5 Database search

5.1 Biological databases

Biological databases contain biological information, mainly collected from molecular biology experiments, life science literature, and bioinformatics analyses.

Categories of databases

Annual Nucleic Acids Research database issue includes the following database categories.

- Nucleotide Sequence Databases
- RNA sequence databases
- Protein sequence databases
- Structure Databases
- Proteomics Resources
- Human and other Vertebrate Genomes
- Genomics Databases (non-vertebrate)
- Plant databases
- Human Genes and Diseases
- Metabolic and Signaling Pathways
- Immunological databases
- Microarray Data and other Gene Expression Databases
- Cell biology
- Organelle databases
- Other Molecular Biology Databases

GenBank

- A comprehensive database of publicly available nucleotide sequences
- Produced and maintained by NCBI (National Center for Biotechnology Information, URL: http://www.ncbi.nlm.nih.gov)

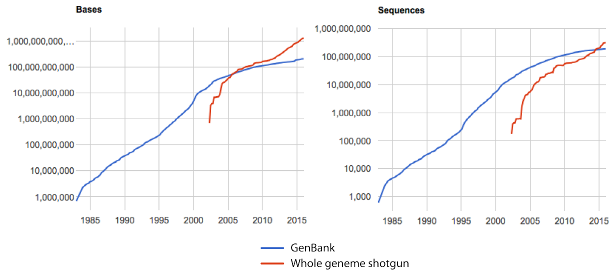


Figure 5.1: Growth of GenBank and WGS (source: NCBI)

UniProt

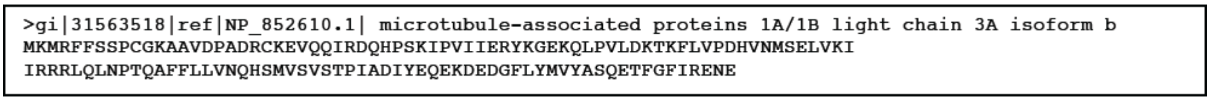
- A central repository of protein data from Swiss-Prot, TrEMBL, and PIR-PSD databases
- Maintained by the UniProt consortium

Sequence data

- Identifier
- Sequence

Data format of sequence data

FASTA is the most popular format for sequence data.



```
>gi|31563518|ref|NP_852610.1| microtubule-associated proteins 1A/1B light chain 3A isoform b
MKMRFFSSPCGKAAVDPADRCKEVQQIRDQHPSKIPVIIERYKGEKQLPVLDKTKFLVPDHVNMSELVKI
IRRRLQLNPTQAFFLLVNQHSMVSVSTPIADIYEQEKDEDGFLYMVYASQETFGFIRENE
```

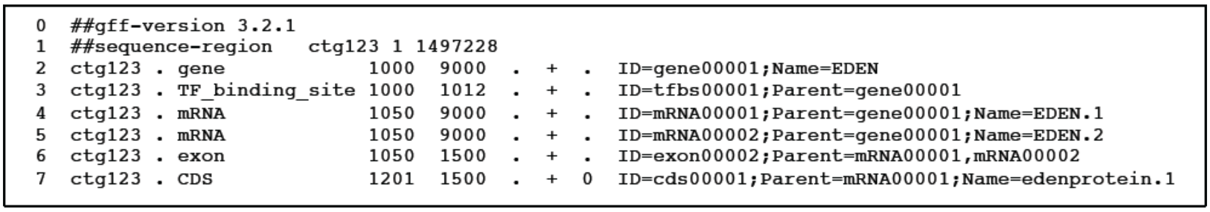
Annotation data

Sequences databases usually contain annotations in addition to sequences.

- Notes and descriptions of important regions and components
- Meta data

Data format of annotation data

Annotation data can be downloaded in many different formats. GFF is one of the popular file formats for storing genomic features.



```
##gff-version 3.2.1
##sequence-region   ctg123 1 1497228
ctg123 . gene             1000  9000  .  +  .  ID=gene00001;Name=EDEN
ctg123 . TF_binding_site  1000  1012  .  +  .  ID=tfbs00001;Parent=gene00001
ctg123 . mRNA             1050  9000  .  +  .  ID=mRNA00001;Parent=gene00001;Name=EDEN.1
ctg123 . mRNA             1050  9000  .  +  .  ID=mRNA00002;Parent=gene00001;Name=EDEN.2
ctg123 . exon             1050  1500  .  +  .  ID=exon00002;Parent=mRNA00001,mRNA00002
ctg123 . CDS              1201  1500  .  +  0  ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
```

Tools

Many database tools are available for various purposes.

Search tools for sequence databases

- BLAST at NCBI (http://blast.ncbi.nlm.nih.gov/Blast.cgi)
- BLAT/BLAST at Ensembl (http://www.ensembl.org/Multi/Tools/Blast)

Data browsing tools of annotation and sequence data

- UCSC Genome Browser (https://genome.ucsc.edu)
- Ensemble Genome Browser (http://www.ensembl.org)

Data download tools for annotation and sequence data

- UCSC Table Browser (http://genome.ucsc.edu/cgi-bin/hgTables)
- Ensemble BioMart (http://www.ensembl.org/biomart)

Tools for protein data

- UniProt (https://www.uniprot.org)

5.2 Search in sequence databases

Since biological databases contain a large number of sequences, heuristics search methods are usually applied to database search.

Aims of searching in sequence databases

- Find homologies
- Find segments with important functionality

Main procedures of sequence search

- Perform local pairwise alignments
- Evaluate the alignments statistically

Estimated computational time for dynamic programming (DP)

Table 5.1: Estimated computational time of DP for the three cases 1ms, 10ms, and 1sec

Time of one alignment	**Database size**		
	1000	**1,000,000**	**1,000,000,000**
1 ms	1 sec	16 min	2.6 h
10 ms	10 sec	2.6 h	11 days
1 sec	16 min	11 days	31 years

Heuristic approach

- Need to search billions of entries
- Tradeoff between accuracy/precision and speed
- Use n-gram based search
- BLAST (Basic Local Alignment Search Tool)
- BLAT (BLAST-like alignment tool)

5.3 BLAST

BLAST (Basic Local Alignment Search Tool) is the most popular tool to find homologous sequences in large-scale sequence databases.

Methods

- Generate n-grams from query sequence
- Find n-gram hits in database
- Expand n-gram hits to HSP
- Increase HSP scores
- Introducing gaps
- Give the expect values (E-values) to HSPs

N-gram hits to HSP

- Connect multiple n-gram hits
- Increase HSP score

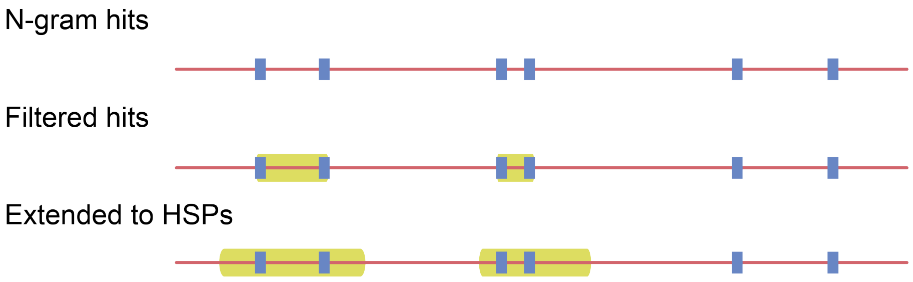


Figure 5.2: N-gram hits to HSPs

Increase HSP score

BLAST changes the length of HSP by shortening or extending in order to increase the score.

Example

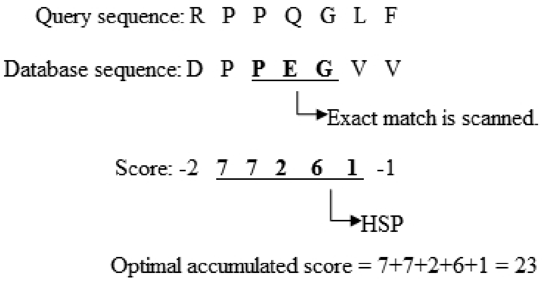

Query sequence: R P P Q G L F
Database sequence: D P P E G V V
Exact match is scanned.
Score: -2 7 7 2 6 1 -1
HSP
Optimal accumulated score = 7+7+2+6+1 = 23

Figure 5.3: HSP extension process (source: DISP, Wikimedia Commons)

Introducing gaps

Banded dynamic programming is used to introduce gaps to an HSP.

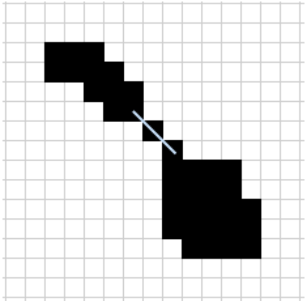


Figure 5.4: Banded DP with the starting seed pair

E-value

“The Expect value (E) is a parameter that describes the number of hits one can expect to see by chance when searching a database of a particular size”

– BLAST Frequently Asked questions (http://blast.ncbi.nlm.nih.gov)

5.4 N-gram based search

Using n-grams is a useful method to find segment pairs.

Equivalent or related concepts to n-gram

- q-gram
- n-letter word
- n-tuple
- n-mer

Create n-grams

Decomposing a given sequence into n-letter words creates a list of n-grams.

Example

```
q: ACGATT

Word size: 2
    AC, CG, GA, AT, TT

Word size: 3
    ACG, CGA, GAT, ATT
```

Find segment pairs in database sequences

N-grams can be used to find segment pairs.

Example

```
q: ACGATT
2-gram: AC, CG, GA, AT, TT

d1: CTAAG
0 hit

d2: CGTAT
2 hits

d3: ATAGA
2 hits
```

5.5 Lookup table of matching n-grams

A lookup table can be used for effectively finding n-gram matches.

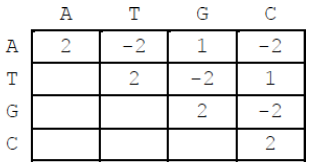
Terminology

- Indices: positions in q
- Matching n-grams: Possible matching n-grams by threshold score T

Example of creating a lookup table

```
q: ACGTAC
2-gram: AC, CG, GT, TA, AC
T: 3
```

Score matrix:



	A	T	G	C
A	2	-2	1	-2
T		2	-2	1
G			2	-2
C				2

Step 1. Index of q

Add indices to all n-grams.

Index	N-gram
1	AC
2	CG
3	GT
4	TA
5	AC

Step 2. Scores of segment pairs and matching n-grams

Calculate scores between the first n-gram AC and all its matching n-grams.

N-gram	Matching n-gram	Score
AC	AA	$2 + (-2) = 0$
AC	AC	$2 + 2 = 4$
AC	AG	$2 + (-2) = 0$
AC	AT	$2 + 1 = 3$
AC	CA	$(-2) + (-2) = -4$
AC	CC	$(-2) + 2 = 0$
AC	CG	$(-2) + (-2) = -4$
AC	CT	$(-2) + 1 = -1$
AC	GA	$1 + (-2) = -1$
AC	GC	$1 + 2 = 3$
AC	GG	$1 + (-2) = -1$
AC	GT	$1 + 1 = 2$
AC	TA	$(-2) + (-2) = -4$
AC	TC	$(-2) + 2 = 0$
AC	TG	$(-2) + (-2) = -4$
AC	TT	$(-2) + 1 = -1$

Use threshold $T = 3$.

N-gram	Matching n-grams	Scores
AC	AC, AT, GC	4, 3, 3

Repeat the same procedure for all n-grams of q and add their indices.

Index	N-gram	Matching n-grams	Scores
1	AC	AC, AT, GC	4, 3, 3
2	CG	CG, TG, CA	4, 3, 3
3	GT	GT, AT, GC	4, 3, 3
4	TA	TA, CA, TG	4, 3, 3
5	AC	AC, GC, AT	4, 3, 3

Step 3. Lookup table of matching n-grams

Transform the table above to create a lookup table of matching n-grams.

Matching n-gram	Indices of q	Scores of segment pairs
AC	1, 5	4, 4
GC	1, 3, 5	3, 3, 3
AT	1, 3, 5	3, 3, 3
CG	2	4
TG	2, 4	3, 3
CA	2, 4	3, 3
GT	3	4
TA	4	4

Step 4. Search

```
d1: AAAGTG

2 hits
GT  index: 3, score: 4
TG  index: (2, 4), score: (3, 3)
```

Exercise 5.1

Create a lookup table of 2-grams with the indices of q and the scores of segment pairs. Use the threshold T and pre-calculated scores of 2-gram segment pairs.

```
q: CATG
T: 3
```

The table below shows pre-calculated scores of 2-gram segment pairs.

Matching n-gram	**N-gram**		
	CA	**AT**	**TG**
AA	0	0	-1
AC	-4	3	-4
AG	-1	0	0
AT	-4	4	-4
CA	4	-4	2
CC	0	-1	-1
CG	3	-4	3
CT	0	0	-1
GA	0	-1	-1
GC	-4	2	-4
GG	-1	-1	0
GT	4	3	-4
TA	3	-4	3
TC	-1	-1	0
TG	2	-4	4
TT	-1	0	0

5.6 Finite-state machine with n-grams

Finite-state machine enables efficient database search by expanding the basic n-gram based search.

Number of potential matching n-grams

The number of potential n-grams increases by the alphabet size and the word size.

DNA
$C = \{A, C, G, T\}$
Word size 2 → $4^2 = 16$

Word size 3 $\rightarrow 4^3 = 64$
Word size 12 $\rightarrow 4^{12} =$ 16,777,216

Protein
C = {A, R, N, D, C, Q, E, G, H, I, L, L, M, F, P, S, T, W, Y, V}
Word size 2 $\rightarrow 20^2 = 400$
Word size 3 $\rightarrow 20^3 = 8000$

Finite-state machine

A finite-state machine can be used to scan database sequences instead of using a lookup table. Finite-state machines are usually faster than lookup tables.

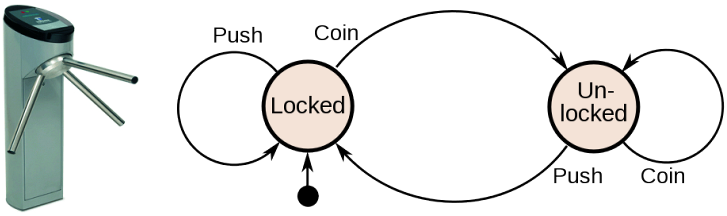


Figure 5.5: Finite-state machine for coin-operated turnstile
(sources: Chetvorno and Sebasgui via Wikimedia Commons)

Example of creating a finite-state machine

```
q: ACGTAC, Word size: 2, T: 3
```

Lookup table

Matching n-gram	Indices of q	Scores of segment pairs
AC	1, 5	4, 4
GC	1, 3, 5	3, 3, 3
AT	1, 3, 5	3, 3, 3
CG	2	4
TG	2, 4	3, 3
CA	2, 4	3, 3
GT	3	4
TA	4	4

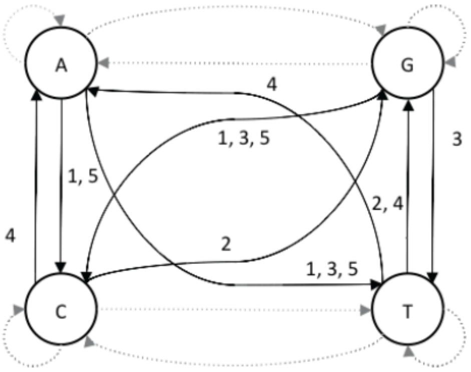


Figure 5.6: Finite-state machine to output the indices of 2-grams

```
d1: AAAGTG

2 hits
GT  index: 3
TG  index: (2, 4)
```

Exercise 5.2

Create a finite-state machine and use it to find a segment pair.

1. Create a finite-state machine for the lookup table for q: ACGTAC. Add both indices and scores to the edges.

   Lookup table

Matching n-gram	Indices of q	Scores of segment pairs
AC	1, 5	2, 2
CG	2	4
GT	3	2
TA	4	0

2. Use the finite-state machine and find a segment pair between q and d: AAAGTG.