cancer Metastasis Rev.* **15**, 153–158 (1996).