Crash course on Computational Biology for Computer Scientists

Bartek Wilczyński
bartek@mimuw.edu.pl
http://regulomics.mimuw.edu.pl

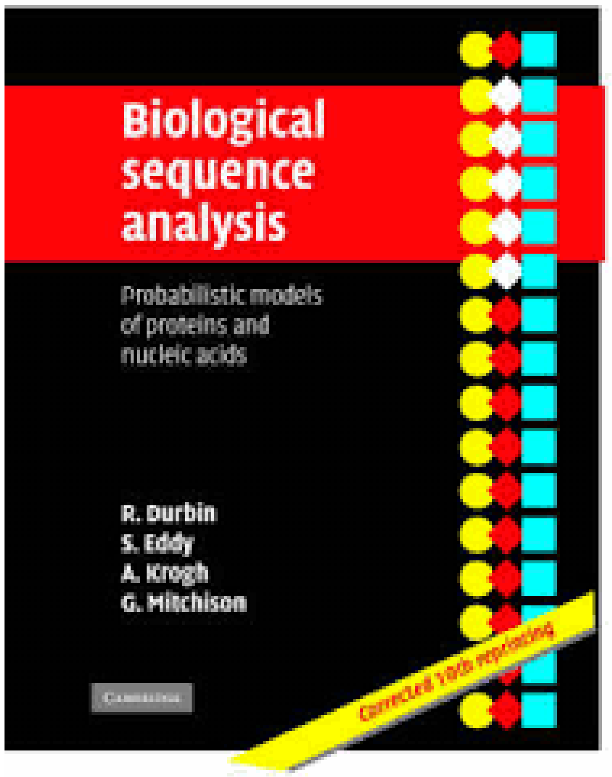
Phd Open lecture series

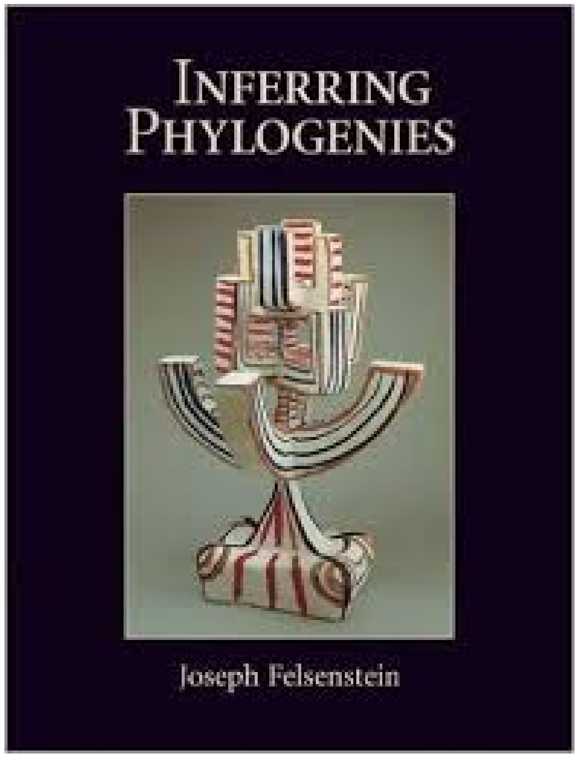
17-19 XI 2016

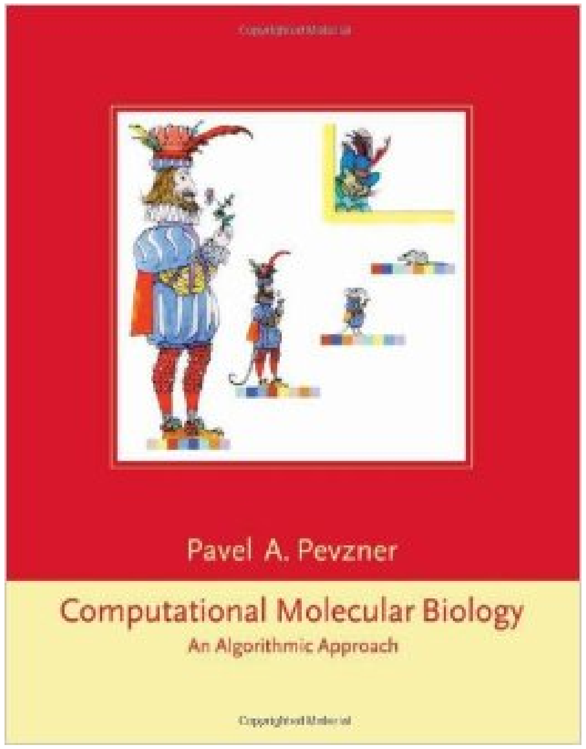
Topics for the course

- Sequences in Biology – what do we study?
- Sequence comparison and searching – how to quickly find relatives in large sequence banks
- Tree-of-life and its construction(s)
- Short sequence mapping – where did this word come from
- DNA sequencing and assembly – puzzles for experts
- Sequence segmentation – finding modules by flipping coins
- Data storage and compression – from DNA to bits and back again
- Structures in Biology – small and smaller

Books to read more


Biological sequence analysis
Probabilistic models of proteins and nucleic acids
R. Durbin
S. Eddy
A. Krogh
G. Mitchison
Corrected 10th reprinting



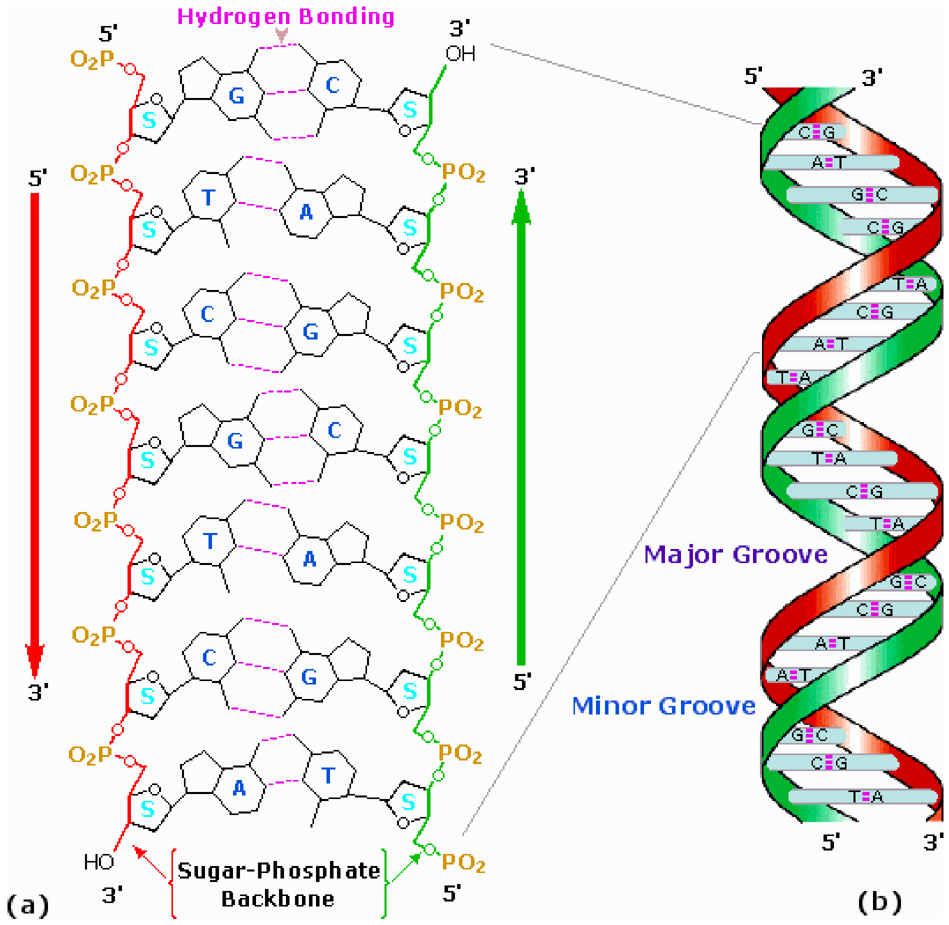


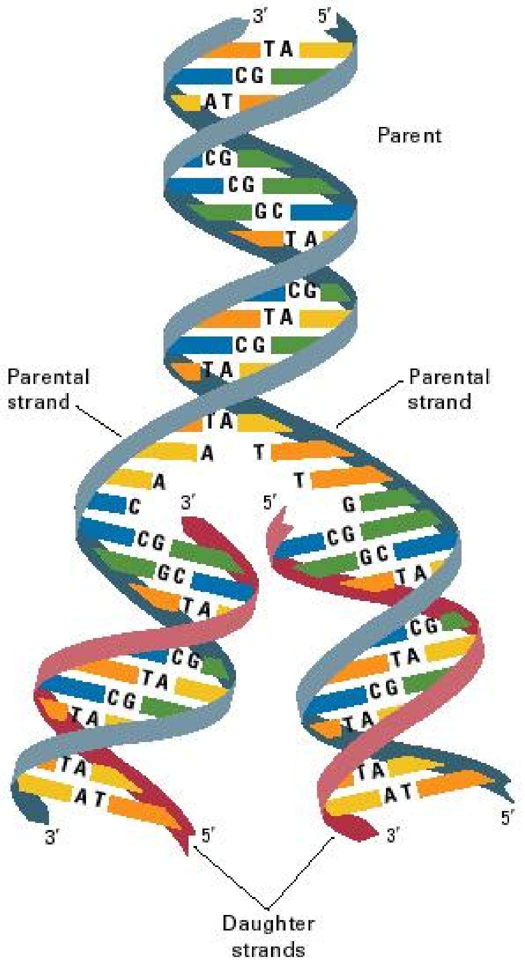
Norbert Dojer slides on Genome Scale Technologies 2 course

How to make it efficient

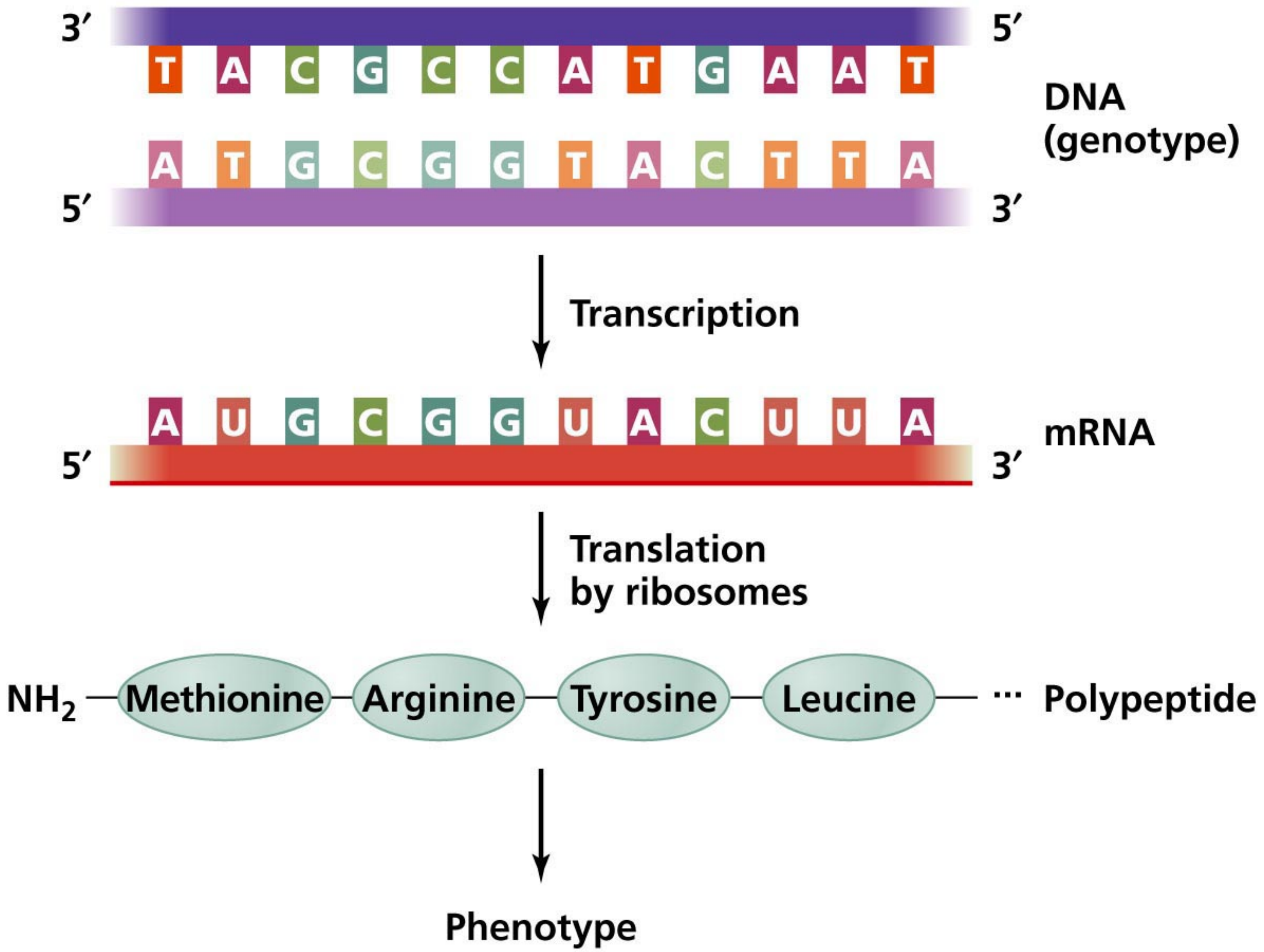
- Diverse audience, I don’t know what you know
- Please **do** interrupt me if you have a question!
- I will not go very deeply into biological details, so if you want more, please ask me later for links to more materials
- I will not go deeply into proofs or derivations, so if you want more, please ask me later for links to more materials
- If you need to ask later: bartek@mimuw.edu.pl

DNA structure





The DNA is not the only sequence

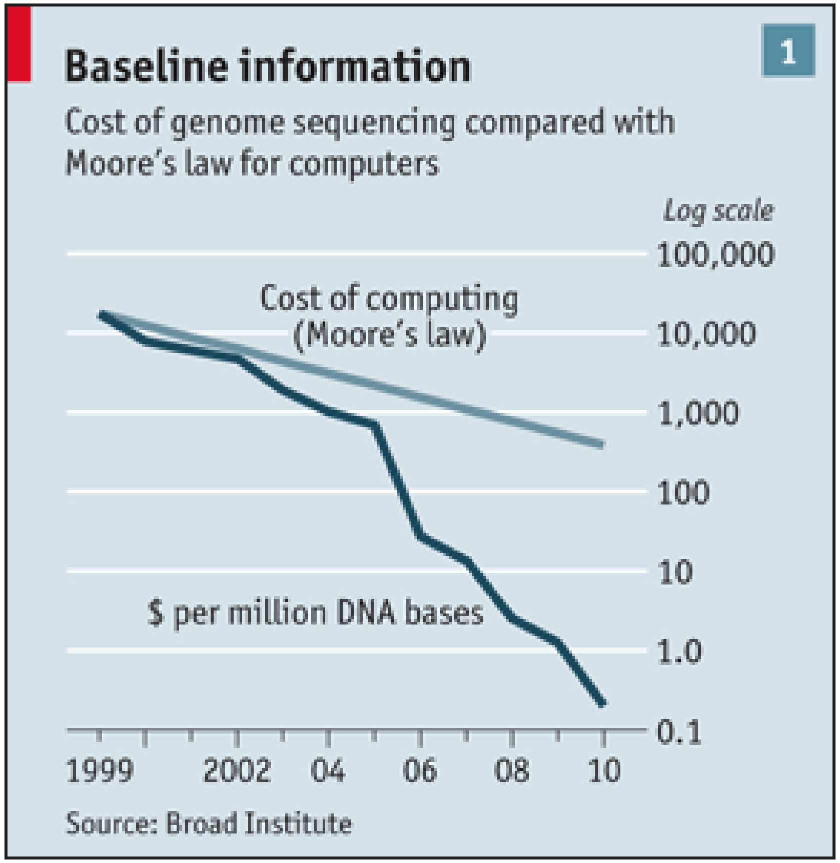


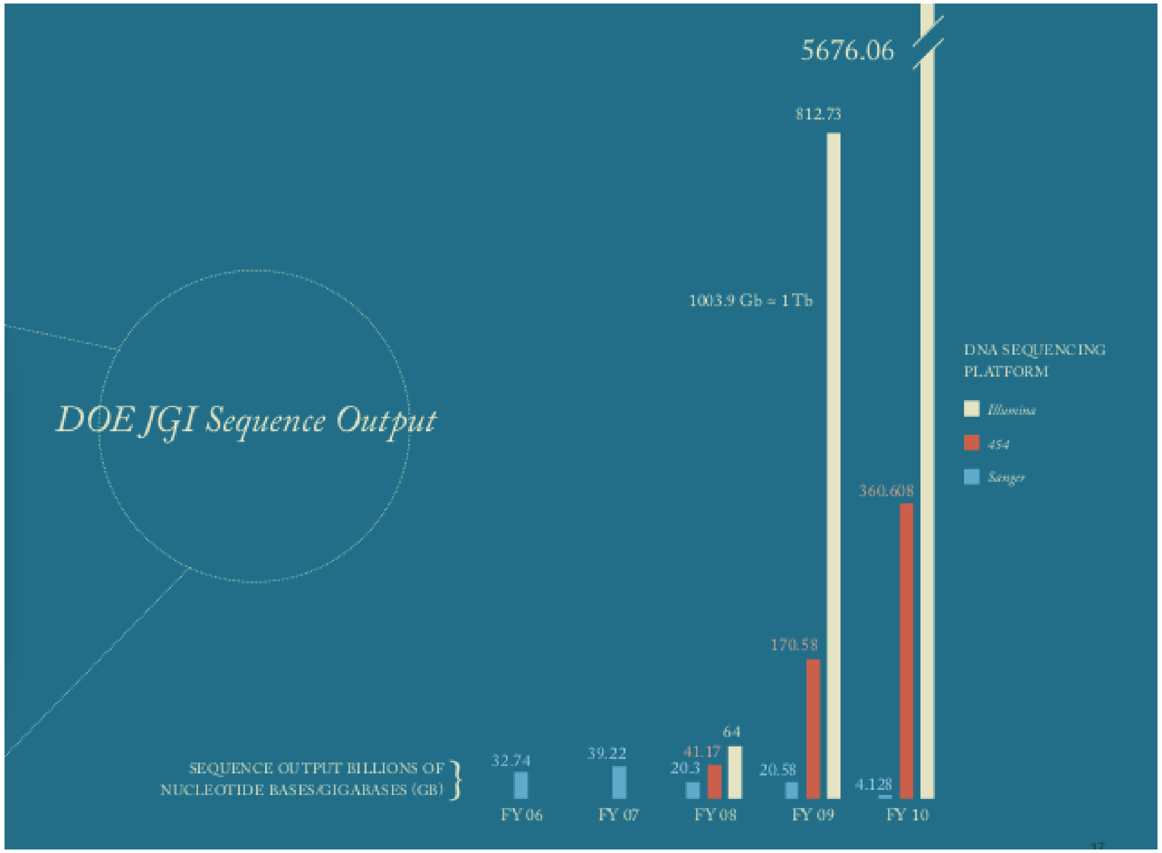
Finding related sequences

- Assume we have a new sequence of a previously unknown species (a new virus, bacteria, etc).
- Can find find its closest relative in the database of known DNA sequences?
- How quickly can this be done?

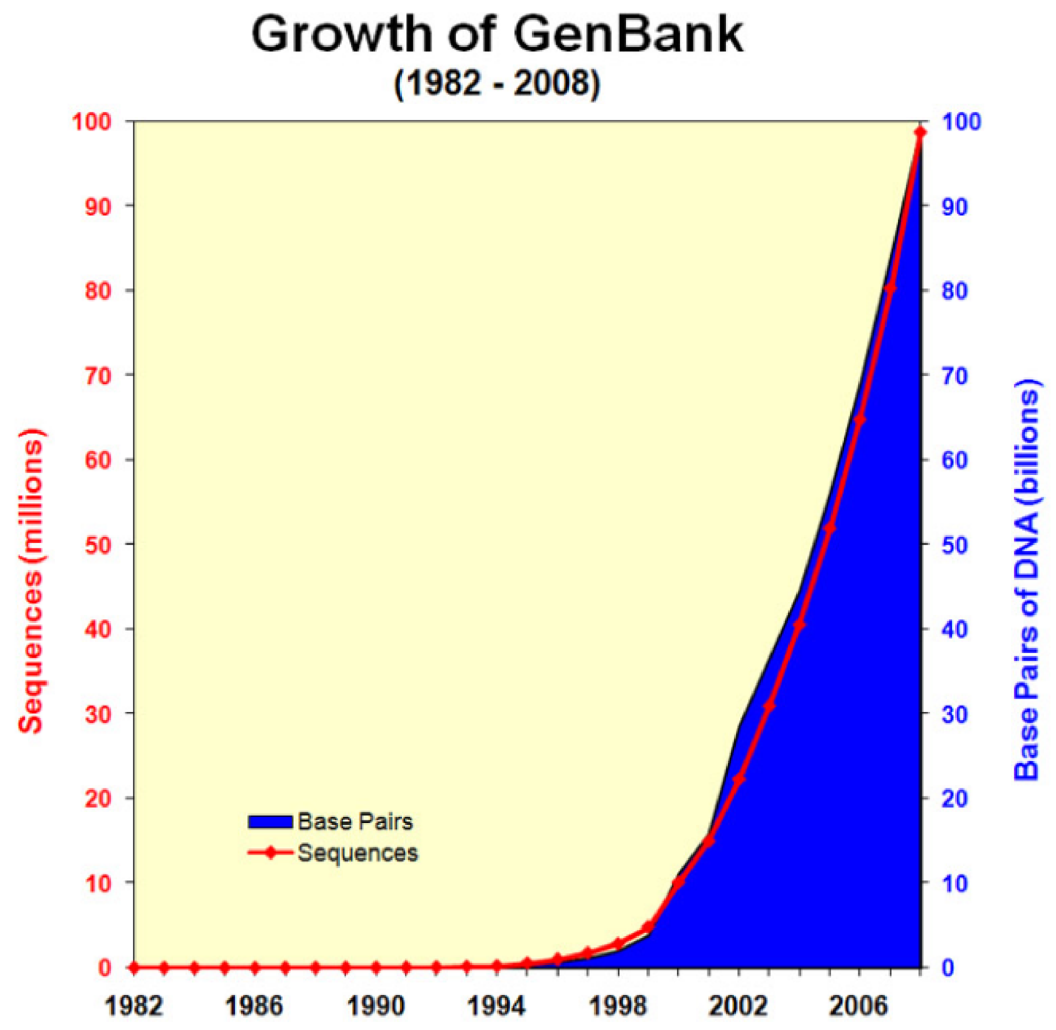
The growing problem

- The cost of sequencing is decreasing exponentially and the throughput is increasing





Naturally databases grow too...



Genbank non-redundant nucleotide count is now $\geq 10^{11}$ and sequence count $\geq 10^8$. image source NIH NCBI release notes

What do we know from yesterday?

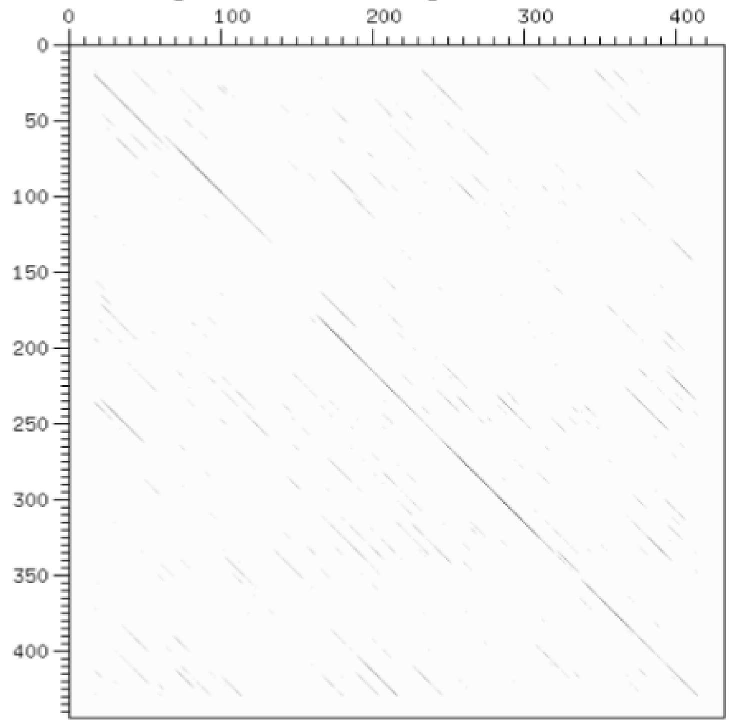
- Indeed, we can find similar sequences by comparing them with local sequence alignment methods
- Such algorithms run in $\mathcal{O}(n \cdot m)$ time scale
- How much would a Smith-Waterman analysis of a single new sequence (1000bp) against genbank take?
- How long for a genome with 10 thousand genes?
- How long for the JGI annual throughput?
- Can we wait that long?
- Can it be done faster?
- What assumptions do we need to make?

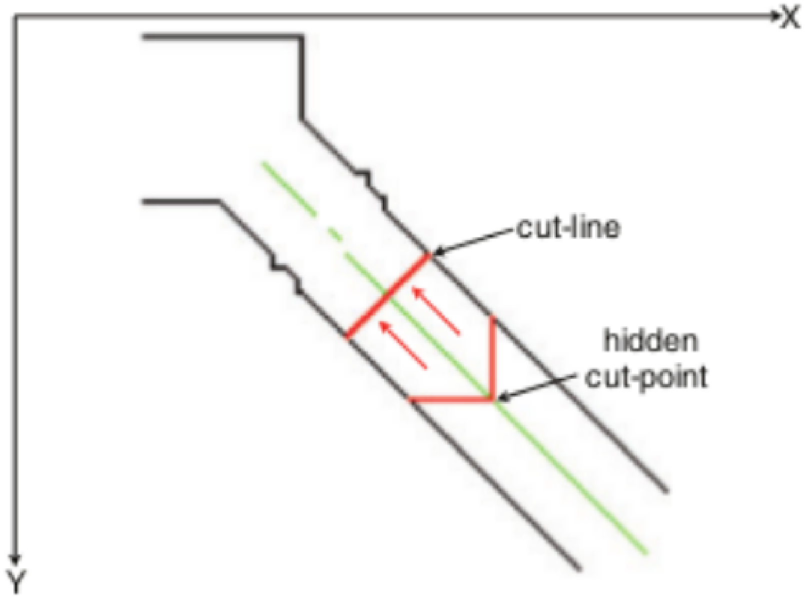
Reversing the nearest sequence problem

- We are looking only for **similar** sequences in the database, so most of our work with S-W algorithm is comparing sequences which will not show up in the result
- Can we tell if a sequence is *not-similar* more quickly than S-W?
- We need to define a meaningful way of specifying our definition of *not-similar*
- **We need an algorithm that can reject bad alignments based on a meaningful and computable criteria**

Near diagonal in DP matrix?

Good global alignments reside close to the diagonal

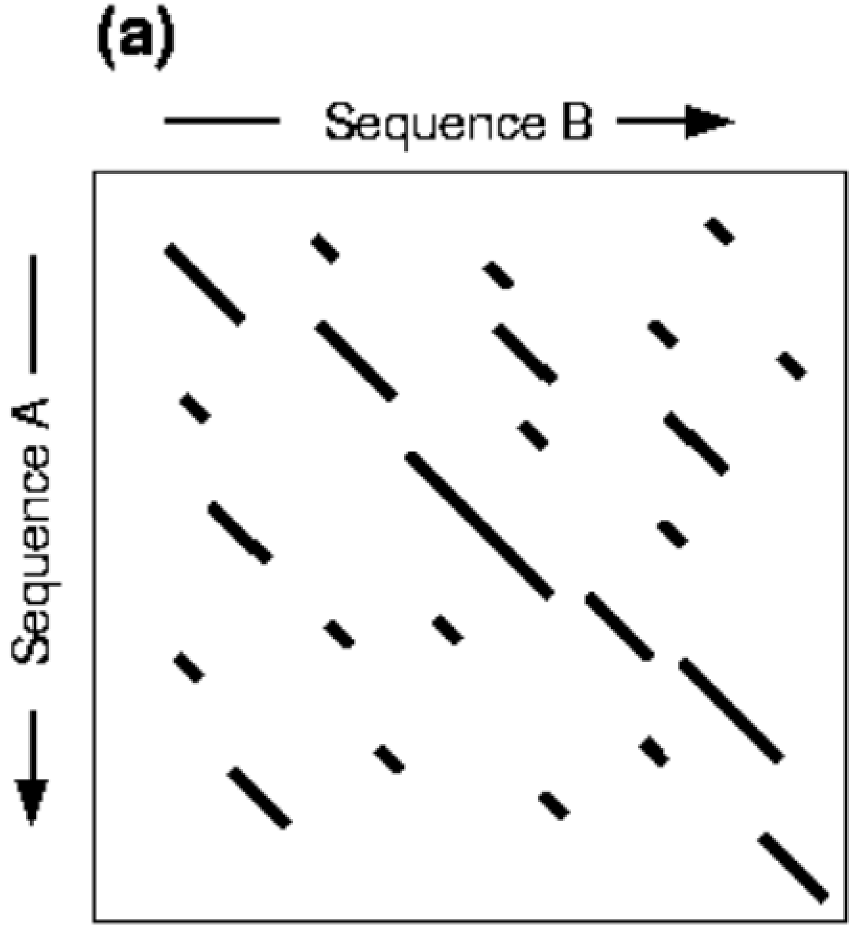




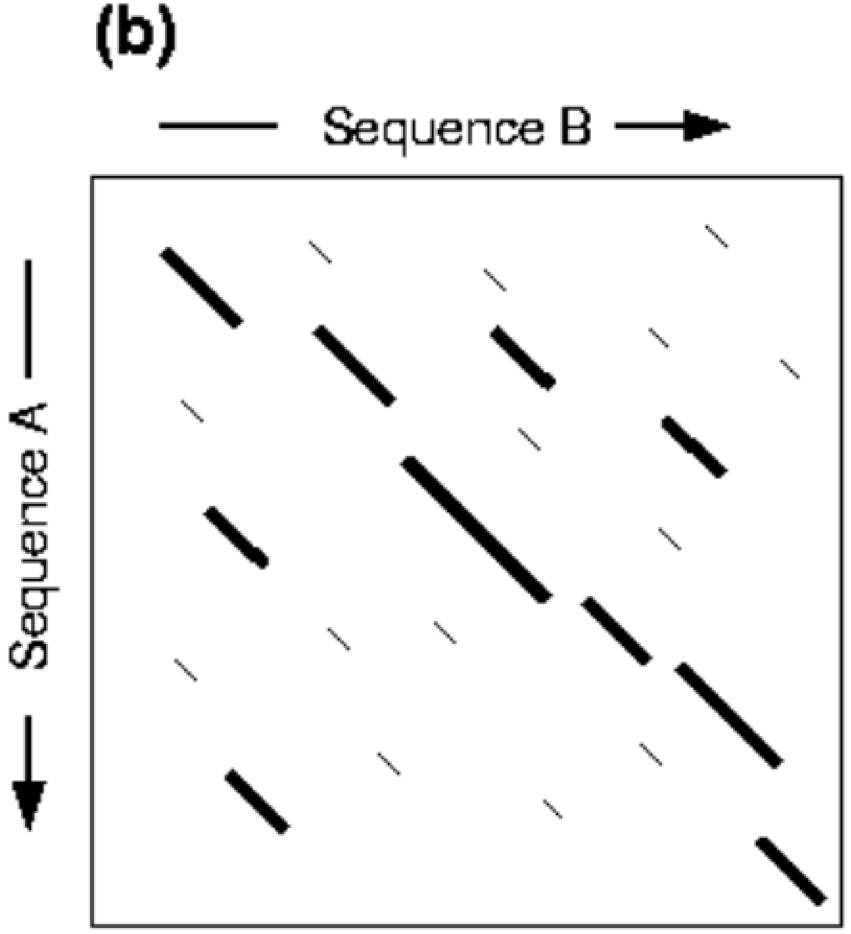
- Restricting to search within fixed distance from diagonal brings our computing time to almost linear
- but **not for local alignments**

image source: pecan algorithm

FASTA search for short ID matches



Find runs of identities

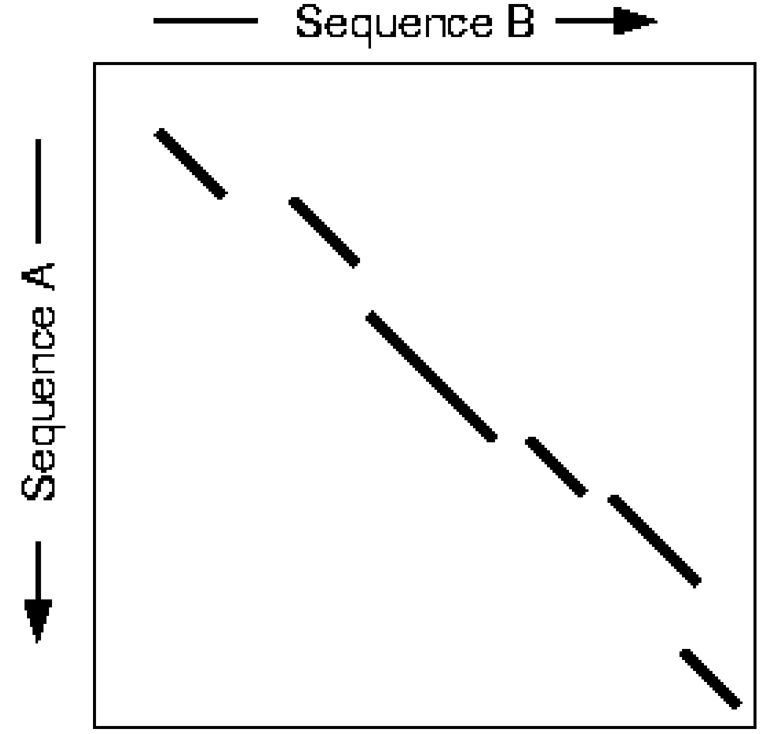


Re-score using PAM matrix
Keep top scoring segments.

image source GJ Barton

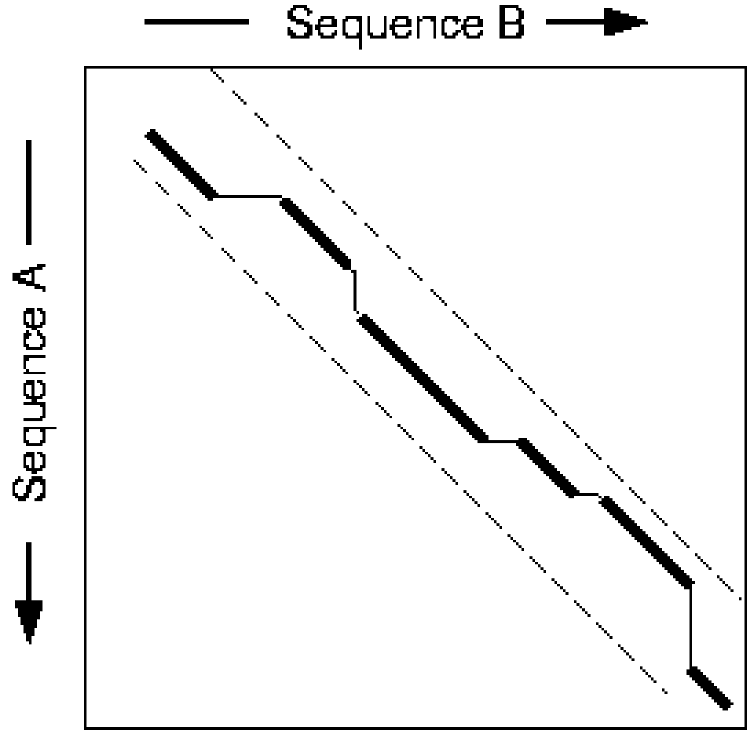
Improve on this idea...

(c)



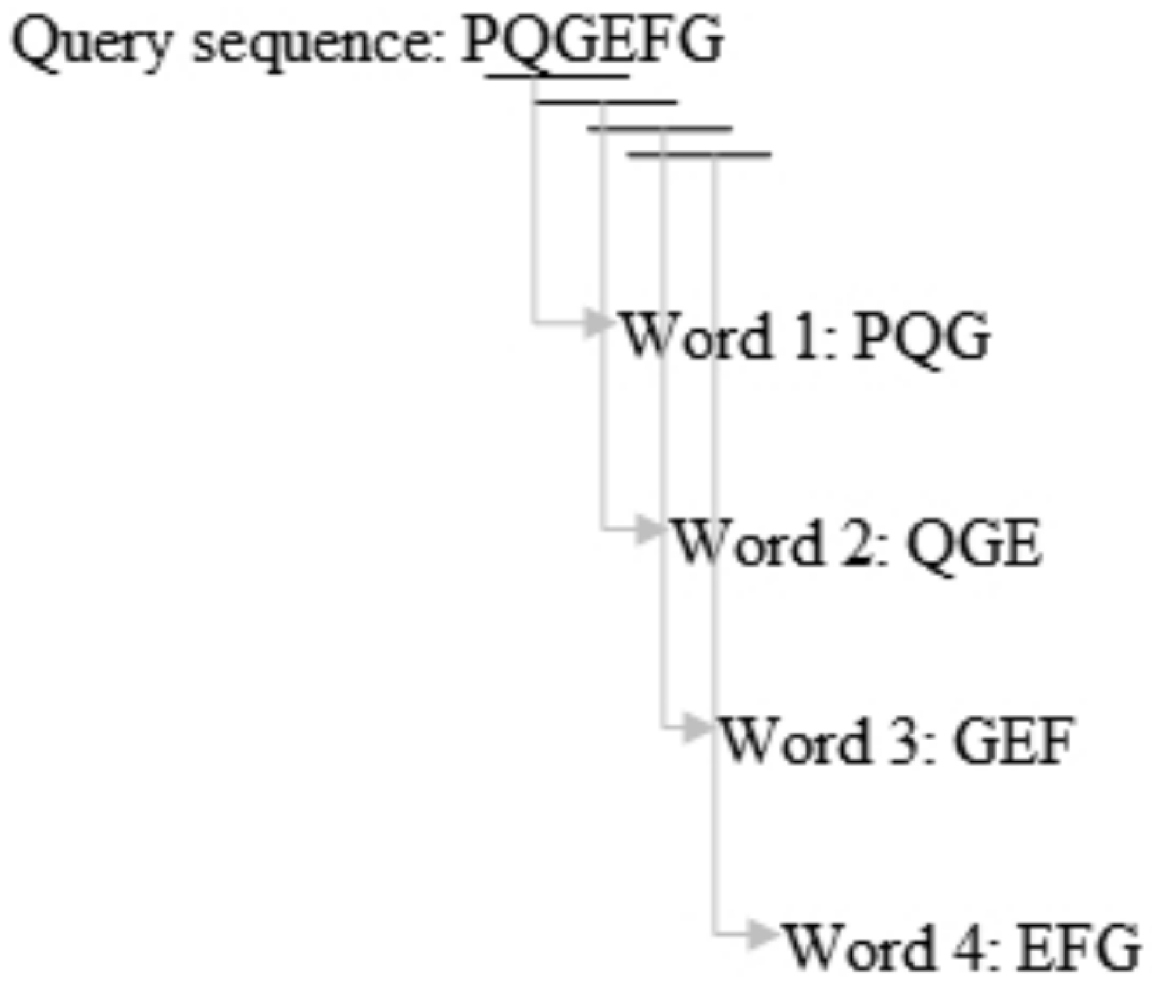
Apply "joining threshold"
to eliminate segments that
are unlikely to be part of the alignment
that includes highest scoring segment.

(d)

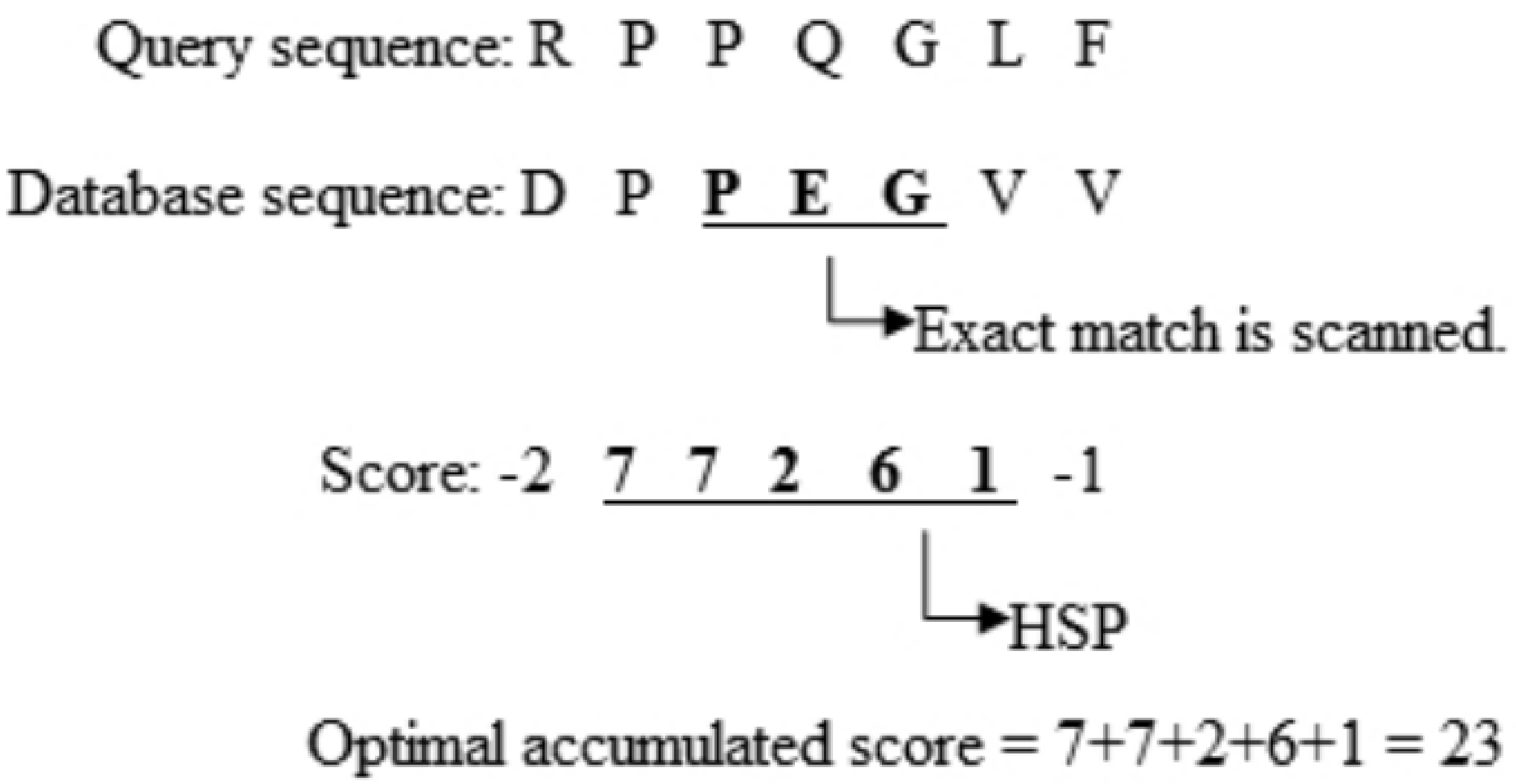


Use dynamic programming
to optimise the alignment in a
narrow band that encompasses
the top scoring segments.

Hashing words similar to the query



Extending words to segments



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

image source wikipedia

High scoring segment pairs (HSP)

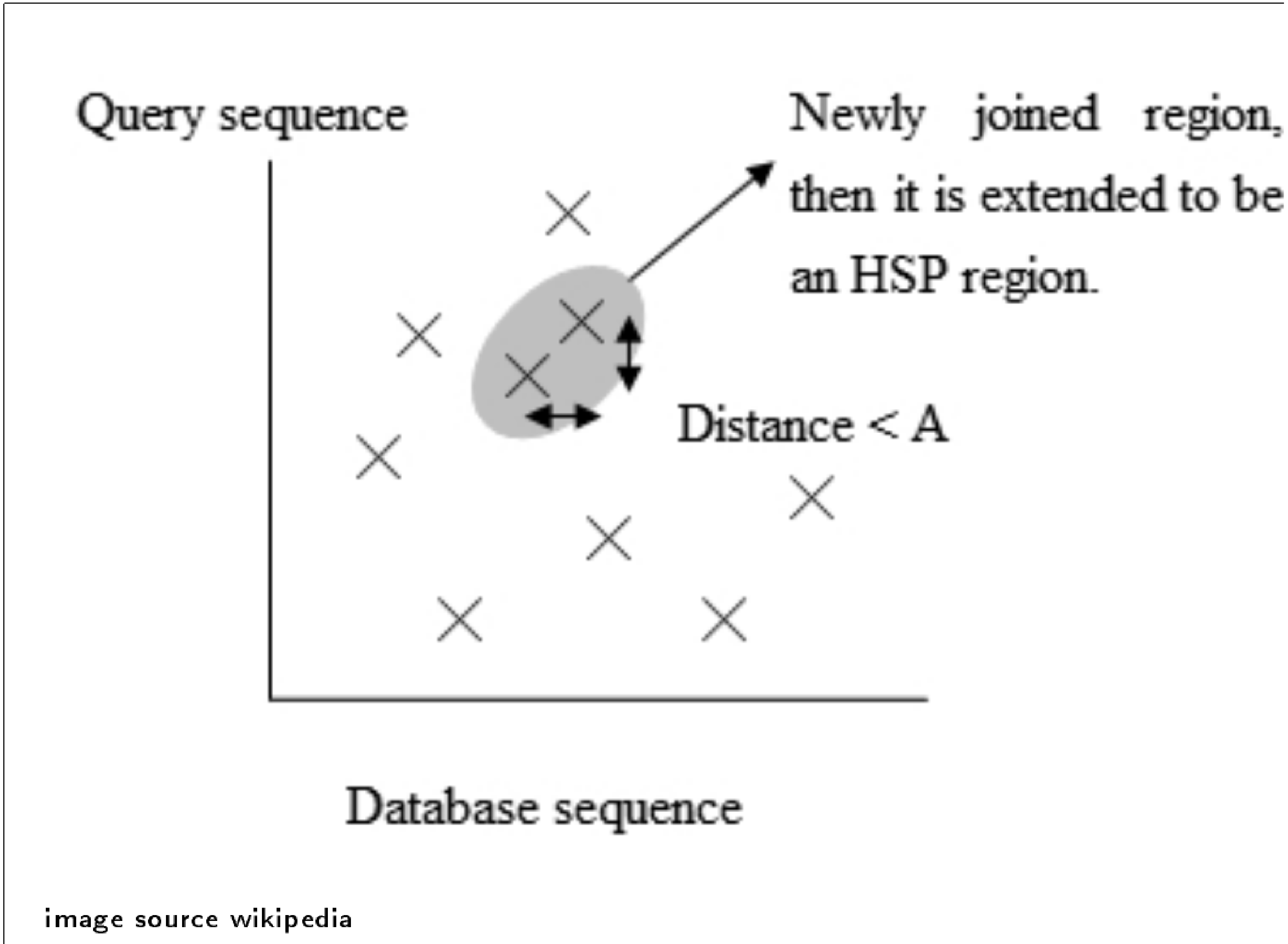
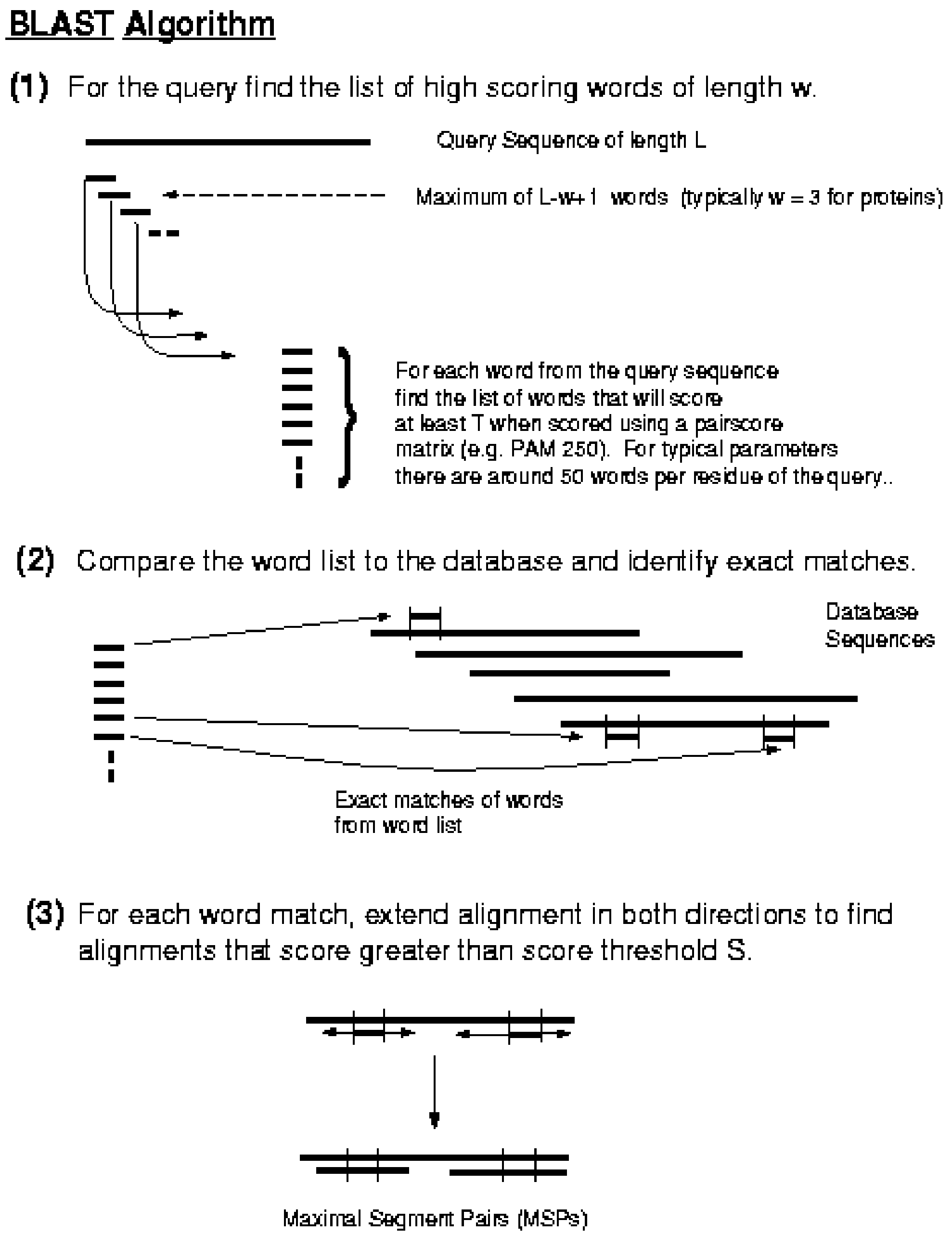


image source wikipedia

Complete BLAST algorithm

- Basic Local Alignment Search Tool
- Hashing words similar to query
- Finding pairs of matches to the same sequence
- Searching for Maximal Segment Pairs among HSPs


BLAST Algorithm
(1) For the query find the list of high scoring words of length w.
Query Sequence of length L
Maximum of L-w+1 words (typically w = 3 for proteins)
For each word from the query sequence find the list of words that will score at least T when scored using a pairscore matrix (e.g. PAM 250). For typical parameters there are around 50 words per residue of the query..
(2) Compare the word list to the database and identify exact matches.
Database Sequences
Exact matches of words from word list
(3) For each word match, extend alignment in both directions to find alignments that score greater than score threshold S.
Maximal Segment Pairs (MSPs)

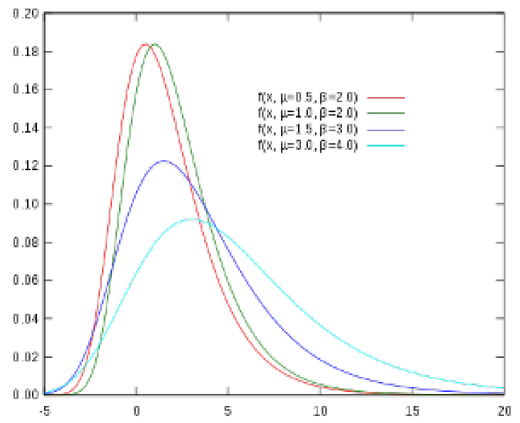
Looking for rare findings

- Assume that you found an HSP, is it worth keeping it in the result?
- Behave like a collector: it's only worth keeping if it is rare
- Formally, we want matches which are **ulikely to occur by random** in similar situations (defined by size and composition of the query and database)
- In statistics, we are performing **hypothesis testing**: under **null hypothesis**, there are no matching sequences in the database
- We are interested in the probability of observing a given score (or higher) under assumption of the null model

BLAST E-values

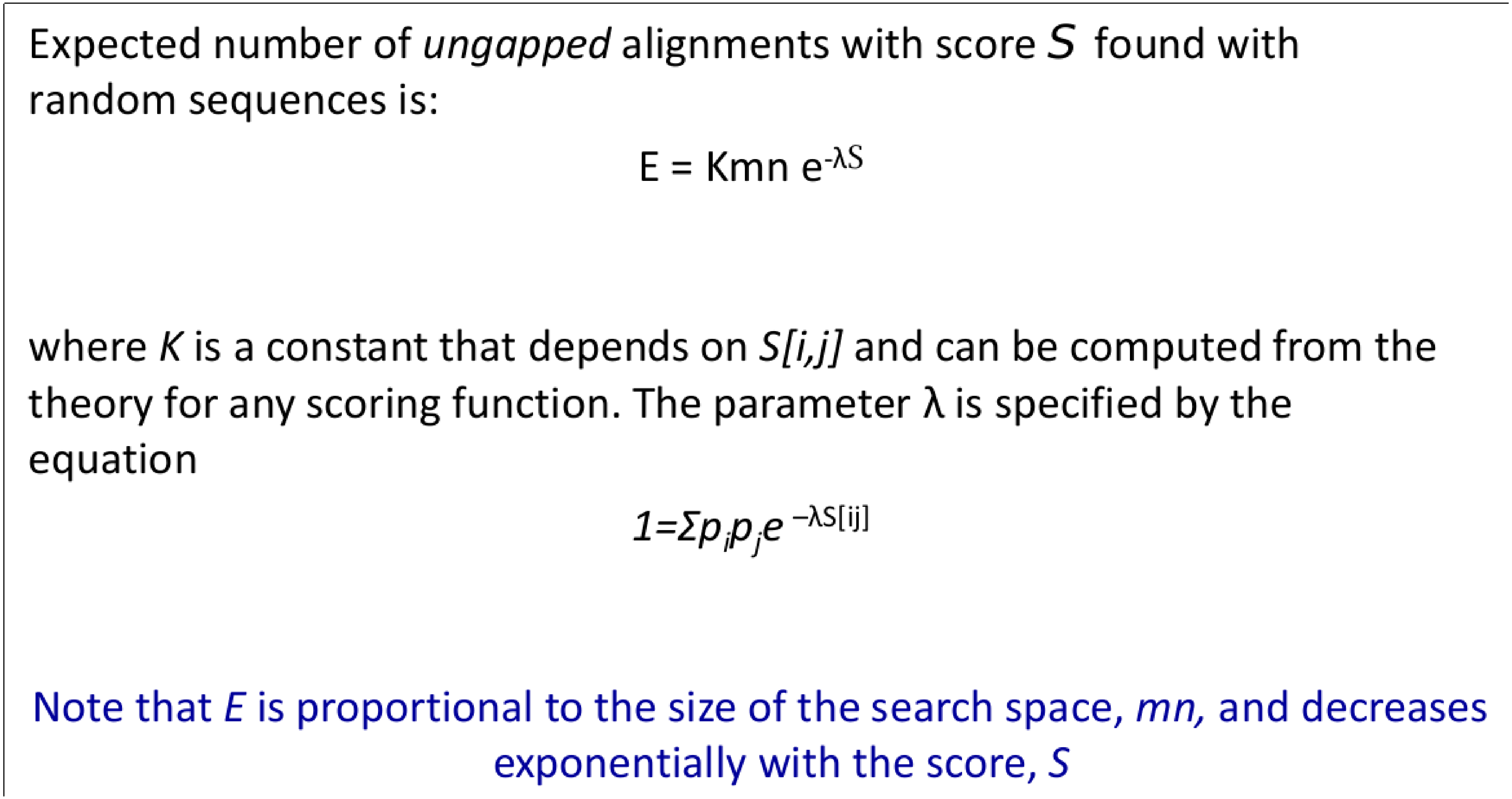
- We cannot really estimate this probability by Monte-Carlo (data is too large for large-scale sampling)
- It is assumed, that it should follow the extreme value distribution (Gumbel distribution)

$$p(s \geq x) = 1 - \exp(-e^{-\lambda(x-\mu)}), \mu = \frac{log(Km'n')}{\lambda}$$



parameters K and λ can be estimated from data, then the E-value is computed $E = pD$, where D is the number of sequences in the database (similar to Bonferroni correction)

Altschul Karlin 1990



Expected number of *ungapped* alignments with score S found with random sequences is:

$$E = Kmn\, e^{-\lambda S}$$

where *K* is a constant that depends on *S[i,j]* and can be computed from the theory for any scoring function. The parameter λ is specified by the equation

$$1=\Sigma p_i p_j e^{-\lambda S[ij]}$$

Note that *E* is proportional to the size of the search space, *mn,* and decreases exponentially with the score, *S*

Target frequencies

Given sequences *a* and *b:*

- Alternate hypothesis ($\hat{H}_a$)*: a* and *b* are related at *n* PAMs divergence.
  - Residues *i* and *j* are aligned with "target" frequencies, $q^n{}_{ij}$
- Null hypothesis ($\hat{H}_0$)*: a* and *b* are unrelated.
  - Residues *i* and *j* are aligned with background frequencies, $p_i p_j$

Note that the PAM and BLOSUM matrices were constructed by estimating q_{ij} from data. However, any scoring matrix (that satisfies the appropriate assumptions for Karlin Altschul statistics) can be expressed as a log odds matrix of the form

$$S^n[i,j] = \log_2 \frac{q^n ij}{p_i p_j}$$

The frequencies q_{ij} in the above equation are the characteristic target frequencies of the matrix *S[]* . In other words, q_{ij} is the frequency with which *i* is aligned with *j* in Maximal Segment Pairs (MSPs) obtained with *S[].* Recall that an MSP is "the highest scoring pair of identical length segments chosen from two sequences. The boundaries of an MSP are chosen to maximize its score, so an MSP may be of any length."[1]

[1]Altschul *et al.* J Mol Biol 215: 403-10 (1990

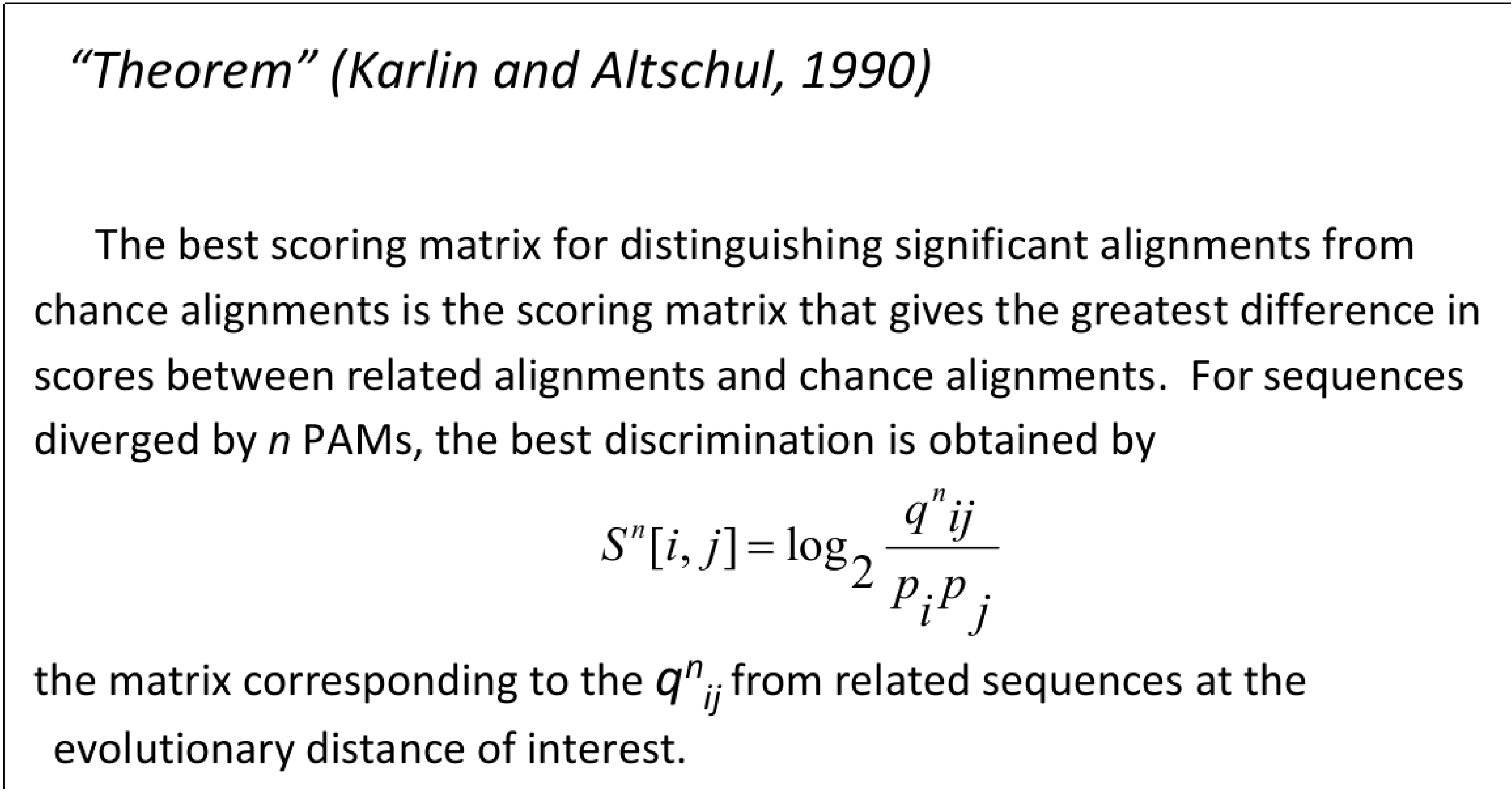
Target frequencies for substitution matrix *S[]*can be estimated empirically as follows:

- Generate "random" sequences from background probabilities
- Find MSPs in pairs of random sequences using *S[]* to score alignments
- Count target frequencies in those MSPs

Target frequencies can also be estimated theoretically using the equation:

$$q^{n}{}_{ij} = p_i p_j e^{-\lambda S^{n}[i,j]}$$

We can choose the best matrix



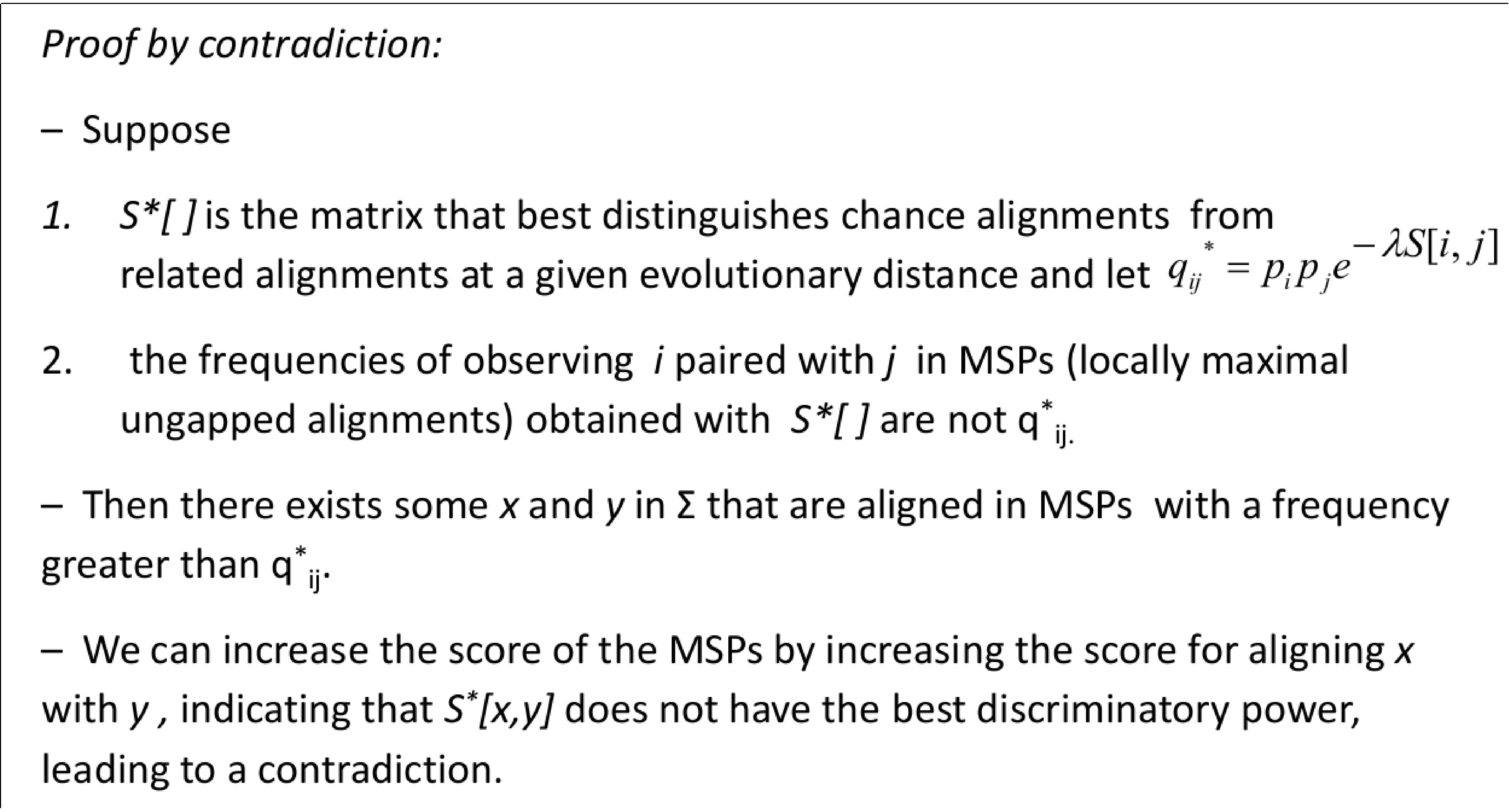
"Theorem" (Karlin and Altschul, 1990)

The best scoring matrix for distinguishing significant alignments from chance alignments is the scoring matrix that gives the greatest difference in scores between related alignments and chance alignments. For sequences diverged by *n* PAMs, the best discrimination is obtained by

$$S^n[i,j] = \log_2 \frac{q^n ij}{p_i p_j}$$

the matrix corresponding to the $q^n{}_{ij}$ from related sequences at the evolutionary distance of interest.

“proof” of the “theorem”



Proof by contradiction:

– Suppose

1. *S*[ ]* is the matrix that best distinguishes chance alignments from related alignments at a given evolutionary distance and let $q_{ij}^{*} = p_i p_j e^{-\lambda S[i,j]}$

2. the frequencies of observing *i* paired with *j* in MSPs (locally maximal ungapped alignments) obtained with *S*[ ]* are not $q^{*}_{ij.}$

– Then there exists some *x* and *y* in Σ that are aligned in MSPs with a frequency greater than q^{*}_{ij}.

– We can increase the score of the MSPs by increasing the score for aligning *x* with *y* , indicating that *S^*[x,y]* does not have the best discriminatory power, leading to a contradiction.

Implications

BLAST will give reasonable accuracy as long as the empirical target frequencies, $q_{ij,}$, in the alignments of interest do not deviate too far from the theoretical target frequencies:

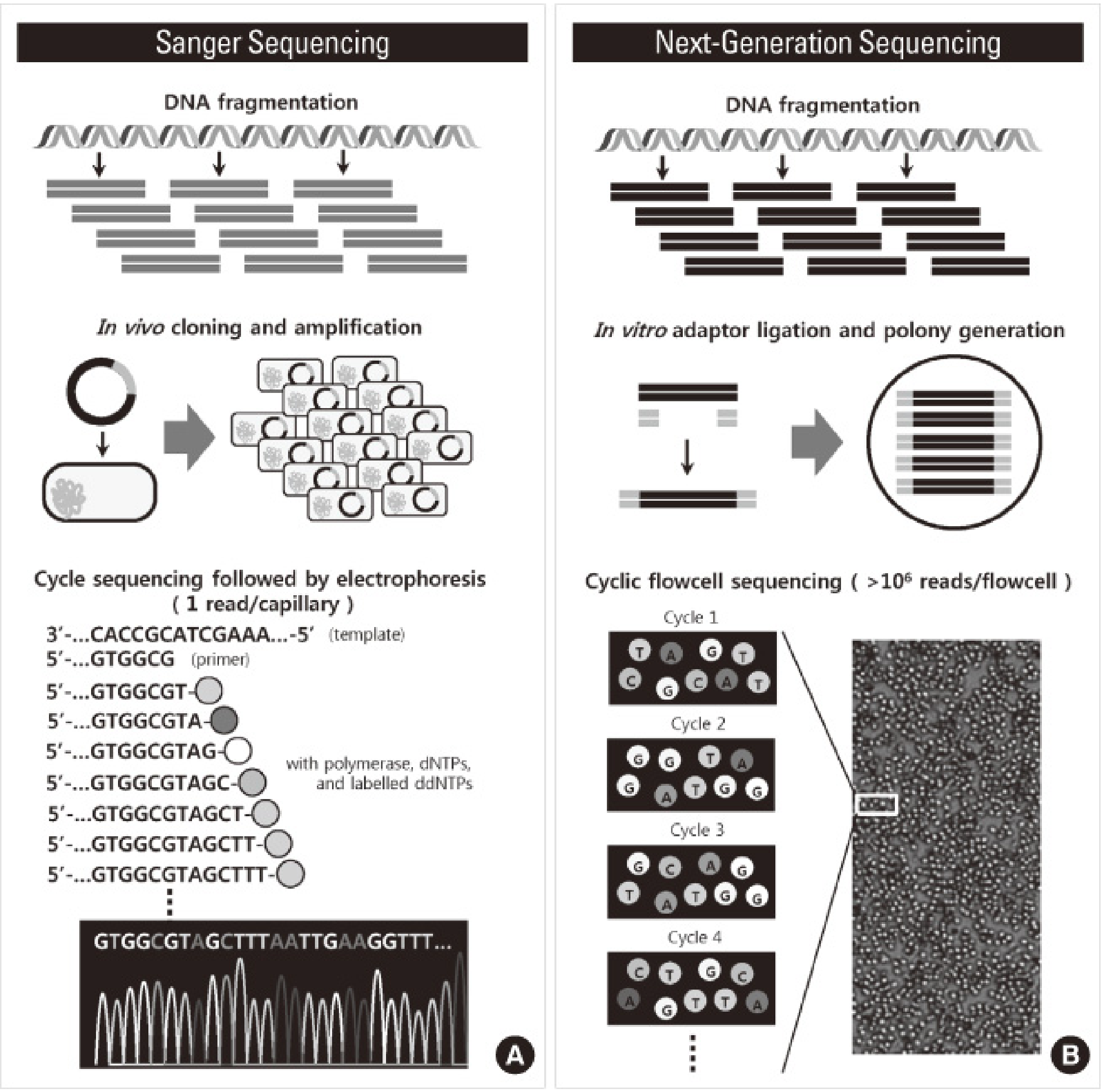
$$\bar{q}_{ij} = p_i p_j e^{-\lambda S[i,j]}$$

Reasonable accuracy can be achieved with two or three matrices.

BLAST summary

- Sufficiently fast heuristic approach
- Smart approach to the problem allows linear speedup of the result
- Heuristic based on statistical reasoning, but not using statistical model as in the rigorous manner
- Currently the most popular bioinformatical tool

Next Generation Sequencing



Sanger Sequencing
DNA fragmentation
In vivo cloning and amplification
Cycle sequencing followed by electrophoresis (1 read/capillary)
3′-...CACCGCATCGAAA...-5′ (template)
5′-...GTGGCG (primer)
5′-...GTGGCGT
5′-...GTGGCGTA
5′-...GTGGCGTAG
5′-...GTGGCGTAGC
5′-...GTGGCGTAGCT
5′-...GTGGCGTAGCTT
5′-...GTGGCGTAGCTTT
with polymerase, dNTPs, and labelled ddNTPs
GTGGCGTAGCTTTAATTGAAGGTTT...
A
Next-Generation Sequencing
DNA fragmentation
In vitro adaptor ligation and polony generation
Cyclic flowcell sequencing (>10⁶ reads/flowcell)
Cycle 1
Cycle 2
Cycle 3
Cycle 4
B

- NGS gives millions of short reads (30-200bp) instead of 1 longer read (up to few kb)
  - Desk-size devices,
  - costly chemistry (in 1000$ range for ~1TB of data)
  - error rates ~0.0001

Single molecule sequencing

- Oxford nanopore MiniION on the ISS (Aug 2016)

- Single molecule sequencing is in the prototype phase – gives even longer reads (up to 100kb), but with large error rate (~10%)
- Small devices for single used are promised to cost below 1000$

How to map a short sequence to the genome?

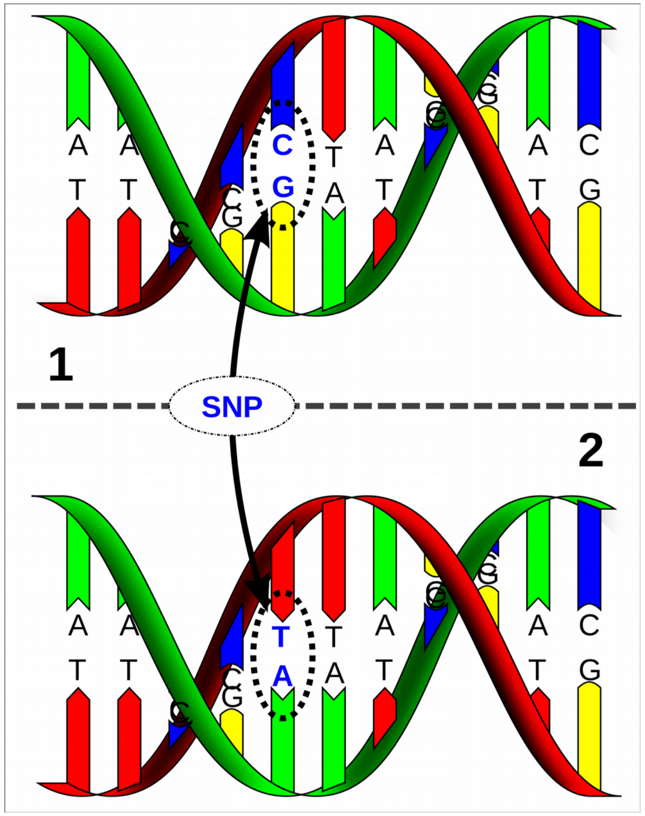
- We frequently sequence DNA originating from a genome closely related to a known one (e.g. human patient samples, bacteria, viruses, etc)
- Even though they are closely related, they are not identical (remember, mutations?)
- Sequence reads are short (30-100), genomes are long (up to 10^10)
- Obviously we need faster methods than DP

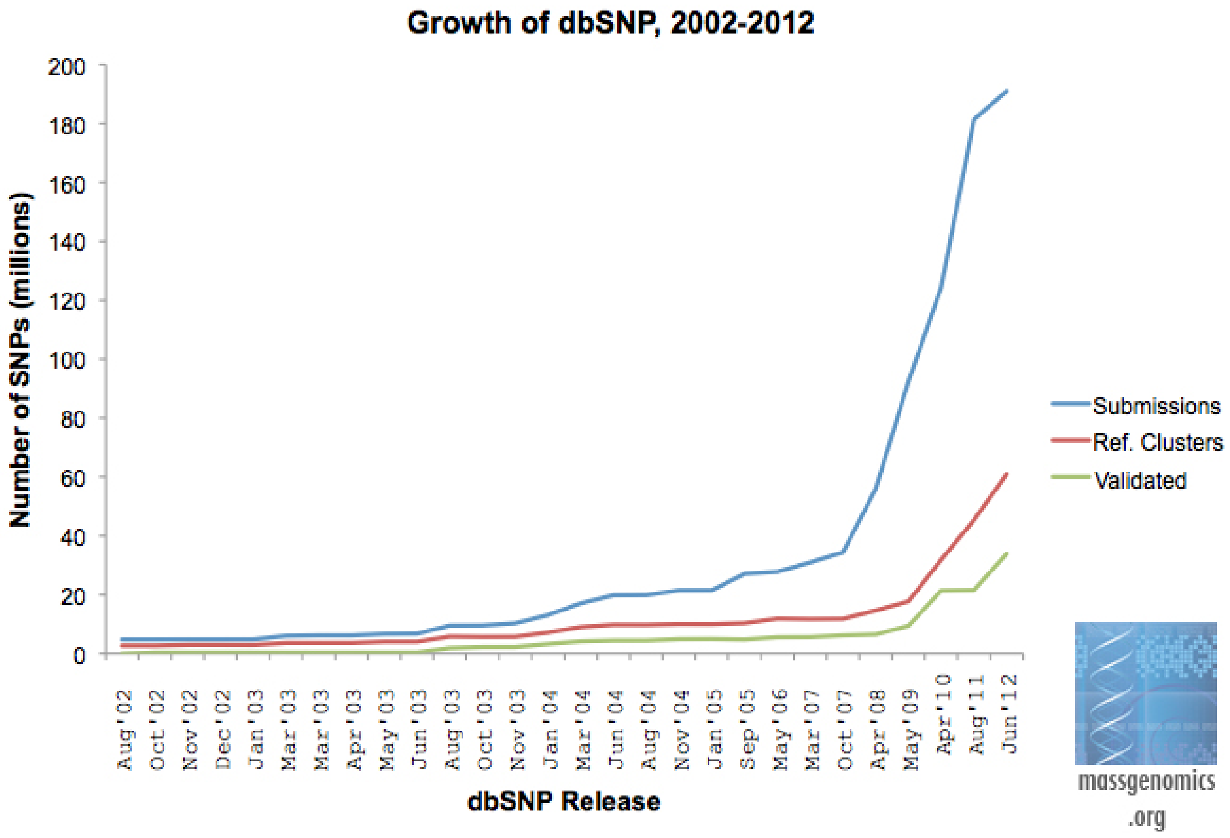
Text searching algorithms

- Exact searching (Knuth-Morris-Pratt, Boyer-Moore) : not applicable
- Many reads and one genome – we would like to index the genome to be able to process the reads quickly
- We need to take errors and variants into account, but hopefully not too many of them in a single read
- We should consider text indexes (Suffix trees, suffix arrays and Burrows-Wheeler transform)

Something about SNPs

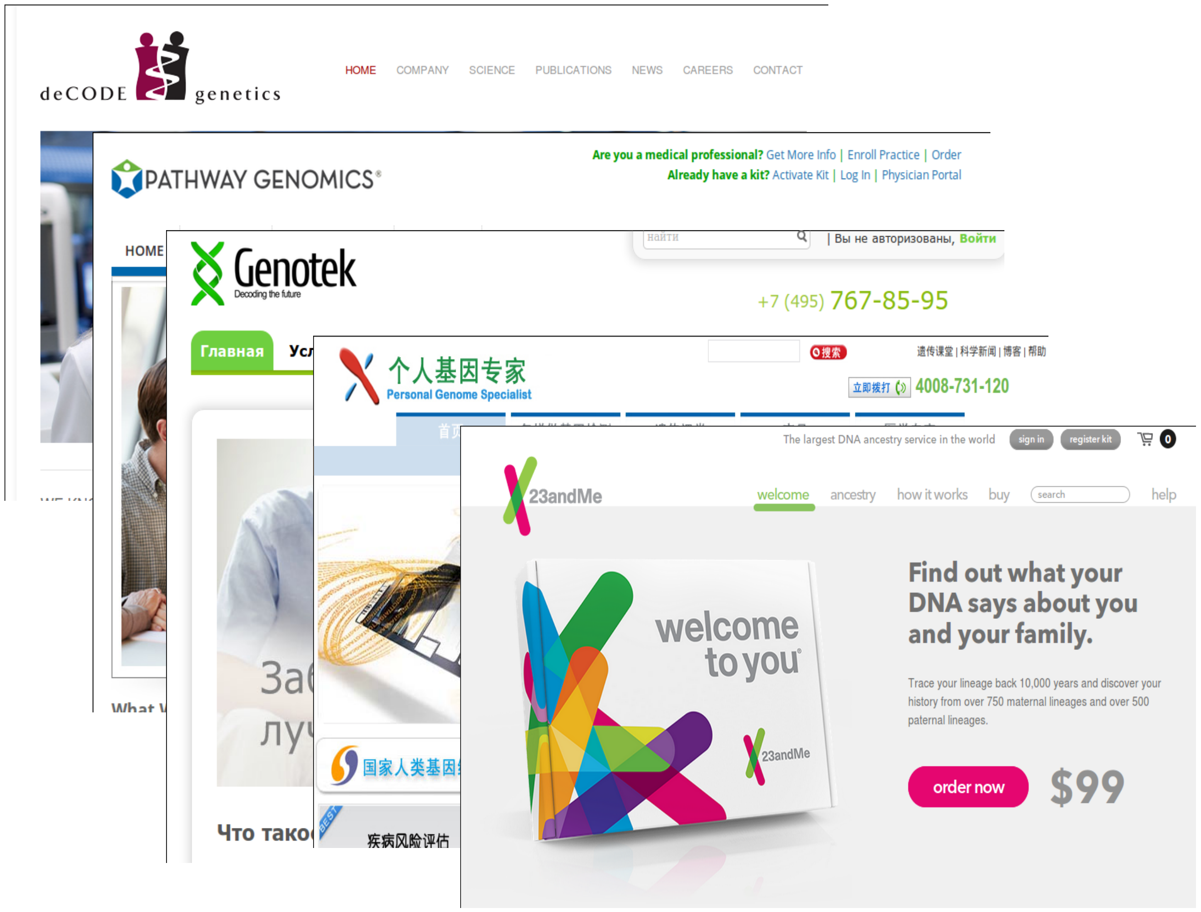
- Single nucleotide polymorhism (SNP) a position in the genome where a natural variation in population occurs





Genotyping vs. Sequencing

- Many commercial services offer genotyping (usually not sequencing) for very low prices
- Some of this information might be important if you are sick
- Most of the information provided by such companies is pure noise and correlative data
- Data security is a big issue



deCODE genetics
HOME COMPANY SCIENCE PUBLICATIONS NEWS CAREERS CONTACT
PATHWAY GENOMICS®
Are you a medical professional? Get More Info | Enroll Practice | Order
Already have a kit? Activate Kit | Log In | Physician Portal
Genotek Decoding the future
| Вы не авторизованы, Войти
+7 (495) 767-85-95
Главная
个人基因专家 Personal Genome Specialist
4008-731-120
国家人类基因
疾病风险评估
Что тако
23andMe
The largest DNA ancestry service in the world
sign in register kit
welcome ancestry how it works buy search help
welcome to you
Find out what your DNA says about you and your family.
Trace your lineage back 10,000 years and discover your history from over 750 maternal lineages and over 500 paternal lineages.
order now
$99

Suffix tree

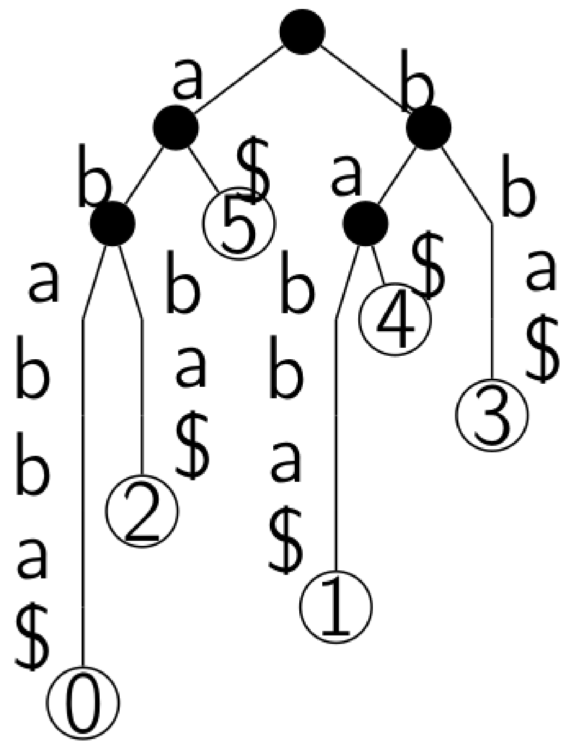
Suffix tree

- every edge is labelled with a text substring
- labels from consecutive edges on pathes from root to leafs constitute suffixes
- each suffix is represented in this way and corresponding leaf is labelled with its position in the text
- labels of sibling edges begin with different symbols

Index size: $\geq 10 \cdot |T|$ bytes

Matching time: $\mathcal{O}(|P| + |occurences|)$

Suffix tree for text ababba



Suffix array

Suffix array contains starting positions of lexicographically ordered suffixes

position	suffix	SA entry
0	ababba$	SA[0]=0
2	abba$	SA[1]=2
5	a$	SA[2]=5
1	babba$	SA[3]=1
4	ba$	SA[4]=4
3	bba$	SA[5]=3
6	$	SA[6]=6

Suffix array for text ababba

Index size: $4 \cdot |T|$ bytes

Matching time: $\mathcal{O}(|P| \cdot \log |T| + |occurences|)$

with additional LCP table

Index size: $5 \cdot |T| - 8 \cdot |T|$ bytes

Matching time: $\mathcal{O}(|P| + \log |T| + |occurences|)$

Burrows-Wheeler transform

Burrows-Wheeler transform
contains symbols predecessing lexicographically ordered suffixes.

$$\mathrm{BWT}[i] = T[\mathrm{SA}[i] - 1]$$

Burrows-Wheeler transform for text ababba

position	suffix	BWT entry
0	ababba\$	BWT[0]=\$
2	abba\$ab	BWT[1]=b
5	a\$ababb	BWT[2]=b
1	babba\$a	BWT[3]=a
4	ba\$abab	BWT[4]=b
3	bba\$aba	BWT[5]=a
6	\$ababba	BWT[6]=a

Last-to-first mapping

Cyclic shifts
of text ababba$

i	F L	SA[i]	LF[i]
0	ababba$	0	6
1	abba$ab	2	3
2	a$ababb	5	4
3	babba$a	1	0
4	ba$abab	4	5
5	bba$aba	3	1
6	$ababba	6	2

Last-to-first mapping

$LF(i)$ is the position in column F of the i-th symbol of column L.

Observation

$$SA[i] = SA[LF(i)] + 1$$

Corollary

$$SA[i] = SA[LF^k(i)] + k$$

Computing last-to-first mapping

Cyclic shifts
of word ababba$

i	F	L
0	a	ababba$
1	a	bba$ab
2	a	$ababb
3	b	abba$a
4	b	a$abab
5	b	ba$aba
6	$	ababba

Observation

Occurences of symbol x in columns F and L are ordered accordingly.

Proof

The order is determined by suffixes following occurences of x.

$C(x)$	number of occurences of symbols lexicographically smaller than x in T
$Occ(x, i)$	number of occurences of symbol x in $BWT[0 : i]$

Observation

$LF(i) = C(BWT[i]) + Occ(BWT[i], i)$

Extracting text

Structure for extracting text

- Burrows-Wheeler transform of T
- array C
- regularly sampled values of arrays $Occ(x, \_)$
- array with regularly sampled values of SA^{-1}

Algorithm

```
function EXTRACT($begin, end$)
    $p \leftarrow cache[\lceil end / CacheEvery_{text} \rceil]$        ▷ Get the closest cached position after end
    $dist \leftarrow end - end \mod CacheEvery_{text}$
    while $dist > 0$ do                                      ▷ $LF$-map to the $end$ position
        $p \leftarrow LF(p)$
    end while
    $dist \leftarrow end - begin$
    $result = \epsilon$
    while $dist > 0$ do                  ▷ $LF$-map and extract next $begin - end$ characters
        $result = BWT[p] + result$                ▷ Prepend current character to the result
        $p \leftarrow LF(p)$
    end while
    return $result$
end function
```

Backward searching

Structure for backward searching

- Burrows-Wheeler transform of T
- array C
- regularly sampled values of arrays $Occ(x, \_)$

Algorithm

```
function FIND($Q_{1..m}$)
    $sp \leftarrow C(Q_m)$
    $ep \leftarrow C(Q_m + 1) - 1$
    for $i \leftarrow m - 1, 1$ do
        $sp = C(Q_i) + Occ(Q_i, sp - 1) + 1$
        $ep = C(Q_i) + Occ(Q_i, ep)$
        if $ep > sp$ then
            break                                ▷ No matches, jump out
        end if
    end for
    return $(sp, ep)$            ▷ The opaque result is just a range in the $BWT$ array
end function
```

Suffix indexes

Suffix tree: suffixes = paths from root to leaves
- index size: $\geq 10 \cdot |genome|$ bytes
- exact mapping time: $\mathcal{O}(|read| + |occurences|)$

Suffix array: lexicographic order on suffixes
- index size: $\geq 4 \cdot |genome|$ bytes
- exact mapping time: $\mathcal{O}(|read| \cdot \log |genome| + |occurences|)$

FM index: self-index based on Burrows-Wheeler transform
- index size: $< 1 \cdot |genome|$ bytes (including sequence!)
- exact mapping time: 2-1000$\times$ slower than suffix arrays

Operations in Ferragina-Manzini index

Find(Q) $\rightarrow R$ searches for all occurrences of sequence Q and returns an opaque result R that can be used with other operations.

FindSuffixes($Q_{1..m}$) $\rightarrow R_{1..m}$ works just like Find, but returns results for each suffix of Q so that R_i is the result of searching for $Q_{i..m}$.

FindContinue($Q_{1..m}, R_{old}, f$) $\rightarrow R_{new}$ just like Find searches for all occurrences of $Q_{1..m}$, but takes advantage of an earlier result R_{old}, assumed to be obtained by searching for $Q_{f..m}$, and returns a new result R_{new}.

Count(R) $\rightarrow k$ returns the number of occurrences k represented by R.

Locate(R) $\rightarrow l_{1..k}$ returns locations of occurrences represented by R.

Extract(b, l) $\rightarrow S$ retrieves a subsequence of the reference sequence T: $S = T[b..b + l - 1]$.

Bowtie (Langmead et al. '09)

Seed – high-quality part of the read (default: first 28bp)

Policy

Search for read occurences in the genome with

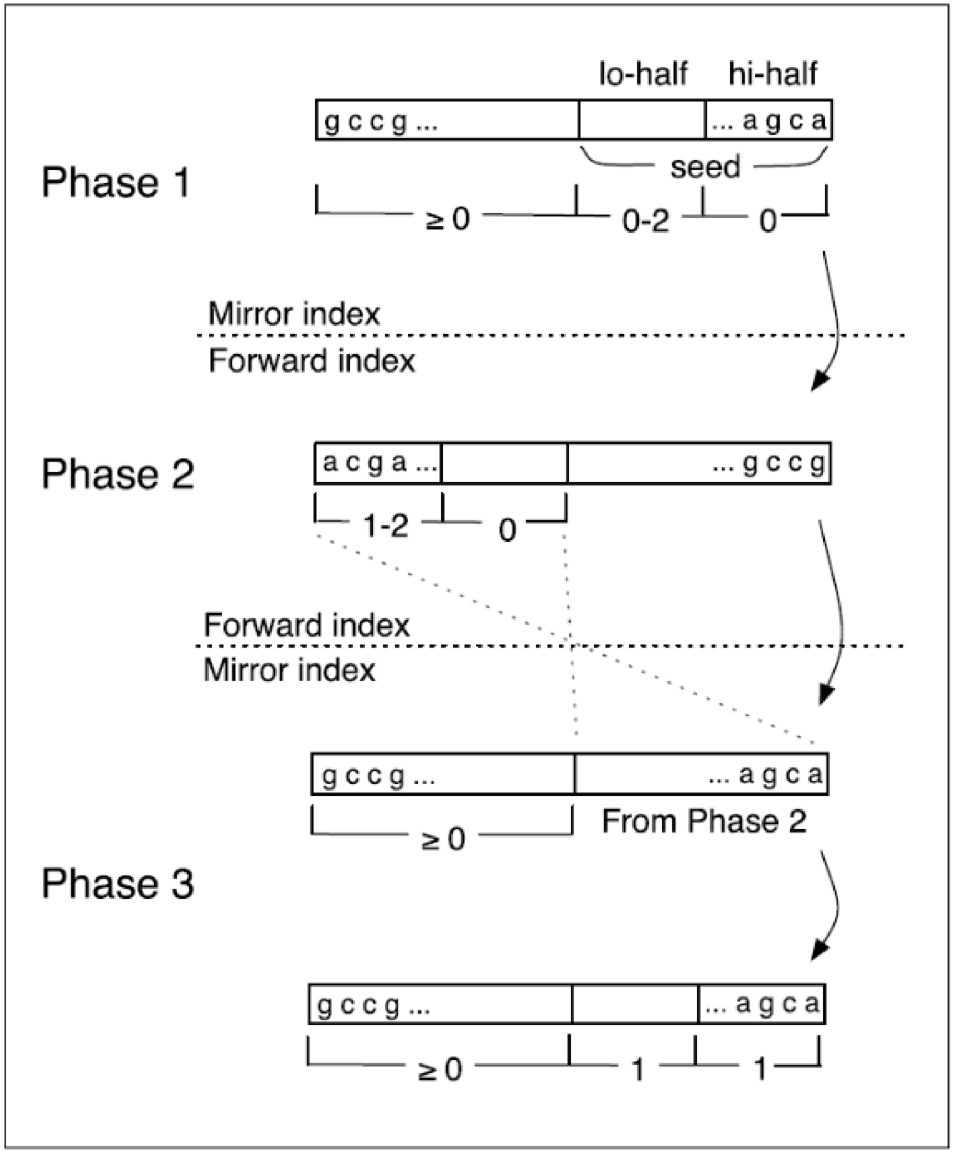
- limited number of errors in the seed (default: first 28bp),
- limited sum of quality values of mismatched positions in the whole read.

Algorithm

- Genome index is searched with k-neighborhood of the seed of a read.
- Located occurences are extended to whole read mappings and the quality criterion is checked.

Bowtie – avoiding excessive backtracking

- $k \leq 3$
- Double indexing:
  FM-index is build for a genome sequence (*forward* index) and for a reverse sequence (*mirror* index).



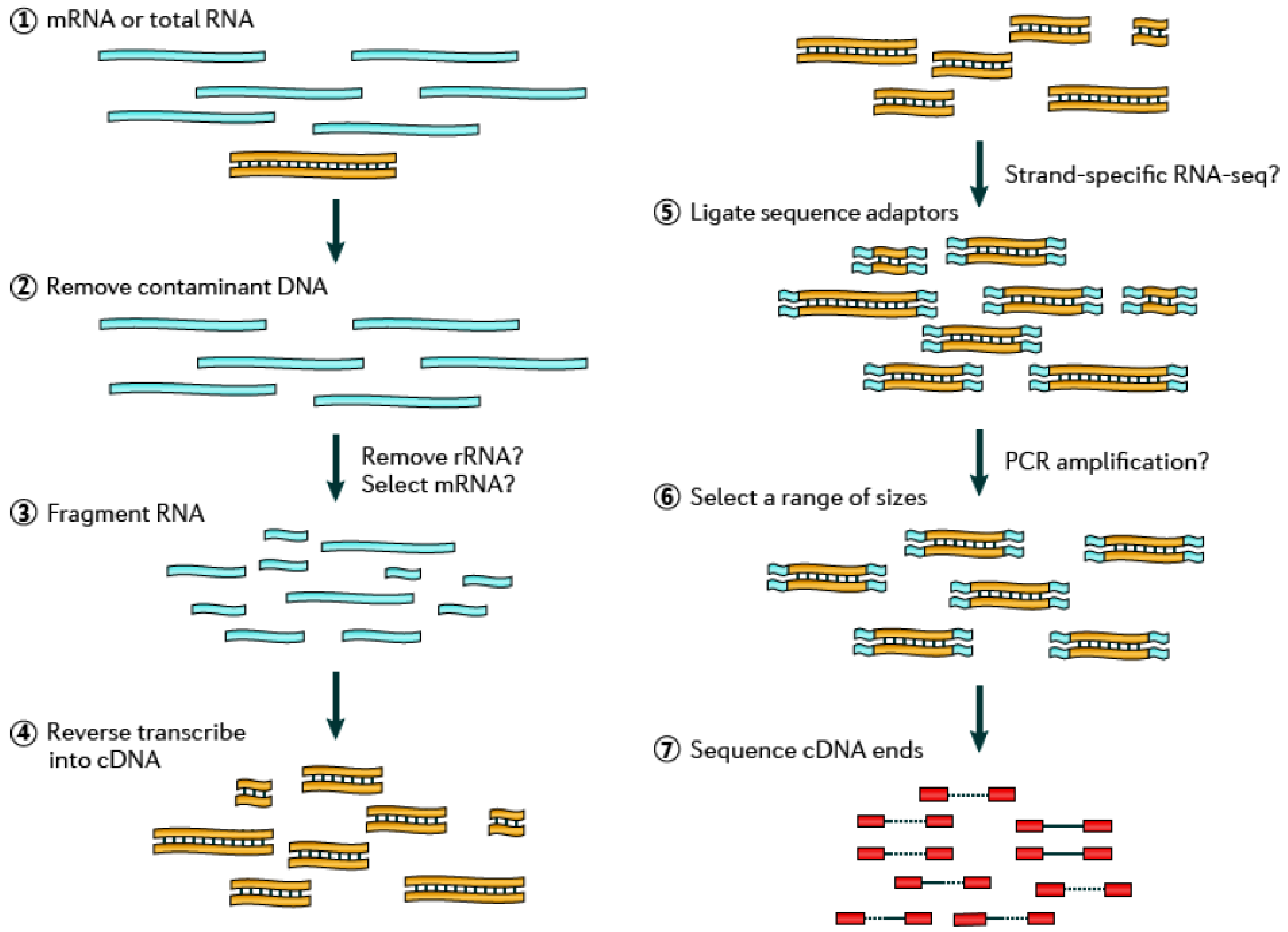
BWT mapping summary

- Effective tools are used in short read mapping using BWT and FMI
- Index can be linear in genome size and match finding with small (<3) number of mismatches is feasible
- Large number of mismatches works against these methods

Even faster read mapping?

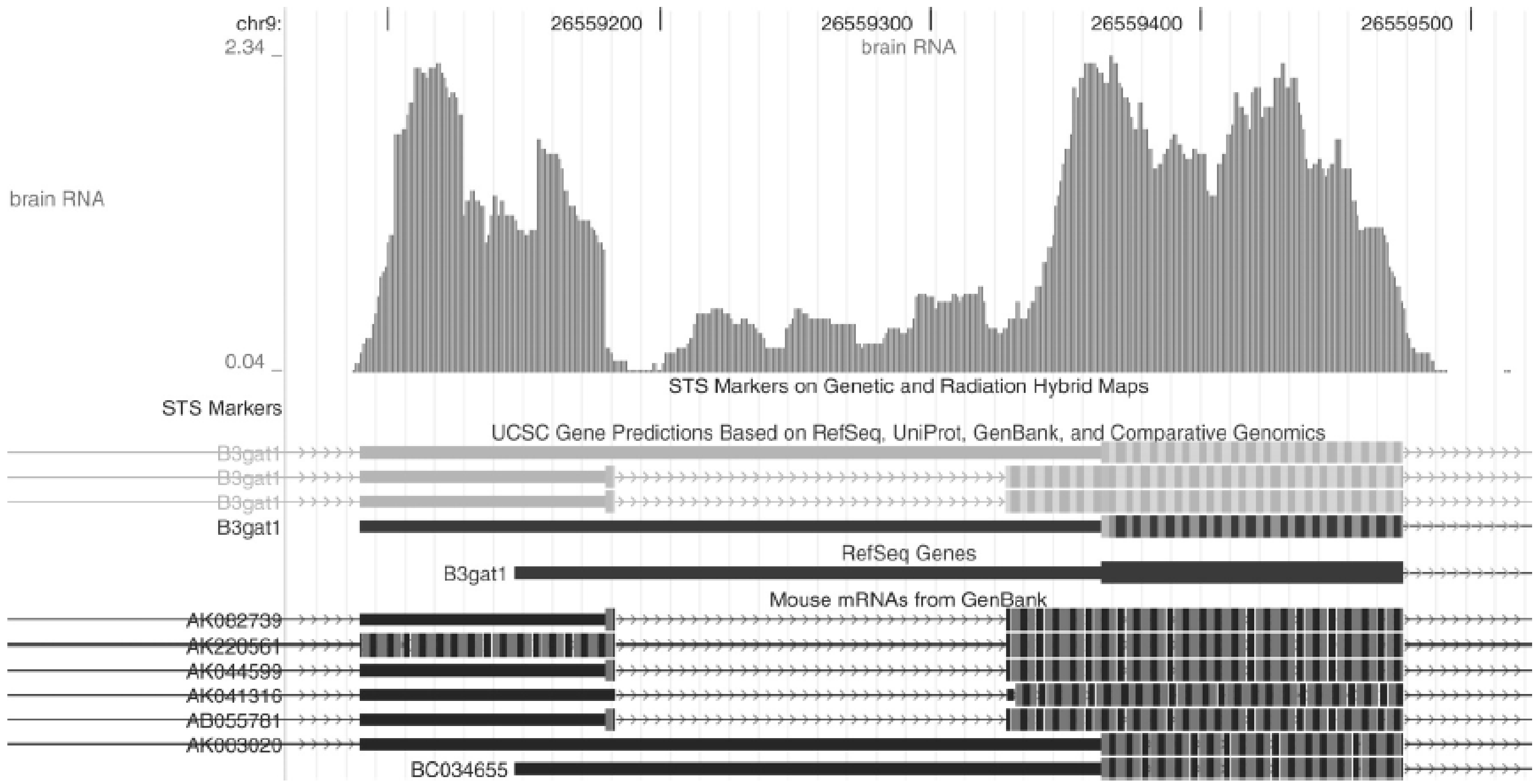
- Sometimes we can agree to a worse mapping efficiency (some random reads not mapped) if it increases the speed of overall mapping
- This is in particular true in cases where we want to count reads rather than identify the variants
- One such case is mRNA expression profiling, when we are interested in relative abundances of fragments of the reference sequence

RNA-seq data preparation

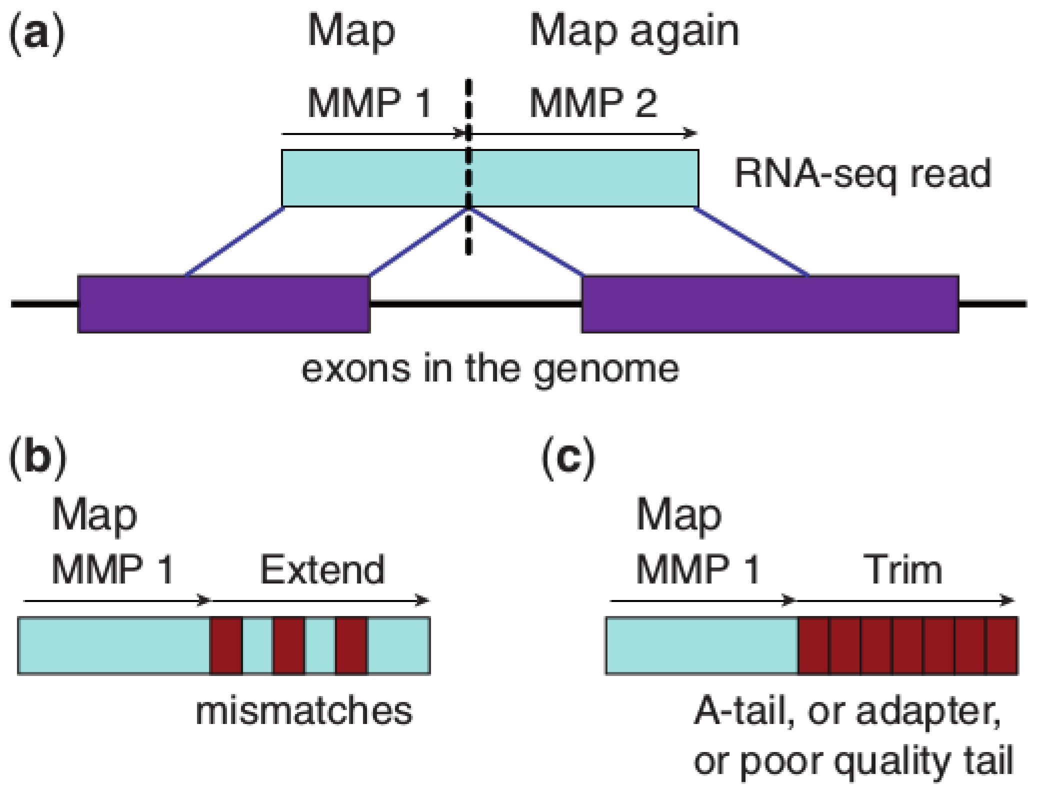


J. A. Martin and Z. Wang *Next-generation transcriptome assembly*. Nature Reviews 2011.

RNAseq Reads mapped to the genome



STAR – ultrafast read mapping (Dobin et al. 2012)



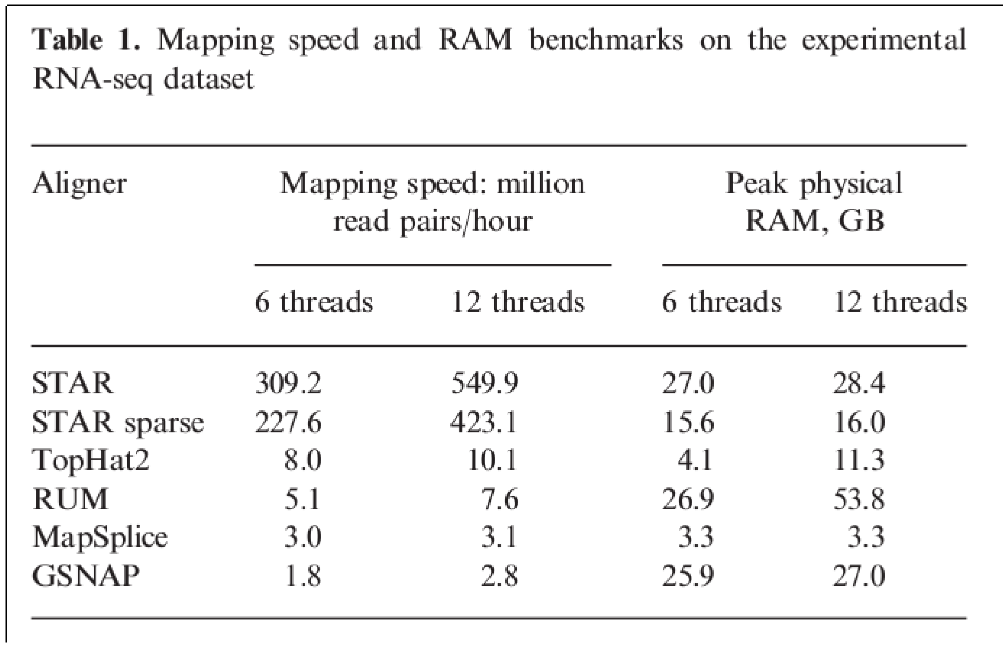
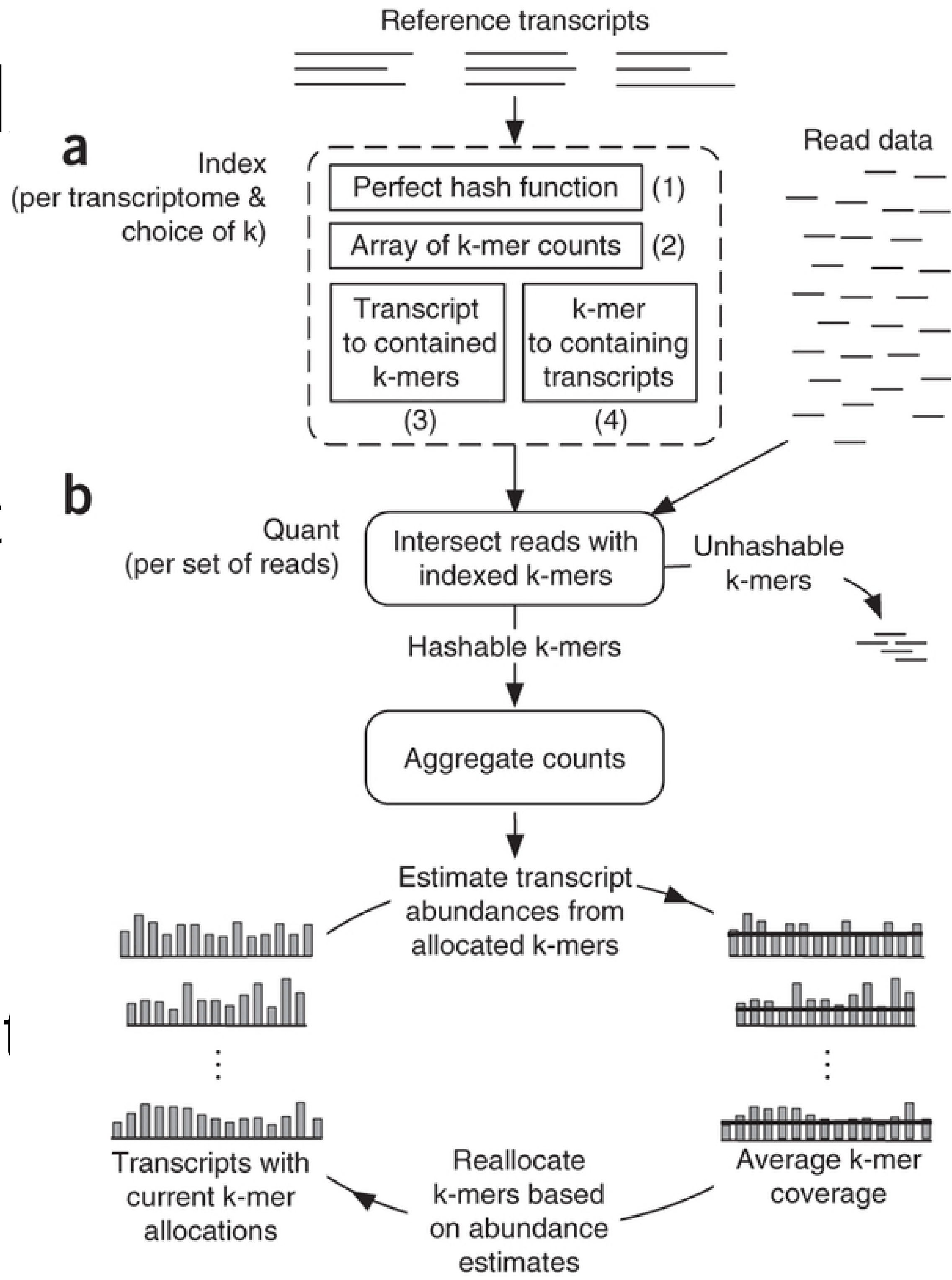


Table 1. Mapping speed and RAM benchmarks on the experimental RNA-seq dataset

Aligner	Mapping speed: million read pairs/hour		Peak physical RAM, GB	
	6 threads	12 threads	6 threads	12 threads
STAR	309.2	549.9	27.0	28.4
STAR sparse	227.6	423.1	15.6	16.0
TopHat2	8.0	10.1	4.1	11.3
RUM	5.1	7.6	26.9	53.8
MapSplice	3.0	3.1	3.3	3.3
GSNAP	1.8	2.8	25.9	27.0

Alignment free RN quantitation

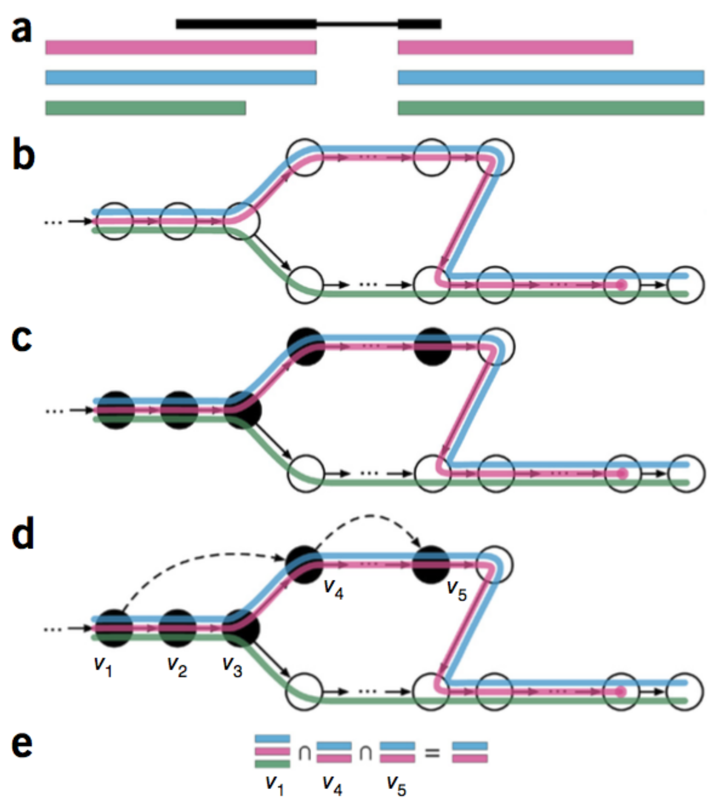
- Sailfish method (Patro et al. 2014)
- We can simply count unique k-mers in the reads and use only those to quantify transcripts
- 25x speed improvement, withou much loss in accuracy



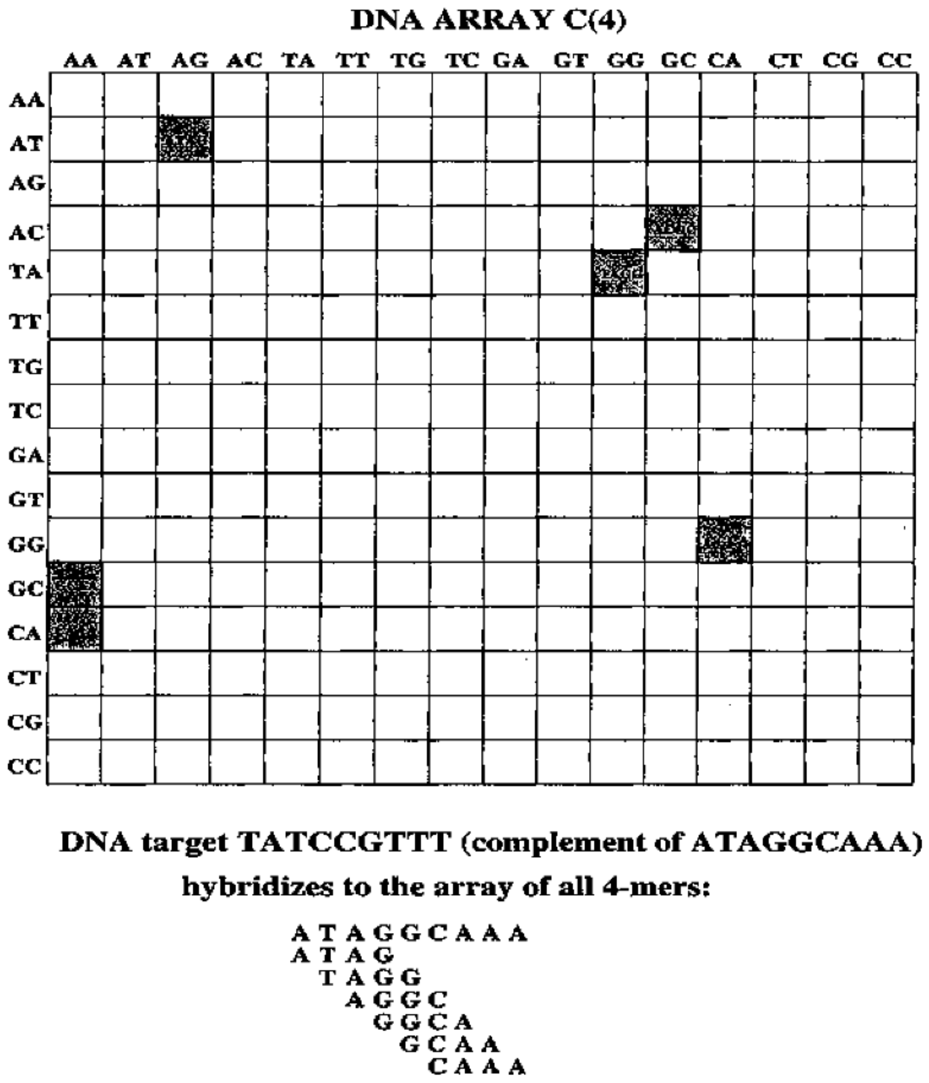
Reference transcripts
a
Index
(per transcriptome & choice of k)
Perfect hash function (1)
Array of k-mer counts (2)
Transcript to contained k-mers (3)
k-mer to containing transcripts (4)
Read data
b
Quant
(per set of reads)
Intersect reads with indexed k-mers
Unhashable k-mers
Hashable k-mers
Aggregate counts
Estimate transcript abundances from allocated k-mers
Average k-mer coverage
Reallocate k-mers based on abundance estimates
Transcripts with current k-mer allocations

Kallisto -even faster quatitation

- Kallisto method (Bray et al. 2015)
- Introducing a graph of overlapping k-mers for the different transcripts as an index
- Better implementation gives another 10x speed improvement



Sequencing by Hybridization



DNA ARRAY C(4)

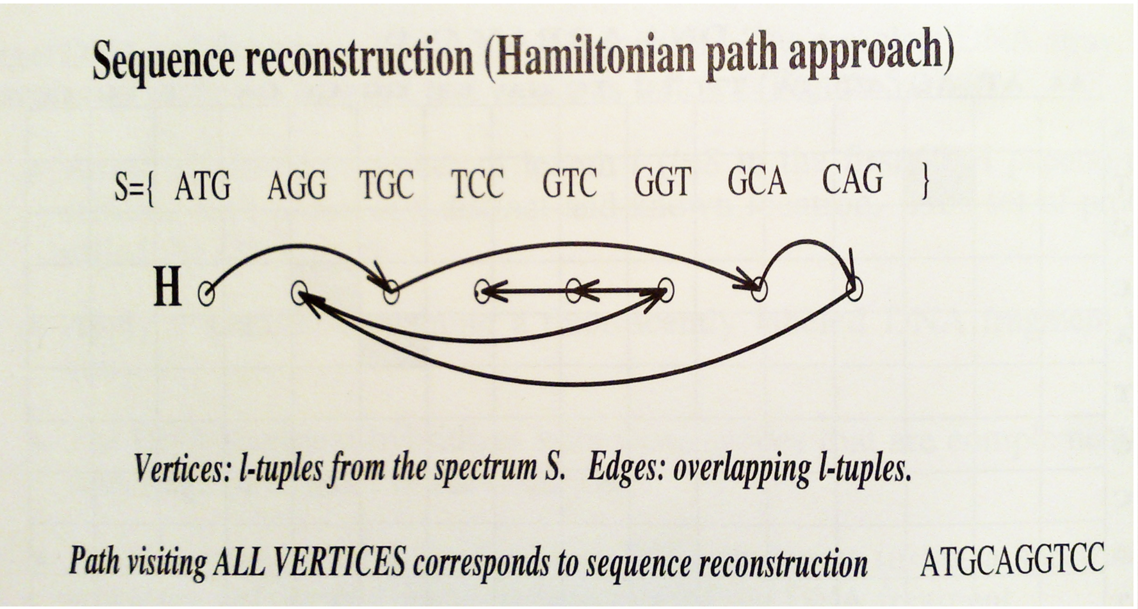
	AA	AT	AG	AC	TA	TT	TG	TC	GA	GT	GG	GC	CA	CT	CG	CC
AA																
AT			▨													
AG																
AC												▨				
TA											▨					
TT																
TG																
TC																
GA																
GT																
GG													▨			
GC	▨															
CA	▨															
CT																
CG																
CC																

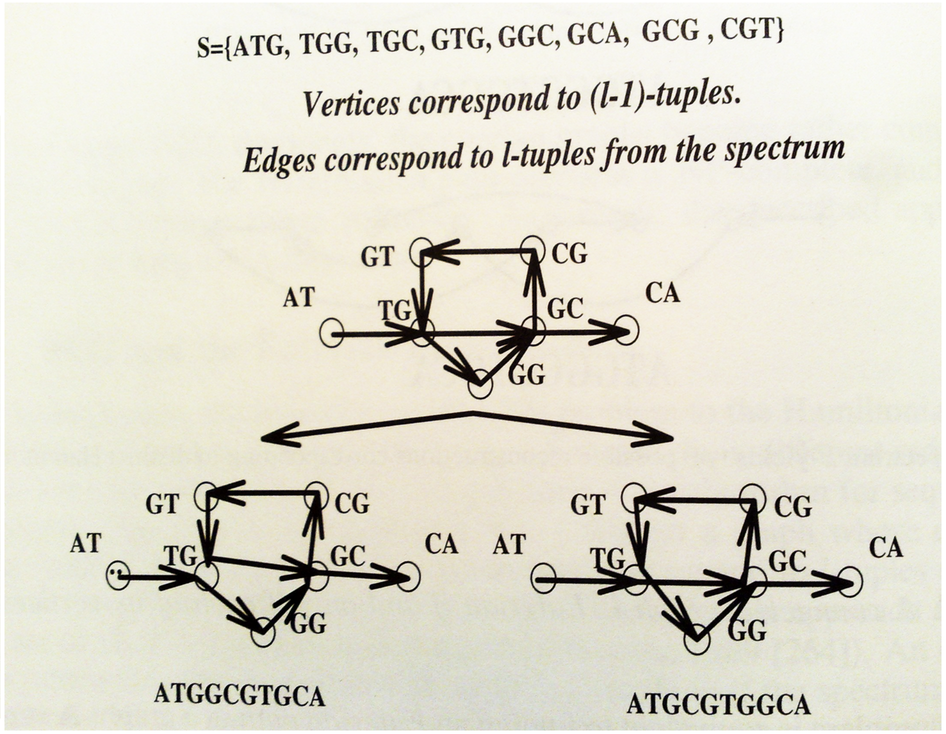
DNA target TATCCGTTT (complement of ATAGGCAAA) hybridizes to the array of all 4-mers:

```
ATAGGCAAA
ATAG
 TAGG
  AGGC
   GGCA
    GCAA
     CAAA
```

Sequence reconstruction

- Given the spectrum of observed k-mers, we can reconstruct the sequence
- Direct approach leads to the Hamiltionian path problem (NP-Complete)
- Small change in the k-mer representation leads to Eulerian path finding (Pevzner 2000)


Sequence reconstruction (Hamiltonian path approach)
S={ ATG AGG TGC TCC GTC GGT GCA CAG }
H
Vertices: l-tuples from the spectrum S. Edges: overlapping l-tuples.
Path visiting ALL VERTICES corresponds to sequence reconstruction
ATGCAGGTCC


S={ATG, TGG, TGC, GTG, GGC, GCA, GCG, CGT}
Vertices correspond to (l-1)-tuples.
Edges correspond to l-tuples from the spectrum
GT
CG
AT
TG
GC
CA
GG
ATGGCGTGCA
ATGCGTGGCA

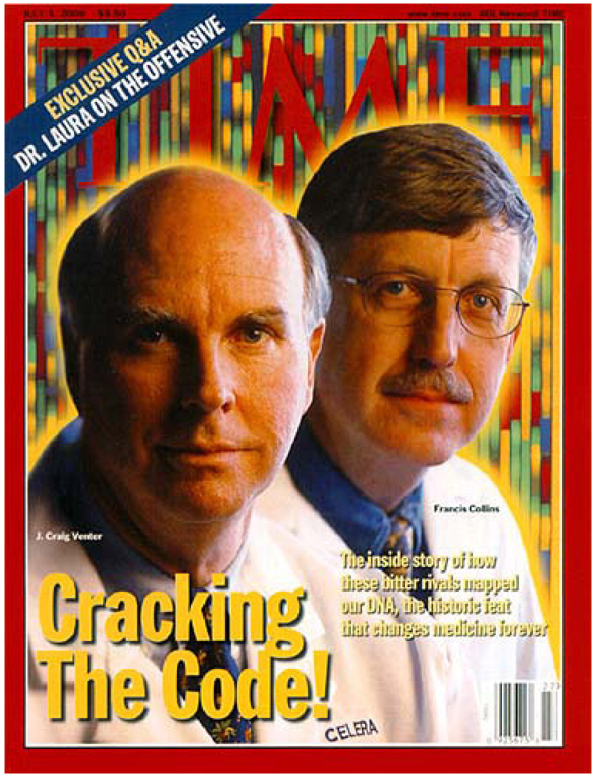
A historical digression on DNA sequence assembly

- Human Genome project
  - Started in 1984, funding since 1990, finished in 2003
  - ~$3 billion
  - Results announced in 2000 by the US president Clinton and UK prime minister Blair

- Celera genomics project
  - Started later in 1996
  - Budget ~$300 million
  - Aimed to commercialize genomic information
  - Results announced jointly with HGP

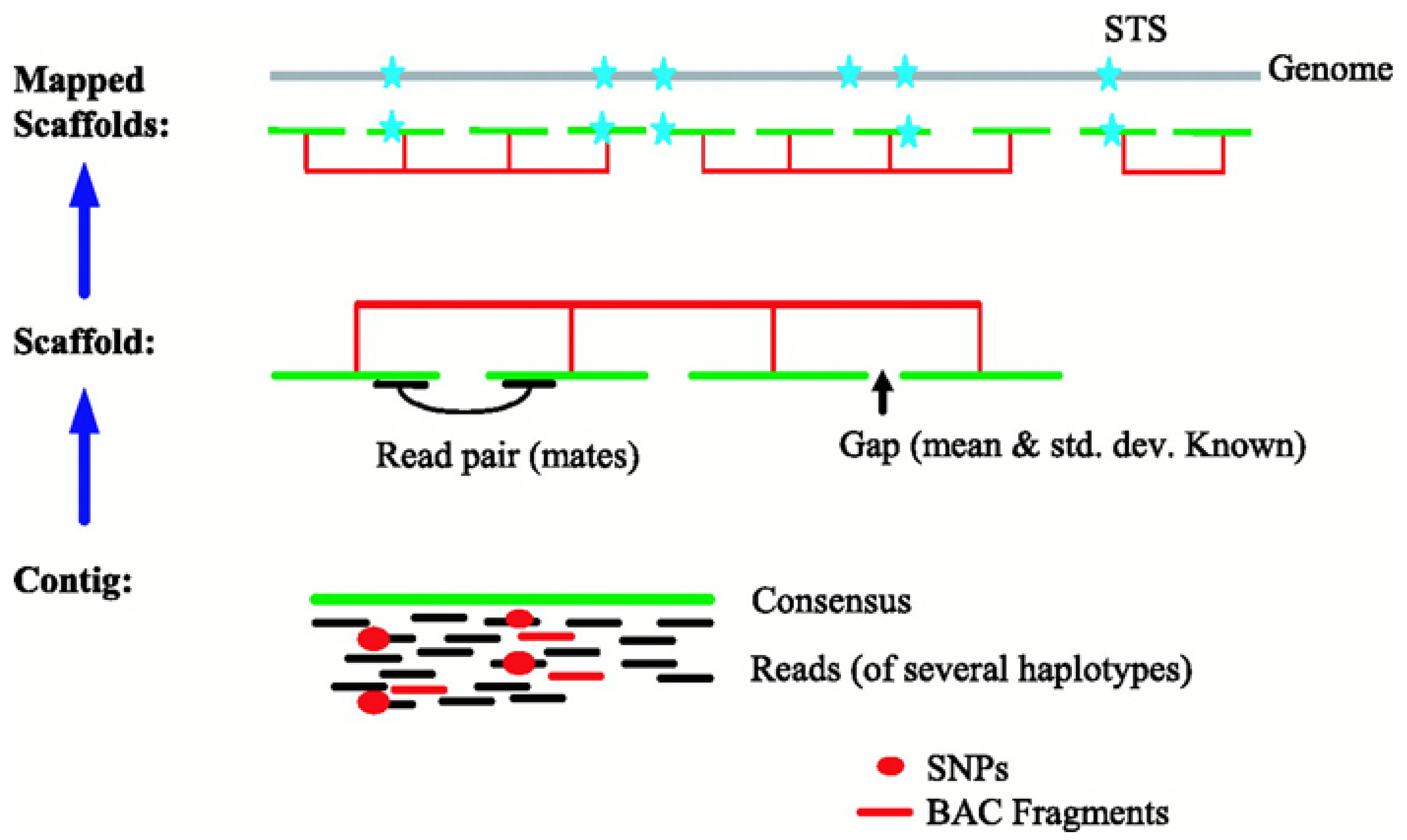
HGP announcement

- First draft announced jointly by two competing consortia
- Brought fame to Craig Venter and Francis Collins, but prevented genome commercialization

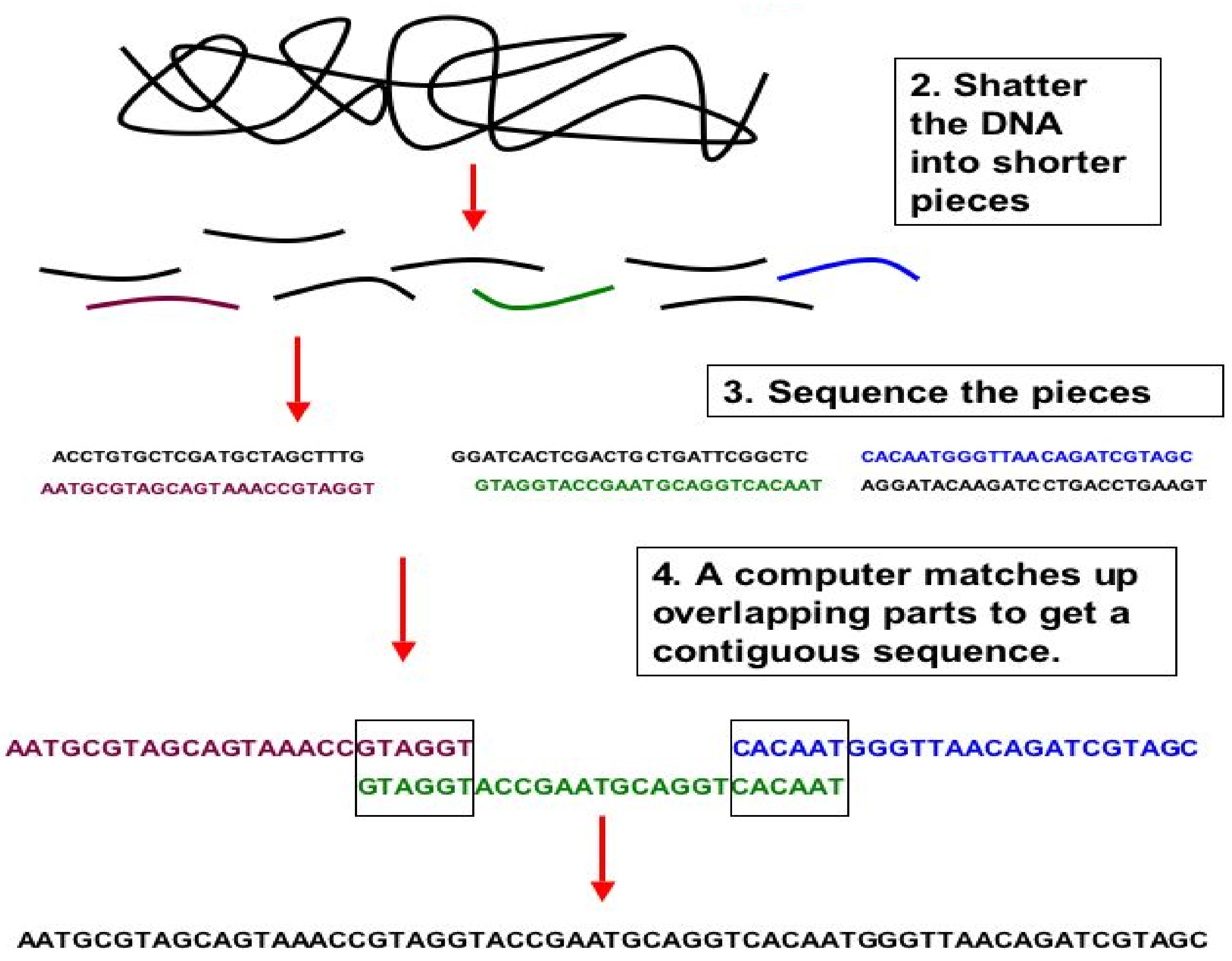

TIME
EXCLUSIVE Q&A
DR. LAURA ON THE OFFENSIVE
Francis Collins
J. Craig Venter
The inside story of how these bitter rivals mapped our DNA, the historic feat that changes medicine forever
Cracking The Code!
CELERA

Classical genome assembly (HGP)

- Oredrly process with restriction mapped fragments and scaffold assembly



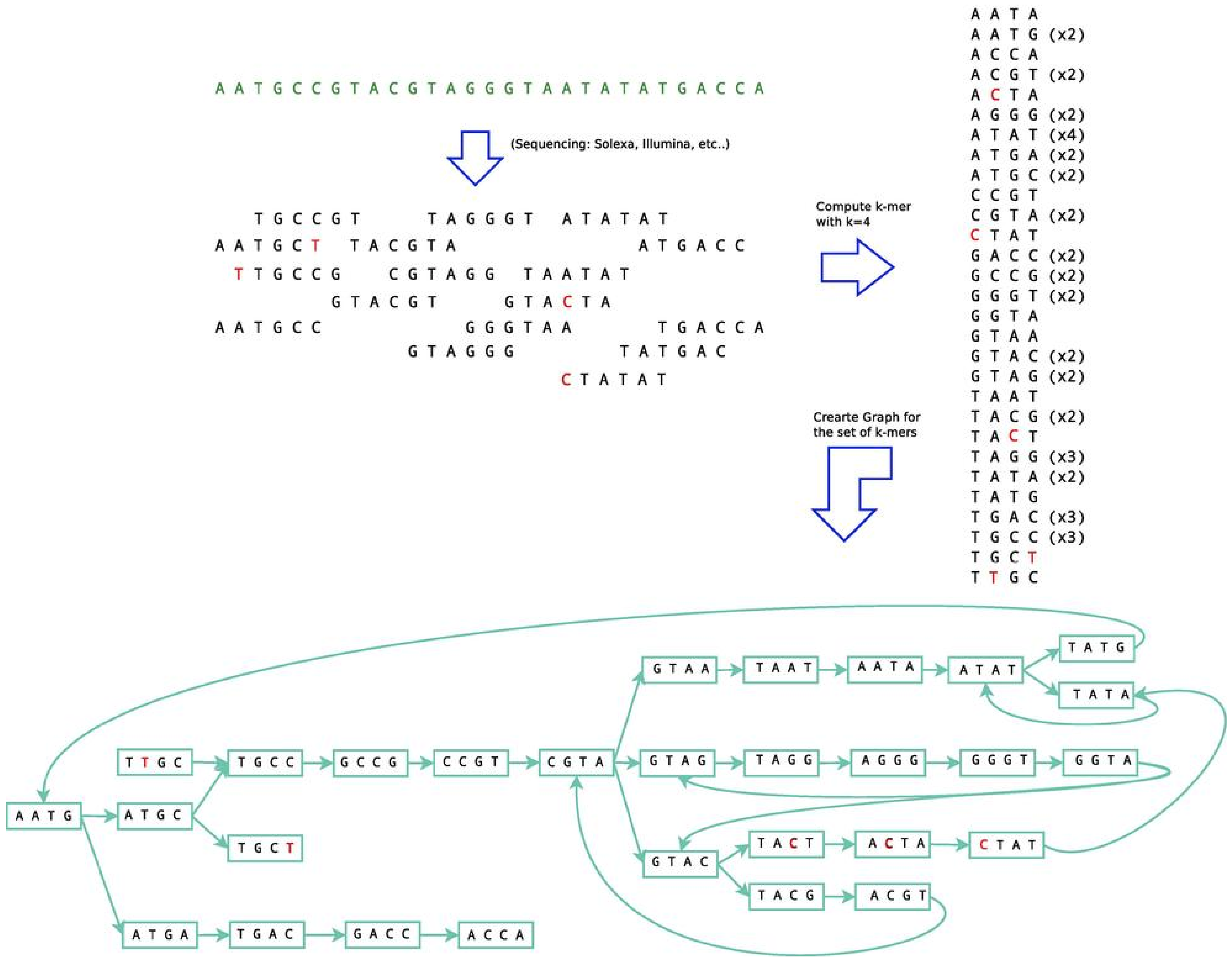
Shotgun genome sequencing (Celera, E. Myers)


2. Shatter the DNA into shorter pieces
3. Sequence the pieces
ACCTGTGCTCGATGCTAGCTTTG
AATGCGTAGCAGTAAACCGTAGGT
GGATCACTCGACTGCTGATTCGGCTC
GTAGGTACCGAATGCAGGTCACAAT
CACAATGGGTTAACAGATCGTAGC
AGGATACAAGATCCTGACCTGAAGT
4. A computer matches up overlapping parts to get a contiguous sequence.
AATGCGTAGCAGTAAACCGTAGGT
GTAGGTACCGAATGCAGGTCACAAT
CACAATGGGTTAACAGATCGTAGC
AATGCGTAGCAGTAAACCGTAGGTACCGAATGCAGGTCACAATGGGTTAACAGATCGTAGC

Take-home message from HGP

- Celera started later and could take advantage of much more computing power, therefore did not waste so much time on planning different stages of the process
- In this case the Moore’s law and smart computer scientists (E. Myers in particular) helped in speeding up the process

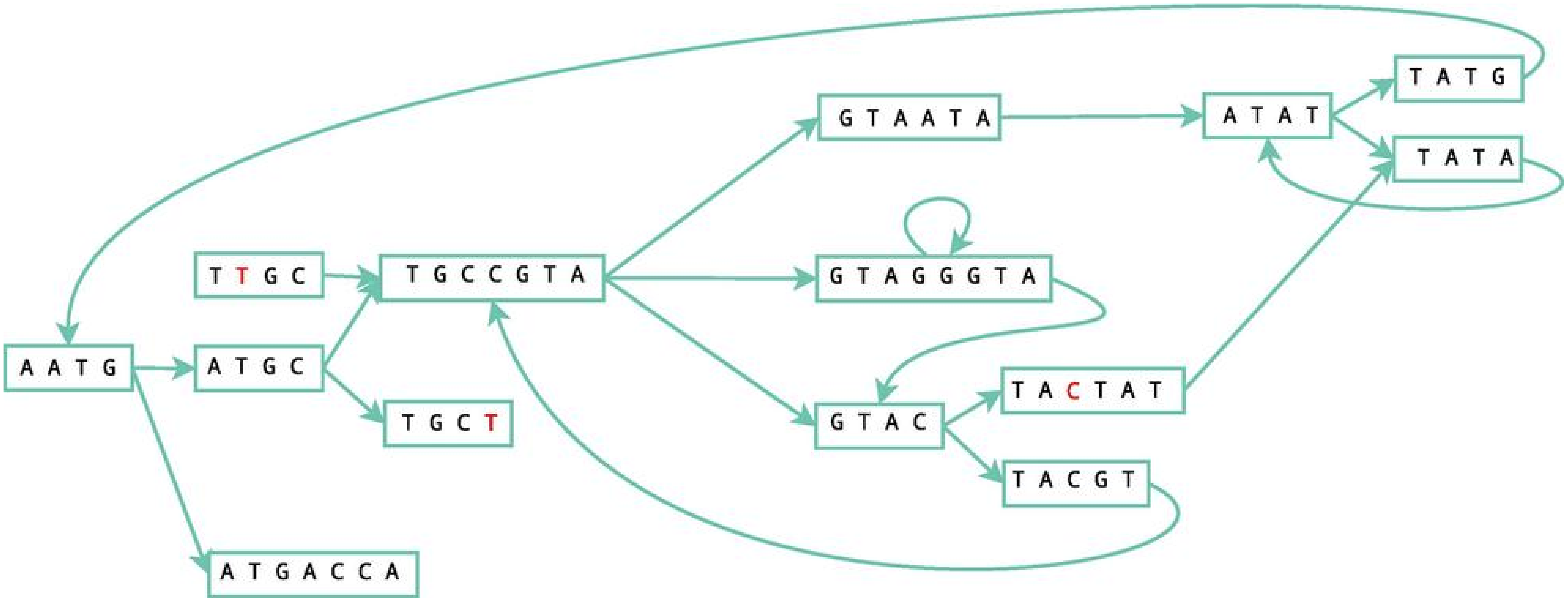
Sequence asembly from short reads



VELVET assembler, Zerbino et al. 2008

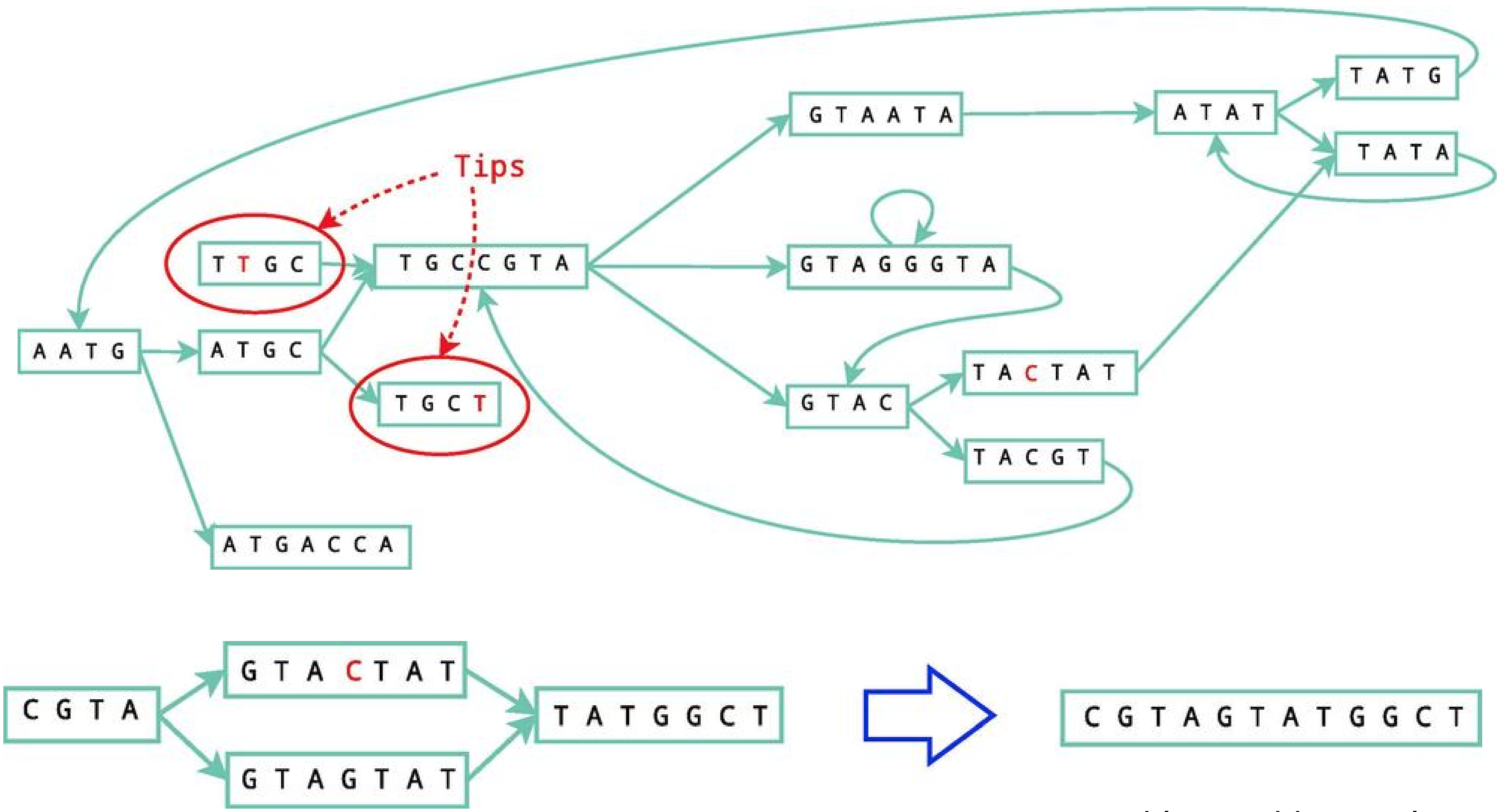
Simplification of deBruijn graph

- We can compress paths without forks



VELVET assembler, Zerbino et al. 2008

Tips and bubble removal



VELVET assembler, Zerbino et al. 2008

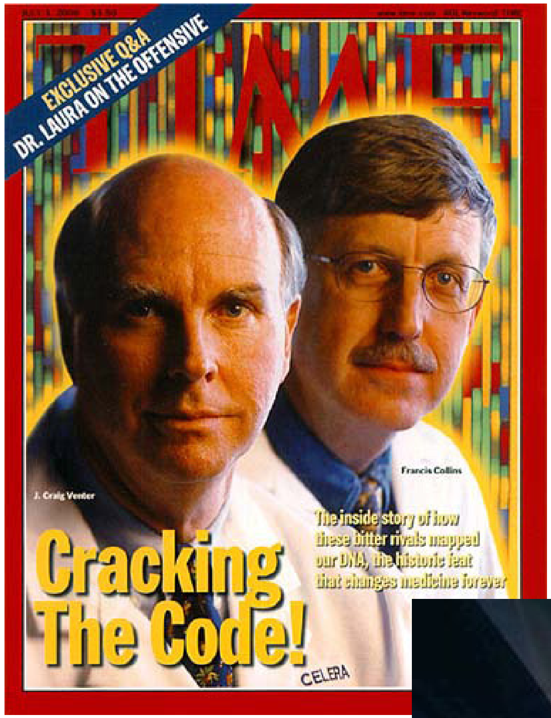
De novo assembly

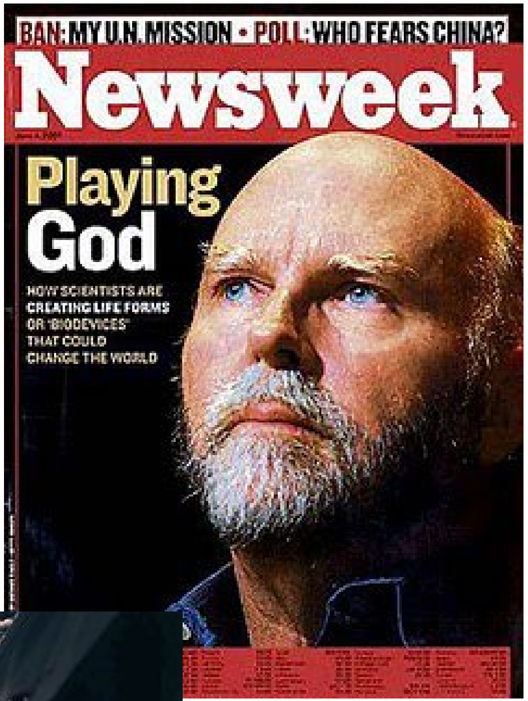
- De novo assemblers (VELVET, Spades, etc.) are ressurecting the idea behind Sequencing by hybridization
- Even though there are limitations to their use (repetitive regions, k-mer length, memory constraints) they are very useful in contig creation from raw reads
- Many heuristic improvements and specialized tools for specific applications

Metagenomics

- Popularized by Craig Venter in Global Ocean Sampling expedition
- Shotgun sequencing of microbes from Sargasso sea
- Identified many novel gene sequences without attributing them to specific species
- Now very frequently done in other environments: soil, human skin, human intestine
- Helpful in finding new important enzymes (from soil around chemical waste facilities)
- Identified some microbes that are relevant for human health

Dr Venter and his projects


TIME
EXCLUSIVE Q&A
DR. LAURA ON THE OFFENSIVE
Francis Collins
J. Craig Venter
The inside story of how these bitter rivals mapped our DNA, the historic feat that changes medicine forever
Cracking The Code!
CELERA


BAN: MY U.N. MISSION • POLL: WHO FEARS CHINA?
Newsweek
Playing God
HOW SCIENTISTS ARE CREATING LIFE FORMS OR 'BIODEVICES' THAT COULD CHANGE THE WORLD

Bruno Lemaitre

An Essay on Science and Narcissism

How do high-ego personalities drive research in life sciences?