

Genomic Information Retrieval

Hugh E. Williams

School of Computer Science and Information Technology,
RMIT University, GPO Box 2476V, Melbourne 3001.

`hugh@cs.rmit.edu.au`

Abstract

The in-silico revolution has changed how biologists characterise DNA and protein sequences. As a first step to exploring the structure and function of an unknown sequence, biologists search large genomic databases for similar sequences. This process of genomic information retrieval has allowed significant advances in biology and led to advancements in critical areas such as cancer research. In this paper, we present a background to genomic information retrieval by describing the problems, collections, and techniques used by biologists for searching large collections. In particular, we identify the problems inherent in the popular search techniques, and discuss how index-based approaches may be applied to solve these problems. We conclude by offering the challenge that information retrieval specialists must continue to make significant contributions to allow further advances in molecular biology research.

1 Introduction

The genomic revolution has changed how molecular biologists discover the structure, function, and role of genes and protein sequences. Understanding the relationship of an unknown gene or protein sequence to known sequences is the key to assigning its function. In this paper, we present an introduction to the genomic information retrieval processes that biologists use, identify the limitations of current approaches, and propose directions for future work.

To characterise an unknown sequence, molecular biologists search genomic databases. To illustrate, consider a simple example of a mutation in the bacteria *E. coli*. This mutation — a change in the DNA sequence — causes an individual to be unable to reproduce. The mutated sequence can be compared to the sequences in a large collection with the aim of finding one or more *homologous* sequences; homology is the sharing of a common evolutionary origin. If homologous sequences are found — and additional work has been performed on a sequence from a different species — the product, for example an enzyme, may be identified. This would allow the research on reproduction in *E. coli* to be directed to that hormone, and may save both time and money in discovering ways to inhibit the growth of the bacteria.

The most popular homology search server processes tens of thousands of queries each day [21]. The BLAST search technique used by this server [2] compares each query exhaustively to every collection sequence, and necessitates that the entire collection being searched is held in main-memory. This has seri-

ous implications for the sustainability of search services: large nucleotide DNA databases — such as GenBank [6] — are roughly doubling in size yearly and have already exceeded 20 billion characters or nucleotide bases in size. Indeed, only high-end hardware can support current searches, and it is unclear whether memory sizes will be sufficient for searching large collections within the next one or two years.

Index-based techniques have been proposed to address the speed and scalability of homology search techniques. These techniques are adaptations of text retrieval techniques that are used in web search engines such as Google¹. Despite its obvious benefits, indexing is not in widespread use because of its limitations in the genomic retrieval domain, including large index sizes, poor accuracy, and the difficulty of supporting long queries. However, it is important that these limitations are addressed: index-based approaches are the only practical method of avoiding exhaustive comparison, and information retrieval practitioners must make new discoveries in this domain.

This paper is structured as follows. We introduce genomics in Section 2, including an introduction to DNA and protein sequences, and nucleotide and amino-acid databases. In Section 3, we present an overview of how biologists use genomic databases for both text- and sequence-based searching. Section 4 discusses search techniques, and includes an introduction to the local alignment recursion used to enumerate the similarity of two genomic sequences, and an overview of the popular heuristic exhaustive search techniques BLAST and FASTA. Section 5 overviews index-based approaches, including our own CAFE technique. We conclude in Section 6 with a discussion of open problems and challenges in genomic information retrieval.

2 Genomics Background

In this section, we overview *genomics*, the study of the complete organisation and structure of the genetic material, the genome. Our introduction is brief and a more detailed background can be found in the texts of Attwood and Parry-Smith [3], Setubal and Meidanis [31], and Lesk [19].

2.1 Nucleotide Sequences

Deoxyribonucleic acid (DNA) contains the complete instructions for the reproduction, development, material synthesis, and differentiation for each cell within an organism.

DNA is present in the chromosomes of nucleated cells in all organisms. It contains genetic information coded using the sequential arrangements of four constituent nucleotide bases, the pyrimidines, *thymine*

Copyright ©2003, Australian Computer Society, Inc. This paper appeared at Fourteenth Australasian Database Conference (ADC2003), Adelaide, Australia. Conferences in Research and Practice in Information Technology, Vol. 17. Klaus-Dieter Schewe and Xiaofang Zhou, Ed. Reproduction for academic, not-for profit purposes permitted provided this text is included.

¹See: <http://www.google.com/>

Amino-Acid	Character	Codon(s)
Alanine (Ala)	A	GCN
Arginine (Arg)	R	CGN & AGR
Asparagine (Asn)	N	AAY
Aspartic Acid (Asp)	D	GAY
Cysteine (Cys)	C	UGY
Glutamine (Gln)	Q	CAR
Glutamic Acid (Glu)	E	GAR
Glycine (Gly)	G	GGN
Histidine (His)	H	CAY
Isoleucine (Ile)	I	AUH
Amino-Acid	Character	Codon(s)
Leucine (Leu)	L	CUN & UUR
Lysine (Lys)	K	AAR
Methionine (Met)	M	AUG
Phenylalanine (Phe)	F	UUY
Proline (Pro)	P	CCN
Serine (Ser)	S	UCN & AGY
Threonine (Thr)	T	ACN
Tryptophan (Trp)	W	UGG
Tyrosine (Tyr)	Y	UAY
Valine (Val)	V	GUN

Table 1: *Amino-acids: the universal code of representation and corresponding codons. Codons are shown with nucleotide wildcards to show codon redundancy. Leucine, for example, is coded by CUN and UUR, that is, the six codons CUA, CUC, CUG, CUU, UUA, and UUG. There are three wildcards, B which represents N or D, Z which represents E or Q, and X which represents any amino-acid.*

T and cytosine C and the purines, *guanine* G and *adenine* A. The bases are arranged into two complementary *polynucleotide chains*; each A in one chain is paired with a T in the other, and each C is paired with a G. A typical short fragment of a DNA strand, that is, a *nucleotide sequence*, is shown in Figure 1; this sequence is described elsewhere [34].

There are around 3×10^9 nucleotide bases in the human genome; each nucleated cell in an organism contains an identical copy of the genome. There are 46 separate polynucleotide chains, that is, 23 pairs of chromosomes, each containing 50 to 250×10^6 base-pairs. However, the genome size and number of chromosomes varies significantly in different species. For example, bacteria such as *E. coli* have genome sizes of around 4×10^6 bases and viruses have genome sizes as small as 170×10^3 base-pairs.

Strings of three nucleotide bases from a DNA molecule, when copied into *messenger RNA* (mRNA), are called a *codon*. Each codon codes for a particular amino-acid that — when combined with other amino-acids through a process involving a third form of genomic nucleotide sequence *transfer RNA* (tRNA) — forms a *polypeptide chain*. There are 64, that is, 4^3 , different possible codons, but only 20 amino-acids. Thus, with this redundancy, there are between 1 and 6 codons that code for each amino-acid. For example the codons GGU, GGC, GGA and GGG all code for Glycine, a single amino-acid. We discuss the amino-acids, and their relationship to codons, later in this section.

Genes are a sequence of codons that yield a product serving a cellular function. If a large amount of a protein is required, a gene may be present several times and the protein synthesised concurrently in several locations throughout the genome. Each human chromosome encodes several thousand genes. Examples of the proteins synthesised include enzymes, the catalysts to virtually all biochemical processes.

2.2 Amino-acid Sequences

Table 1 shows the twenty amino-acids. We show in the table the standard three-letter and single-

character representations, and the corresponding codons. Nucleotide wildcards are shown in the codons to show the redundancy in coding of amino-acids by multiple codons; each amino-acid has between one and six corresponding codons. The wildcard R represents that either an A or G can be substituted at that position. The wildcard N represents any of the four nucleotide bases, and the wildcard Y represents C or T. Note that for our purposes the nucleotide bases T and U are interchangeable.

Methionine is known as the *start codon*, as it signals the beginning of a coding region. The combinations UAA, UAG, and UGA are called *stop codons*, as they do not code amino-acids, but rather signal the end of a coding sequence for a polypeptide protein chain.

Proteins are long polypeptide chains that typically contain hundreds of amino-acids, while some consist of more than a thousand. A typical protein sequence, in this case a gene product from part of the *Haemophilus influenzae* bacteria, is shown in Figure 2; this sequence is described in detail elsewhere [18].

Proteins have a complex structure dictated by the characteristics displayed by individual amino-acids. Amino-acids can be grouped according to characteristics including charge, hydrophobicity, and acidity. The classification of amino-acids allows prediction of relationships between sequences that are not easily seen by comparing the *primary structure* of amino-acids. For example, the primary structure — the linear character sequence — does not represent three-dimensional folds, helices, and sheets.

The properties of amino-acids give rise to *secondary structure*. Secondary structure results from relationships between amino-acids close in the linear sequence and results in three-dimensional helices or sheets. Further, bonds between amino-acids near and distant in the linear sequence gives rise to *tertiary structure*, a further three-dimensional structure that results from a protein folding back on itself. Finally, two or more fold structures brought together may give rise to *quaternary structure* [15].

2.3 Nucleotide Databases

Genomic databases originally consisted of only nucleotide sequences. Indeed, the largest genomic databases are the several publicly available nucleotide sequence databases. The three primary repositories are *GenBank* [6], which is the product of the US human genome initiative, the *European Molecular Biology Laboratory* (EMBL) database [28], which is favoured within Europe, and the DNA Databank of Japan (DDBJ). Typically, organisations such as the European Bioinformatics Institute, which is responsible for EMBL, also maintain many other smaller special-purpose nucleotide databases that contain nucleotide sequences from individual species, or nucleotide sequences of specific function [32].

GenBank is the most popular combined nucleotide and amino-acid sequence database and is cross-updated daily with data from EMBL and DDBJ. We focus on GenBank throughout, but the techniques and principles apply equally to other genomic nucleotide databases.

GenBank stores sequence data generated through the human genome initiative, which not only focuses on the human genome, but also on model organisms such as the bacteria *E. coli*, the fruit-fly *D. melanogaster*, the nematode *C. elegans*, and yeast *S. cerevisiae* [6, 8, 9]. In July 2002, GenBank contained around $20,000 \times 10^6$ nucleotide bases and it continues to exhibit exponential growth. Figure 3

```

ACTCCCCCTCCCGCCCCACTGAACCCCTTGACCCCTGCCCTGCGGCCCCCGC
AGCTTGCTGTTTGGCCGCTCTATTTGCCAGCCCCAAGGACAGAGCTGATC
CTTGAACCTCTTAAGTTCACATTGCCAGGACCAGTGAGCAGCAACAGGGCT
GGGGCTGGGCTTATCTGCCTCCCAGCCAGCCCTGGCTGGAGACATAAAT
AGGCCCTGCAAGAGCTGGCTGCTTAGAGACTGCGAGAAGGAGGTAAGTCCT
GCTCCCTGCTCCGGGTGACTCTGGCTCCCGAGCTCGAGGTTTCAGGCCCTGC
TCCAGGCCGGGCTCTGGGTACCTGAGGTCTTCTCCCACTCTATGCCCTT
CTCCTCACCTTGCTGCAATGAGTGGGGGAGCACGGGCCCTTCTGCATGCTAG
AGGCCCCCACTCAACCAGGCCCTTCTTCTCCTCCAGGTCCCCCAGGGCC
TTCAGGATGAAAGCTACGGTGCTGACCTTGCCCGTGC

```

Figure 1: A nucleotide sequence from *C. aethiops*, the African Green Monkey.

```

MMNFFNFRCIHCRGNLHIAKNGLCSGCQKQIKSFPYCGHCGSELQYYAQHCGNC
LKQEPSWDK MVIIGHYIEPLSILIQRFKFNQFWIDRTLARLLYLAVRDAKRTHQL
KLPEAIPVPLYHFRQWRRGYNQADLLSQQLSRWLDIPNLNNIVKRVKHTYTQRG
LSAKDRRQNLKNAFSLAVSKNEFPYRRVALVFFVITTGSTLNEISKLLRKLGVVEIQ
VWGLARA

```

Figure 2: Gene product from *H. influenzae*, a completely sequenced bacterial genome.

shows the increase in nucleotide bases stored in GenBank since 1983.

Data is submitted to GenBank most often by the users who sequenced the genomic data. However, a large amount of data is gathered by searching over 325,000 journal articles in 3,400 journals each year for sequence data. Additional patented sequence data is gathered by the US Patent and Trademark Office and submitted on approval of each patent. In all cases, prior to release, the data is audited manually for errors.

2.3.1 Textual Components of GenBank

The GenBank distribution is manually indexed, with sequence submissions labelled with natural language descriptors provided by the author. As an example, an author may classify a sequence according to the presence and offsets in the sequence of a coding region, but this may not encompass other components, such as the presence of a repeating sequence. Despite the obvious limitations of manual indexing techniques, the natural language facilitates queries on aspects of the nucleotide data that cannot at present be directly interpreted from the sequence itself.

Figure 4 illustrates a typical, but reasonably short, GenBank flat-file entry. GenBank flat-file records include keyword phrases, the author, journal citation, and a gene name if the sequence is part of a gene. The entry shown is a coding region and the amino-acid translation of the nucleotide sequence is transcribed; the offset of the first codon in the nucleotide sequence is also noted, in this case at offset 3. Often, particularly in longer sequences, a detailed “FEATURES” section is included to annotate sequence regions with details, such as the start and end of each coding region, and the location of introns.

2.4 Protein Databases

GenBank contains amino-acid translations of nucleotide sequences, however several protein databases also exist, such as the Protein Identification Resource (PIR) [14] and Swiss-PROT [4]. Protein databases are typically well-managed and less redundant than nucleotide databases, commonly including classification of sequences into related families and, in some cases, families of families. However, most protein databases are small in comparison to the nucleotide collections: for example, GenBank is over 200 times larger than PIR.

3 An Introduction to Database Searching

Genomic databases, such as GenBank or PIR, are typically searched in two ways: using genomic sequence data or by English textual annotation. Locating *homologous* sequences — by either method — is the most informative result of a database search. Homology means “possessing a common evolutionary origin” [27]. Finding homology may allow inference as to function and role of a sequence, and it always infers structure and classification. We briefly discuss sequence and text-based homology searching in this section.

A query for homology searching of a genomic database is a sequence. The query sequence can be nucleotide or amino-acid and can be compared with the nucleotide or amino-acid components of the database. This leads to four possible searching tasks: nucleotide-to-nucleotide comparison, amino-acid-to-amino-acid comparison, amino-acid-to-nucleotide comparison (in which an amino-acid is first translated into all possible codons), and nucleotide-to-amino-acid. In nucleotide-to-amino-acid searching the nucleotide sequence may be translated into up to six amino-acid sequences, depending on whether the offset of the first codon, or *reading frame*, is known.

Querying returns a ranked list of related sequences, usually with pertinent data including the species, record accession number, and a brief natural language descriptor. Accession numbers are typically used to extract the original flat-file database entries for any results of interest.

Determining rankings by likely homology requires an estimation of the similarity of each sequence in the database to the query sequence. This estimation is a usually either a score or a probability of the similarity having occurred by chance. Intuitively, the ranking should preferentially select sequences more closely related to a query sequence, that is, those that undergo the least number of evolutionary changes should be ranked higher.

A common benchmark is that if more than 30% of two amino-acid sequences are identical, then the sequences are most likely homologous [11]. However, lack of statistical similarity does not infer non-homology; for example, two sequences that do not share significant statistical similarity are homologous if they are both related to a third sequence. We discuss the process of computing the similarity of sequences in more detail in Section 4.

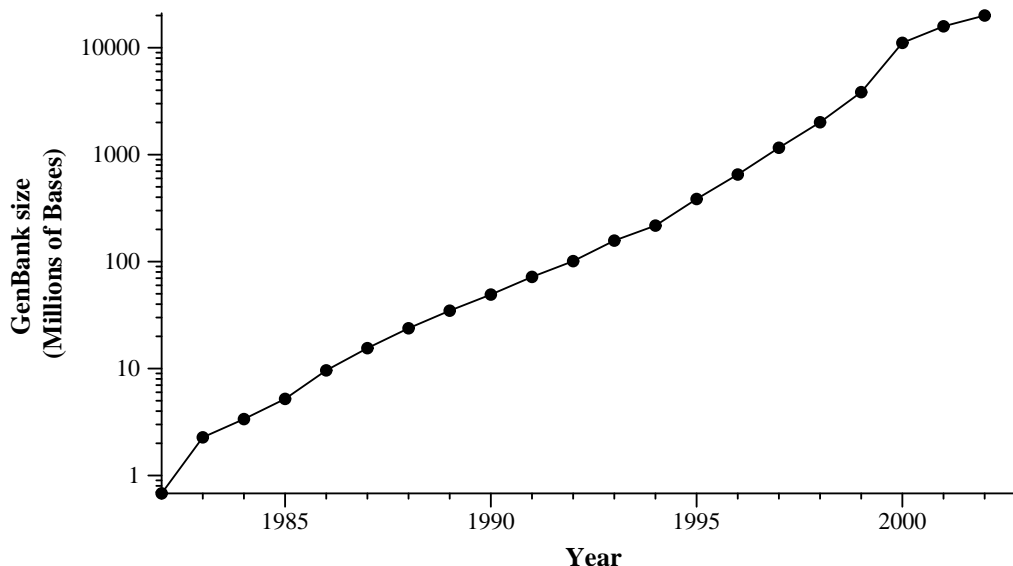


Figure 3: *GenBank growth during the period 1983 to 2002.*

For text-based searching, there are several word-based retrieval techniques for fast query evaluation on genomic databases. Entrez² is an excellent example of using an indexed approach for word-based querying of genomic sequence and literature databases. Using such tools it is possible to formulate Boolean queries that allow searching based on the textual components associated with genomic sequence entries, bibliographical databases, and other related text annotations. Text information retrieval techniques find immediate application in this domain.

Genomic databases are used primarily for either genomic sequence- or word-based searches, as typically there is insufficient genome coverage to enable any other meaningful information to be extracted. However, with the recent complete sequencing of several genomes, genome databases are available that can provide data on mapping, that is, the location of genes within the chromosomes. For most sequences, it is difficult to extract information about gene positioning on a chromosome — or other global statistics — from larger repositories such as GenBank.

3.1 Query Types

If an amino-acid sequence is available, a biologist will almost always use it as a query in preference to a nucleotide sequence. There are several reasons for this. First, amino-acid sequences do not have the codon redundancy of nucleotide sequences and, as such, mutations in the nucleotide sequence will often not affect the amino-acid sequence; second, the ambiguity of nucleotide sequences in terms of codon start position and gene location is removed [12]; third, several non-redundant, well-characterised amino-acid databases exist, in contrast to the vast number of nucleotide sequences in databases that are partially redundant, and poorly or completely unannotated; fourth, nucleotide sequences may come from non-coding regions but may be similar to coding sequences; last, amino-acids have richer properties and scaled similarity scoring schemes can be employed — we discuss this further in Section 4.

Nucleotide sequence searching is the only searching alternative in many circumstances. For example, large libraries of nucleotide *Expressed Sequence Tag* sequences (ESTs) are being created: such sequences

are the result of large-scale sequencing efforts aimed at capturing the sequence of significant portions of the genome. The comparison of these nucleotide sequences to other known nucleotide sequences may determine characteristics where the amino-acid sequence is not yet known. In addition, since the vast majority of the human genome is non-coding, nucleotide comparison is often the only alternative. Indeed, around 70% of the queries on databases at the US National Centre for Biotechnology Information are nucleotide queries [21].

4 Measuring evolutionary distance

Generally, homology between sequences is measured using *local alignment* to determine the similarity of regions, rather than measuring the overall similarity of complete sequences. Local alignment proceeds by character-wise comparison of sequences, and the result is a similarity score and a reconstruction of the evolutionary process that shows how one sequence may have mutated into the other.

Estimation of evolutionary distance requires a measurement of the number of point mutations, or elementary changes, to transform one sequence into another. Generally, evolutionary distance is estimated using the local alignment [33] recursion to find the least number of mutations, that is, the one or more *optimal alignments* between two sequences. Local alignment is exhaustive: all possible evolutionary paths are computed between two sequences, and the basic algorithm is $O(n^2)$ in both time and space.

A local alignment of two homologous globin sequences — human β -chain and α -chain hemoglobin — is shown in Figure 5. The only optimal local alignment extending over 145 amino-acids is shown in the format typically returned to the user: identical amino-acid residues are shown aligned with a “.”, amino-acids with similar properties are aligned with a “:”, and dissimilar alignments or *substitutions* are shown with a space character. Insertions — or deletions from the perspective of the other sequence — are shown with a “-” character; insertions and deletions are collectively known as *indels*, and more than one consecutive indel is a *gap*.

More detail of string matching techniques and their application to biological problems can be found elsewhere [16, 30].

²See: <http://www.ncbi.nlm.nih.gov/entrez/>

LOCUS	AALMTCYTOB 307 bp ds-DNA MAM 14-JAN-1993						
DEFINITION	Alces alces americana cytochrome b gene, 3' end.						
ACCESSION	M98484						
KEYWORDS	cytochrome b.						
SOURCE	Mitochondrion Alces alces americana (organelle Mitochondrion Alces alces americana, subspecies americana) adult liver DNA.						
ORGANISM	Mitochondrion Alces alces americana Eukaryota; Animalia; Chordata; Vertebrata; Mammalia; Theria; Eutheria; Artiodactyla; Ruminantia; Pecora; Cervidae; Odocoileinae.						
REFERENCE	1 (bases 1 to 307)						
AUTHORS	Carr,S.M. and Hughes,G.A.						
TITLE	The direction of hybridisation between species of North American deer (Odocoileus) as inferred from mitochondrial cytochrome b sequences						
JOURNAL	J. Mammal. (1992) In press						
STANDARD	full automatic						
COMMENT	NCBI gi: 336199						
FEATURES	Location/Qualifiers						
CDS	1..298 /note="NCBI gi: 336200" /product="cytochrome b" /codonstart=3 /translation="FGSLLGVCLILQILTGLFLAMHYTPDTMTAFSSVTHICRDVNYGWIIRYMHANGASMFFICLFMHVGRGLYYGSYTFLETWNIGVILLFTVMATAFAG"						
source	1..307 /organism="Alces alces americana" /mitochondrion						
BASE COUNT	90 a 75 c 48 g 94 t						
ORIGIN	001	acttcggttc	tctattagga	gtttgcttaa	tcttacaat	ccttacagga	ctattcctag
	061	caatacatta	tacaccgac	acaataacag	cattctctc	tgtcaccac	atcgccgag
	121	atgtaaatta	cggetgaatc	attcgatata	tgcatgcaa	cggagcctca	atattctca
	181	tctgcttatt	tatacatgta	ggacgaggac	tatactacgg	atcctatact	ttctagaaa
	241	catgaaacat	cggagtgatc	cttctatatta	cagtaatagc	cacagcattc	gctgggatg
	301	tcctacc					

Figure 4: A nucleotide sequence flat-file entry for the “Alces alces americana cytochrome b gene” in GenBank.

4.1 Scoring Sequence Alignments

Amino-acid sequence comparison requires scoring models that incorporate different scores for a match, substitution, and an indel. To compare individual amino-acid residues, scores are often tabulated for each of the 210 possible pairings of the 20 amino-acids. Such tabulations form a substitution matrix of scores on a sliding scale, where amino-acid residues of similar character are given positive substitution scores and those of differing character negative scores. An example scoring matrix is shown in Figure 6. Substitution matrices are generally derived from the observation of homologous families of proteins and the derivation process is discussed in detail elsewhere [10, 17].

Scoring matrices are rarely used in nucleotide comparison. Altschul et al. found that using fixed scores of +5 for any identity and -4 for any substitution worked well in practice [1], and these scores are used almost universally. These fixed scores build on long-term intuition and experience, where several other simple scoring schemes have been applied to nucleotide identities, substitutions, and indels [30].

For both amino-acid and nucleotide comparison, the evolutionary model is unsuited to a single weight for an indel. Instead, a model incorporating different weights for opening and extending gaps is more appropriate [30, 35]. The most often used approach is the application of a single gap-opening weight followed by a linear gap-extension cost. This general form of *affine gap cost* is $c + dn$, where n is the gap length and c and d are positive integers. Affine gap costs are implemented as the gap model in most sequence search tools, including tools such as BLAST [1, 2]

and FASTA [20, 25, 26] that we discuss next in Section 4.2. In general, affine gap costs can be applied in local alignment schemes with no significant affect on the $O(n^2)$ time cost for aligning two sequences of length n [5, 36].

4.2 Fast Exhaustive Searching

Smith-Waterman local alignment identifies the optimal local alignments between two sequences for a given scoring scheme. Unfortunately, local alignment is impractical for database searches on general-purpose hardware: the comparison of a typical query to the sequences in a large collection requires up to a day of processing time using an efficiently-implemented scheme [23]. Therefore, in practice, heuristic variations of local alignment are used for searching large collections. We discuss these in brief in this section.

Two popular heuristic homology search tools are in widespread use: FASTA [20, 25, 26] and BLAST [1, 2]. Both techniques exhaustively compare a query sequence to each sequence in a genomic database, but only perform local alignment when the *coarse similarity* of the query to a database sequence exceeds a threshold. In both approaches, the query sequence is first pre-processed by extracting intervals [37], and the offset or offsets of each interval are recorded. An interval in this context is a fixed-length overlapping subsequence from a sequence, such that there are $l - n + 1$ intervals, for a sequence of length l and interval of length n .

By first preprocessing the query, each database sequence can be processed by extracting intervals, and a single-step lookup used to locate each inter-

approaches, including BLAST and FASTA, require overlapping intervals to achieve acceptable retrieval effectiveness. Third, the algorithms of Orcutt and Barker do not use current approaches to managing large document collections that make indexing practical.

5.2 RAMdb

The Rapid Access Motif database (RAMDB) [13] system was proposed for finding short patterns in genomic databases. In the approach of Fondrat and Dessen, each genomic sequence is indexed by its constituent overlapping intervals in a hash table structure. For each interval in the collection, an associated list of sequence numbers and offsets is stored, allowing rapid location of any motif matching a query motif.

The primary application of RAMDB is the location of motifs either equal or slightly longer in length than the indexed interval length. A user can query with a motif, which may be approximate, and have the sequence and offset of matching motifs in the collection returned without need to exhaustively scan the database. RAMDB is illustrated with examples of locating motif patterns, for example finding the pattern GC(FY)(GA)GGTXXR in the PROSITE database; the use of brackets indicates an optional component, while the character X matches any amino-acid. The indexed approach of RAMDB is shown to result in a 0 to 800-fold speedup in search times over comparable exhaustive approximate pattern matching approaches.

5.3 FLASH

The FLASH search tool redundantly indexes genomic data based on a probabilistic scheme [7]. For each interval of length n , the FLASH search structure stores, in a hash-table, all possible similarly-ordered contiguous and non-contiguous subsequences of length m that begin with the first base in the interval, where $m < n$. Califano and Rigoutsos found that FLASH was of the order of ten times faster for a small test collection than BLAST and was clearly superior in accurately and sensitively determining homologies in database searching. However, the redundant index is impractically large: Barton [5] reports that the index for SWISS-PROT Release 25 — a collection of only 10 Mb — requires almost 2.8 Gb of disk space, around 280 times the size of the SWISS-PROT collection.

5.4 Cafe

Inverted files have been shown to be a successful tool for large text database retrieval [22, 29, 39]. To address the problems with indexing encountered in other attempts, Williams and Zobel [38] propose a two-component partitioned search process embodied in a research prototype system, CAFE. The first component of their approach — a *coarse search* — uses an inverted index to select a subset of sequences that display broad similarity to the query sequence. The second component, a computationally more expensive *fine search* mechanism, ranks the resultant sequences from the coarse search in order of relevance to the query, presenting the ranked results to the user. The partitioning of searching into coarse and fine mechanisms has, for example, been successfully used for pattern matching in databases of names [40, 41]. To ensure efficient query evaluation, they use a query evaluation technique adapted from such methods.

To achieve efficient and effective retrieval from genomic databases, CAFE uses several ranking techniques based on the index structure in the coarse

search phase. These ranking techniques — which the authors refer to as FRAMES — incorporate the relative positioning of matching intervals, as well as other calculated metrics, to give a model of likely homologous alignments. In their fine search scheme, the FRAMES structure is used as the basis of an optimised gapped local alignment.

Results show that CAFE searching is only marginally less accurate than the original 1990 version of BLAST and FASTA, but that the 1997 version of BLAST is less accurate than the other approaches. Most importantly, CAFE is significantly faster than exhaustive approaches: in 1997, Williams and Zobel showed that CAFE was around eight times faster than BLAST and more scalable; in unreported work, we have found that CAFE is now over fifty times faster for short queries.

CAFE has several drawbacks. First, the CAFE index is around 2 to 2.5 times the size of the compressed collection; while this is practical on disk, only a small part of the index can be cached in main-memory at any time, leading to frequent disk accesses for query evaluation. Second, the FRAMES ranking structure is large: Williams and Zobel do not report results for query sequences of greater than 500 characters in length, and these are now common in practice. Third, CAFE is a research prototype and does not support index updates; genomic collections are updated daily. Last, the local alignment process used in the fine searching step does not implement the state-of-the-art. CAFE is, however, a promising step towards indexing for fast homology searching.

6 Conclusion

Large-scale homology searching is key to the in-silico biological revolution. With large genomic collections that double in size almost yearly, increasing user numbers, and growing query lengths, search efficiency is essential. In this paper, we have presented an overview of why homology search is crucial to molecular biologists, the collections used, and the exhaustive techniques that are in popular use.

We have argued that exhaustive searching is unsustainable because each sequence in a collection must be compared to a query. Collections — even when compressed to approximately one quarter of their original size — no longer fit within the main-memory of standard desktop workstations, and this is a known requirement for efficient processing. Specialised hardware-based solutions mitigate but do not solve these problems. Index-based, software solutions that adapt text retrieval techniques are therefore essential to reduce the fundamental cost of homology search.

Index-based solutions — including our own CAFE technique — are not in widespread use. There are several reasons for this and each is a product of working in a difficult domain: index sizes are large, maintaining accuracy comparable to exhaustive approaches is difficult, and processing long queries requires novel techniques.

Index-based solutions are the only sustainable software solution to homology search and problems inherent in these approaches must be addressed. Information retrieval practitioners must therefore be engaged to make fundamental discoveries in this difficult domain. The intersection between Information Retrieval and Genomics is an exciting and important emerging area.

References

- [1] S. Altschul, W. Gish, W. Miller, E. Myers, and D. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215:403–410, 1990.
- [2] S. Altschul, T. Madden, A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D. Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–3402, 1997.
- [3] T. Attwood and D. Parry-Smith. *Introduction to Bioinformatics*. Prentice-Hall, 1999.
- [4] A. Bairoch and R. Apweiler. The SWISS-PROT protein sequence data bank and its new supplement TrEMBL. *Nucleic Acids Research*, 24:21–25, 1996.
- [5] G. Barton. Protein sequence alignment and database scanning. In M. J. E. Sternberg, editor, *Protein Structure Prediction: A Practical Approach*. IRL Press at Oxford University Press, 1996.
- [6] D. A. Benson, I. Karsh-Mizrachi, D. J. Lipman, J. Ostell, B. A. Rapp, and D. L. Wheeler. Genbank. *Nucleic Acids Research*, 30:17–20, 2002.
- [7] A. Califano and I. Rigoutsos. FLASH: A fast look-up algorithm for string homology. In *International Conference on Intelligent Systems for Molecular Biology*, pages 56–64, Bethesda, MD, 1993.
- [8] M. Cinkosky, J. Fickett, P. Gilna, and C. Burks. Electronic data publishing in Genbank. *Science*, 252:1273–1277, 1991.
- [9] F. Collins, A. Patrinos, E. Jordan, A. Chakravarti, R. Gesteland, and L. Walters. New goals for the U.S. human genome project: 1998–2003. *Science*, 282(5389):682–689, 1998.
- [10] M. Dayhoff. *Atlas of Protein Sequence and Structure*. National Biomedical Research Foundation, Washington, D.C., 1978.
- [11] R. Doolittle. *Of URFs and ORFs*. University Science Books, Mill Valley, CA, 1986.
- [12] R. Doolittle. Searching through sequence databases. *Methods in Enzymology*, 183:99–110, 1990.
- [13] C. Fondrat and P. Dessen. A rapid access motif database (RAMdb) with a search algorithm for the retrieval patterns in nucleic acids or protein databanks. *Computer Applications in the Biosciences*, 11(3):273–279, 1995.
- [14] D. George, W. Barker, H. Mewes, F. Pfeiffer, and A. Tsugita. The PIR-international protein sequence database. *Nucleic Acids Research*, 24:17–20, 1996.
- [15] A. Griffiths, J. Miller, D. Suzuki, R. Lewontin, and W. Gelbart. *An Introduction to Genetic Analysis*. Freeman, New York, fifth edition, 1993.
- [16] D. Gusfield. *Algorithms on Strings, Trees, and Sequences*. Cambridge University Press, 1997.
- [17] S. Henikoff. Performance evaluation of amino acid substitution matrices. *Proteins*, 17(1):49–61, 1993.
- [18] T. Larson and S. Goodgal. Sequence and transcriptional regulation of com101a, a locus required for genetic transformation in haemophilus influenzae. *Journal of Bacteriology*, 173:4683–4691, 1991.
- [19] A. Lesk. *Introduction to Bioinformatics*. Oxford University Press, 2002.
- [20] D. Lipman and W. Pearson. Rapid and sensitive protein similarity searches. *Science*, 227:1435–1441, 1985.
- [21] S. McGinnis. Personal communication. (GenBank user services, National Centre for Biotechnology Information (NCBI), National Library of Medicine, US National Institute of Health), Jan. 1998.
- [22] A. Moffat and J. Zobel. Self-indexing inverted files for fast text retrieval. *ACM Transactions on Information Systems*, 14(4):349–379, Oct. 1996.
- [23] E. Myers and W. Miller. Optimal alignments in linear space. *Computer Applications in the Biosciences*, 4:11–17, 1988.
- [24] B. Orcutt and W. Barker. Searching the protein database. *Bulletin of Mathematical Biology*, 46:545–552, 1984.
- [25] W. Pearson. Rapid and sensitive sequence comparison with FASTP and FASTA. *Methods in Enzymology*, 183:63–98, 1990.
- [26] W. Pearson and D. Lipman. Improved tools for biological sequence comparison. *Proc. National Academy of Sciences USA*, 85:2444–2448, 1988.
- [27] G. Reeck, C. de Haen, D. Teller, R. Doolittle, W. Fitch, R. Dickerson, P. Chambon, A. McLachlan, E. Margoliash, T. Jukes, and E. Zuckerkandl. Homology in proteins and nucleic acids: A terminology muddle and a way out of it. *Cell*, 500:667, 1987.
- [28] C. Rice, R. Fuchs, D. Higgins, P. Stoehr, and G. Cameron. The EMBL data library. *Nucleic Acids Research*, 21:2967–2971, 1993.
- [29] G. Salton and M. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, New York, 1983.
- [30] D. Sankoff and J. Kruskal, editors. *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*. Cambridge University Press, 1999.
- [31] J. C. Setubal and J. Meidanis. *Introduction to Computational Molecular Biology*. PWS Publishing Company, 1997.
- [32] B. Shomer, R. Harper, and G. Cameron. Information services of the European bioinformatics institute. *Methods in Enzymology*, 266:3–27, 1996.
- [33] T. Smith and M. Waterman. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147:195–197, 1981.
- [34] M. Sorci-Thomas and M. Kearns. Transcriptional regulation of the apolipoprotein A-I gene. *Journal of Biological Chemistry*, 266:18,045–18,050, 1991.
- [35] M. Waterman. *Introduction to Computational Biology: Maps, Sequences and Genomes*. Chapman and Hall, London, 1995.

- [36] M. Waterman, T. Smith, and W. Beyer. Some biological sequence metrics. *Advances in Mathematics*, 20:367–387, 1976.
- [37] W. Wilbur and D. Lipman. Rapid similarity searches of nucleic acid and protein data banks. *Proceedings of the National Academy of Science*, 80:726–730, 1983.
- [38] H. Williams and J. Zobel. Indexing and retrieval for genomic databases. *IEEE Transactions on Knowledge and Data Engineering*, 14(1):63–78, 2002.
- [39] I. Witten, A. Moffat, and T. Bell. *Managing Gigabytes: Compressing and Indexing Documents and Images*. Morgan Kaufmann Publishers, Los Altos, CA 94022, USA, second edition, 1999.
- [40] J. Zobel and P. Dart. Finding approximate matches in large lexicons. *Software Practice and Experience*, 25(3):331–345, Mar. 1995.
- [41] J. Zobel, A. Moffat, and R. Sacks-Davis. Searching large lexicons for partially specified terms using compressed inverted files. In R. Agrawal, S. Baker, and D. Bell, editors, *Proc. International Conference on Very Large Databases*, pages 290–301, Dublin, Aug. 1993.