



Chinese Academy of Sciences

ZHANGGroup

生物信息学

新一代测序技术

吴凌云

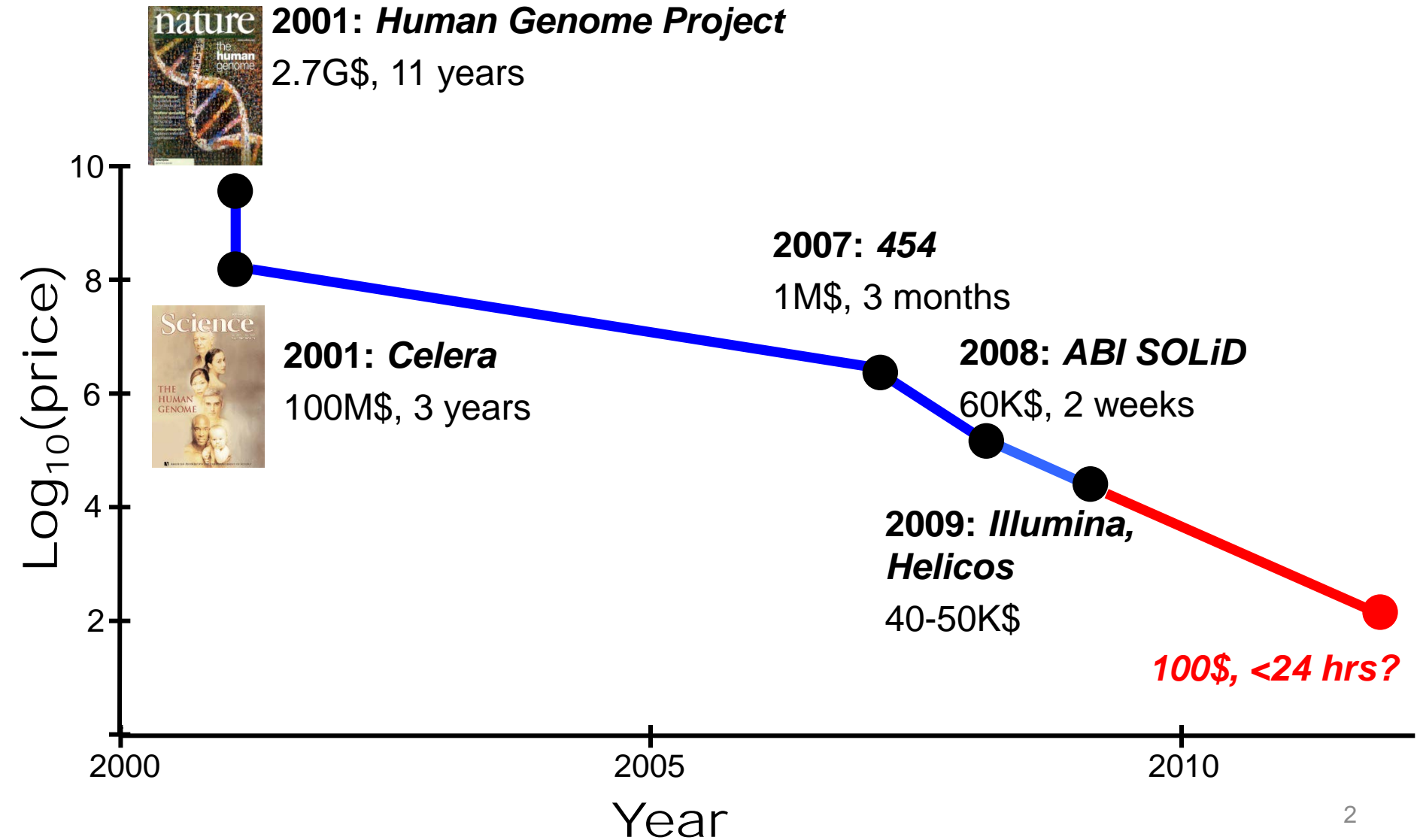
中国科学院数学与系统科学研究院



<http://zhanggroup.aporc.org>
Chinese Academy of Sciences



Sequencing the Human Genome





目录

- **电泳测序**
- 杂交测序技术
- 新一代测序技术



电泳 (Electrophoresis)

- 变性聚丙烯酰胺凝胶（测序胶）
- 在凝胶一端小槽中放入荧光标记的DNA片断，两端加电压，短DNA片断跑得快，长DNA片断跑得慢。
- 测序时需要区分长度只差一个碱基的片断。



PCR (聚合酶链式反应)

- Polymerase Chain Reaction
- DNA体外扩增方法的一种，能够将很少的试样（比如只有罪犯的一滴血），扩增成完全相同的无数拷贝。
- 每PCR一轮，扩增两倍 1-2-4-8-16...

引物 (Primer) 退火反应

Annealing Reaction

Template

5' **ATTAGACGTCCG** **TGCAATGC** 3'

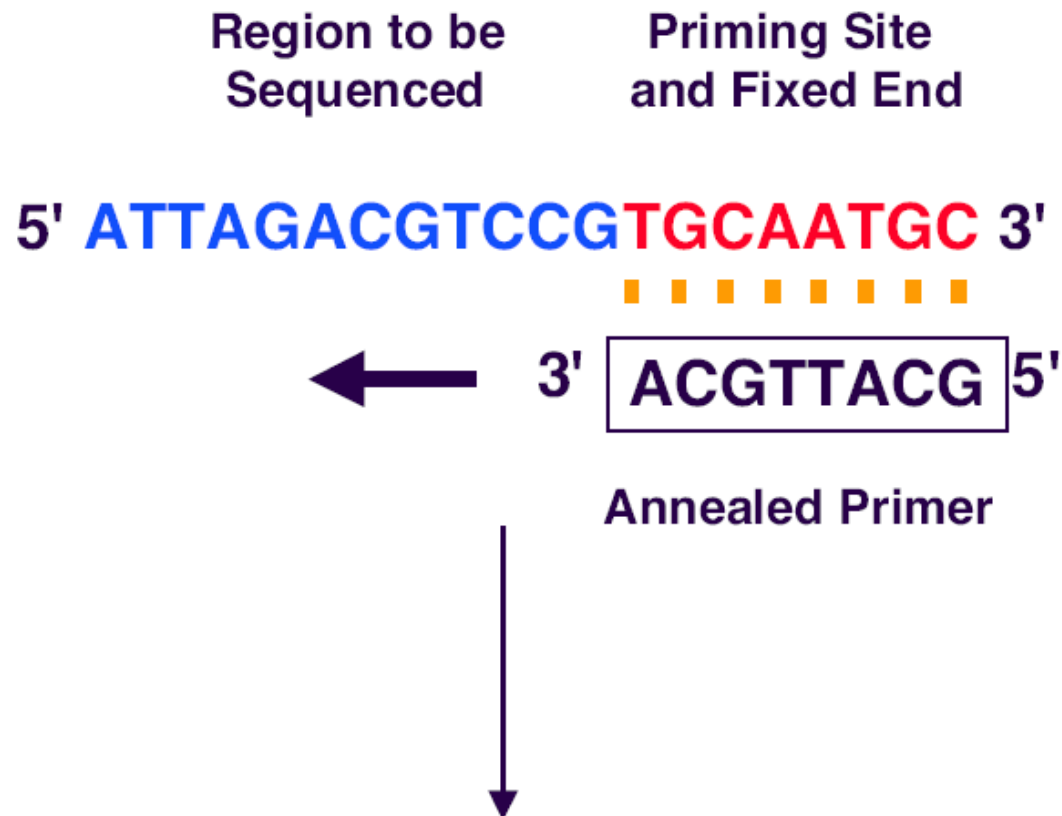


3' **ACGTTACG** 5'

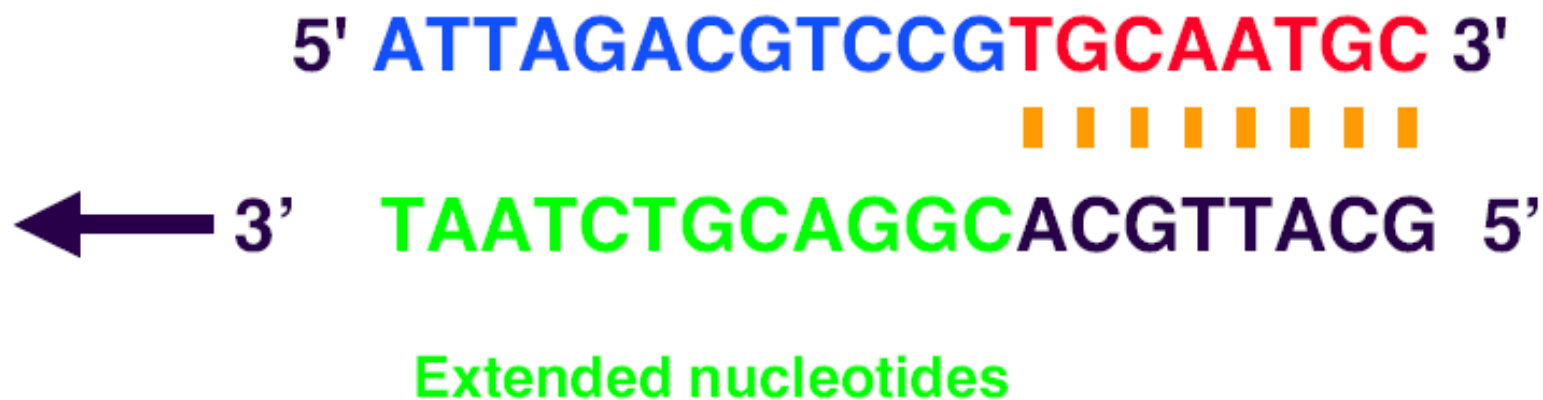
Primer

延长反应

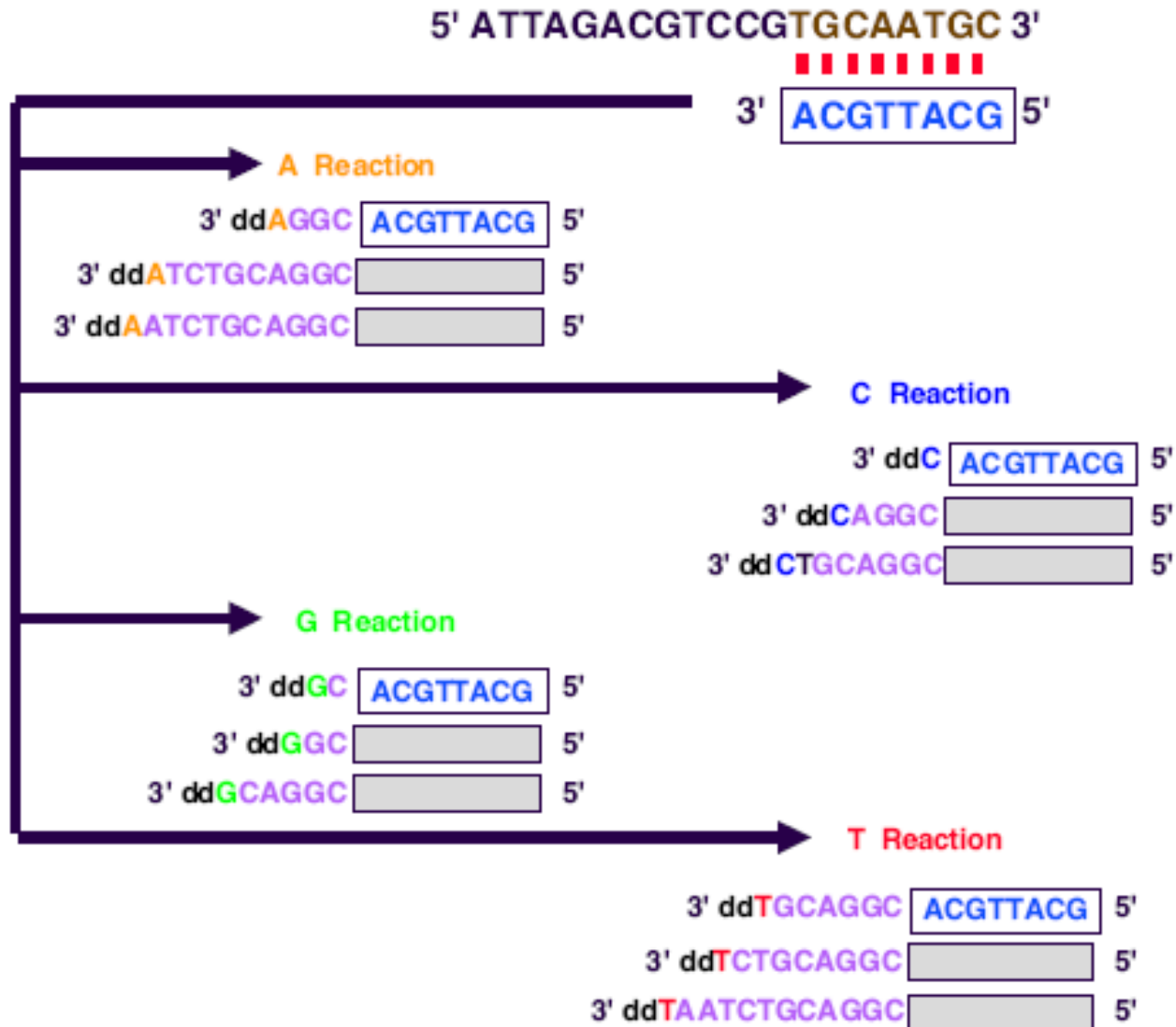
Elongation Reaction



延长反应

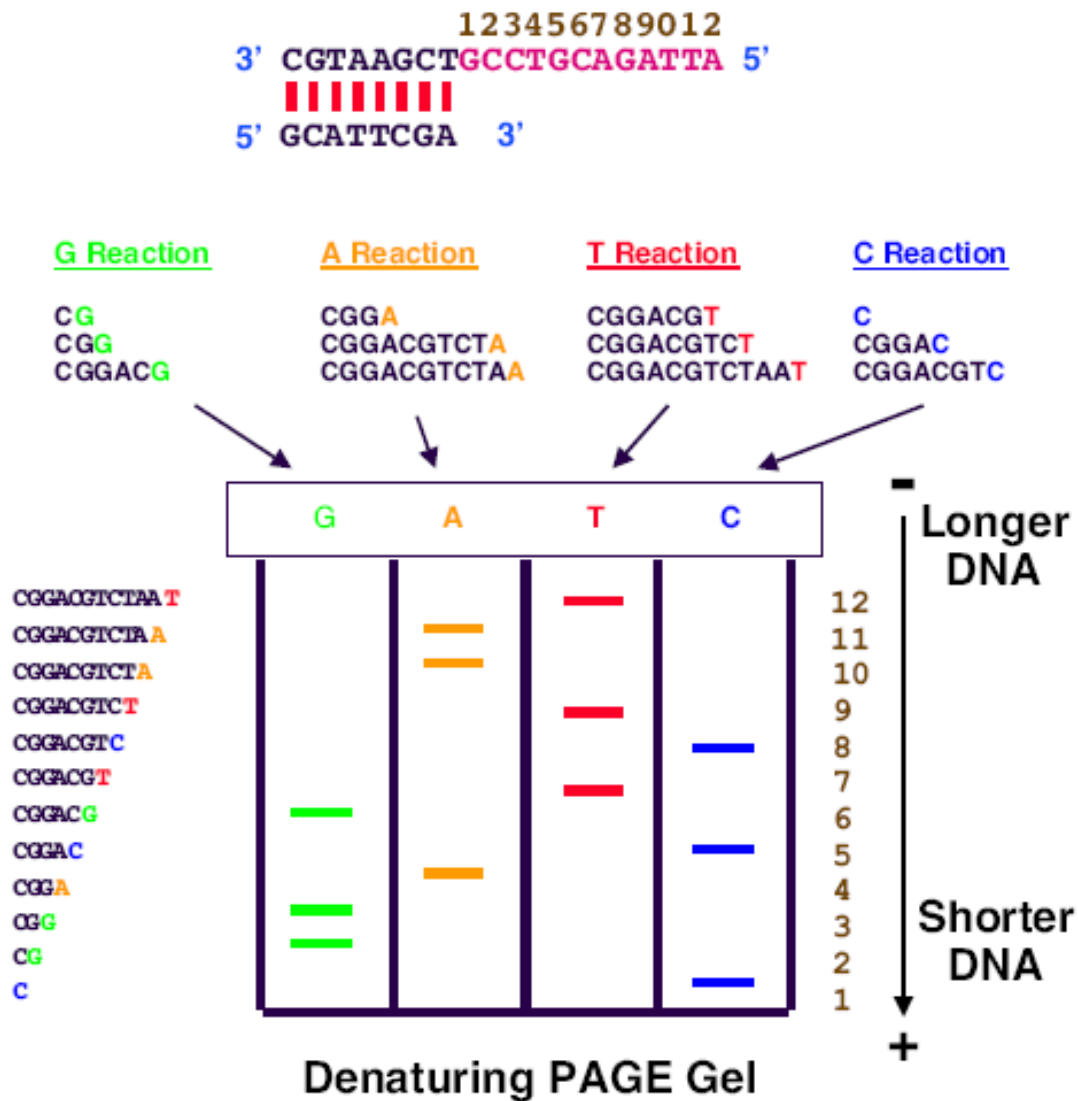


终止反应





电泳测序法





Sanger法

- 在PCR时加入荧光标记的复制终止剂（双脱氧核糖核苷酸），比如ddA,ddT,ddC,ddG（相应于4种碱基）和普通的脱氧核糖核苷酸
- ddX的两个作用：
 - 可以当作正常碱基参与复制
 - 一旦链入DNA中，其后就不能再继续连接






Sanger法

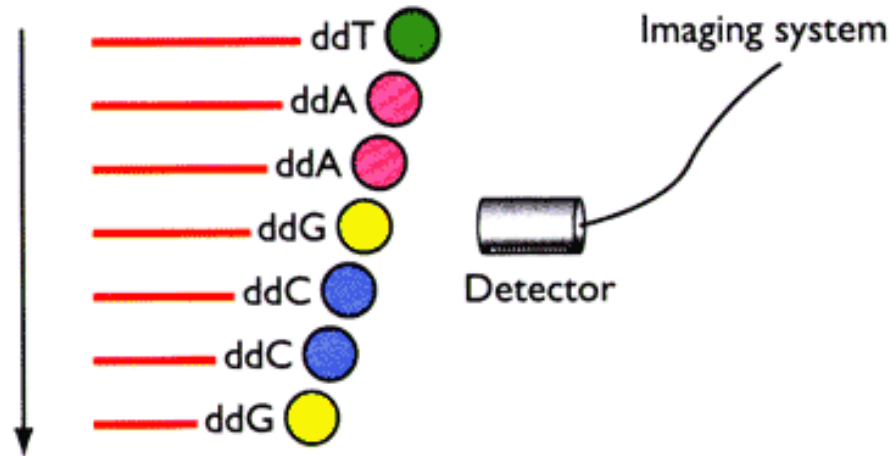
- 获1980年的Nobel奖（Frederick Sanger）
- 其他方法：化学测序法（Maxam-Gilbert法）

第一步：加入复制终止剂

(A)

ddA  ddC  ddNTPs – each with a
ddT  ddG  different fluorescent label

Sequencing reactions,
fractionation of products

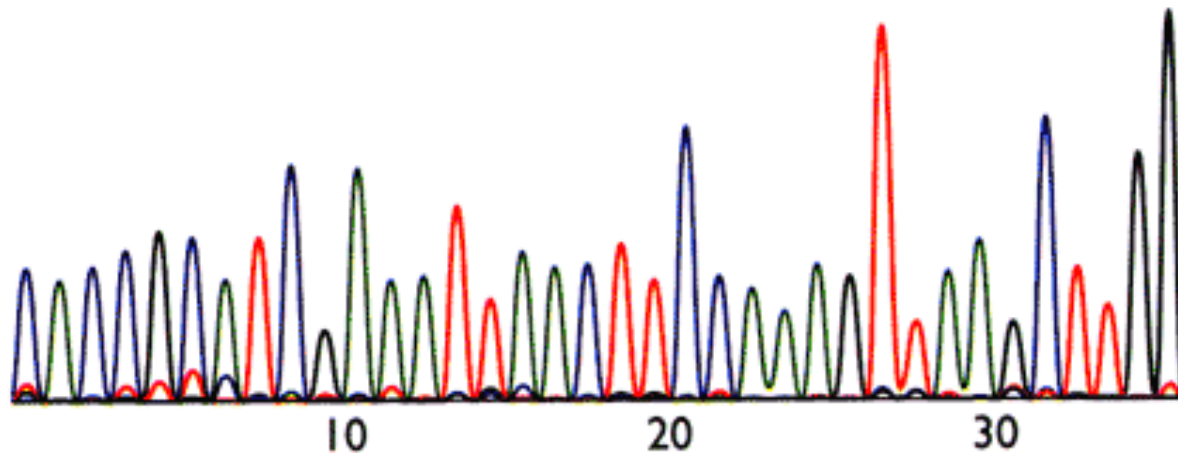


Fluorescent bands
move past the detector

第二步：荧光检测

(B)

CACCGCATCGAAATTAAC TTCCAAAGTTAAGCTTGG



跑得快的先监测到

跑得慢的后检测到

跑得快慢表示长短不同

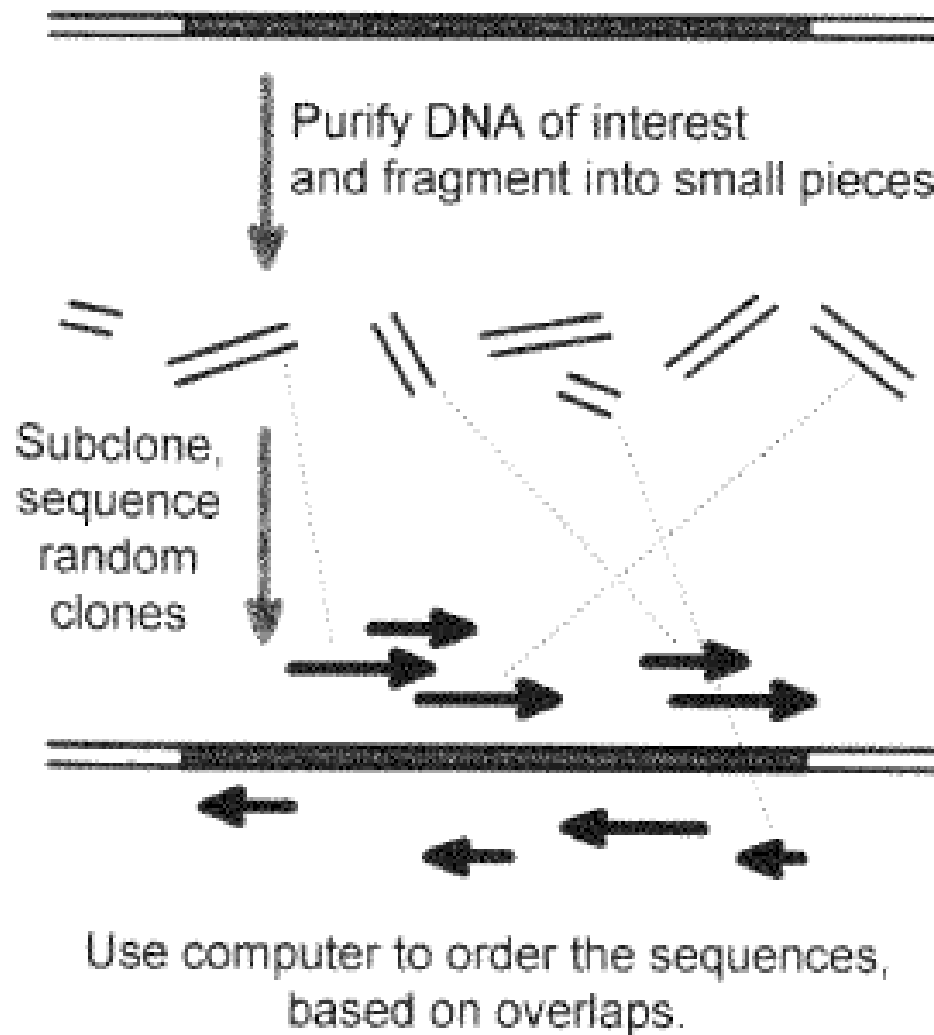
同一个起点开始复制，短的表示在前头，长的表示在后头

根据荧光定序列。

荧光自动测序仪

- 用四种荧光标记代替传统的四个泳道
- 全自动、效率高、精度高
- 速度快：200bp/h
- 并行：同时测定64~96个样品
- 费用：RMB20/500bp
- 一般测序长度：500bp
- 最好的测序仪：1000bp

鸟枪测序法 (Shotgun)





鸟枪测序法 (Shotgun)

- DNA的提取和纯化
- 载体预备：和DNA片断结合，从而能够在细菌中扩增
- DNA片段的制备：将DNA用超声波（或者限制性内切酶）切成能够测序的小片断



鸟枪测序法 (Shotgun)

- 转化培养：小片断和载体结合，植入细菌中进行扩增
- 提质粒：从细菌中提取出繁殖好的质粒
- 电泳检测：检测质量的好坏
- 测序：上测序仪测序



目录

- 电泳测序技术
- **杂交测序技术**
- 新一代测序技术

Sequencing by Hybridization

- Hybridize target to array containing a spot for each possible k -tuple (k -mer)
- The **spectrum** of a sequence
 - multi-set of all its k -long substrings (k -tuples)
- **Goal**
 - reconstruct the sequence from its spectrum
- Pevzner (1989): reconstruction is polynomial

SBH Array

- DNA array (DNA chip) with 4^k probes
 - Target DNA: **AAATGCG**

AAA↵	AAC↵	AAG↵	AAT↵	ACA↵	ACC↵	ACG↵	ACT↵	↵
ATT↵	ATG↵	ATC↵	ATA↵	AGG↵	AGT↵	AGC↵	AGA↵	↵
CCC↵	CCA↵	CCG↵	CCT↵	CAA↵	CAC↵	CAG↵	CAT↵	↵
CTC↵	CTG↵	CTA↵	CTT↵	CGA↵	CGC↵	CGG↵	CGT↵	↵
GGA↵	GGC↵	GGT↵	GGG↵	GAA↵	GAT↵	GAC↵	GAG↵	↵
GTT↵	GTG↵	GTC↵	GTA↵	GCG↵	GCT↵	GCC↵	GCA↵	↵
TTA↵	TTC↵	TTG↵	TTT↵	TAA↵	TAC↵	TAG↵	TAT↵	↵
TGT↵	TGG↵	TGC↵	TGA↵	TCC↵	TCA↵	TCG↵	TCT↵	↵

Experiment Errors

- Hybridization experiments are error prone
- False negative error
 - k -tuple appears in target DNA but does not appear in its measured spectrum
 - Repetition of k -tuple
- False positive error
 - k -tuple does not appear in target DNA but does appear in its measured spectrum

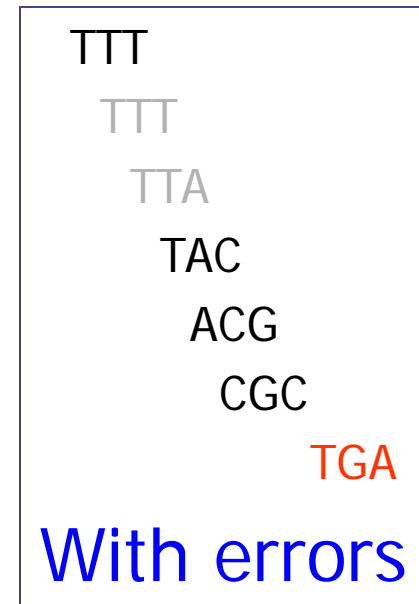
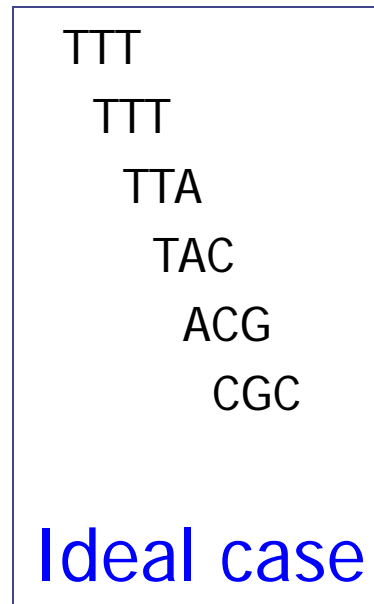
Example

Target DNA

.....TTTTACGC.....



Spectrum

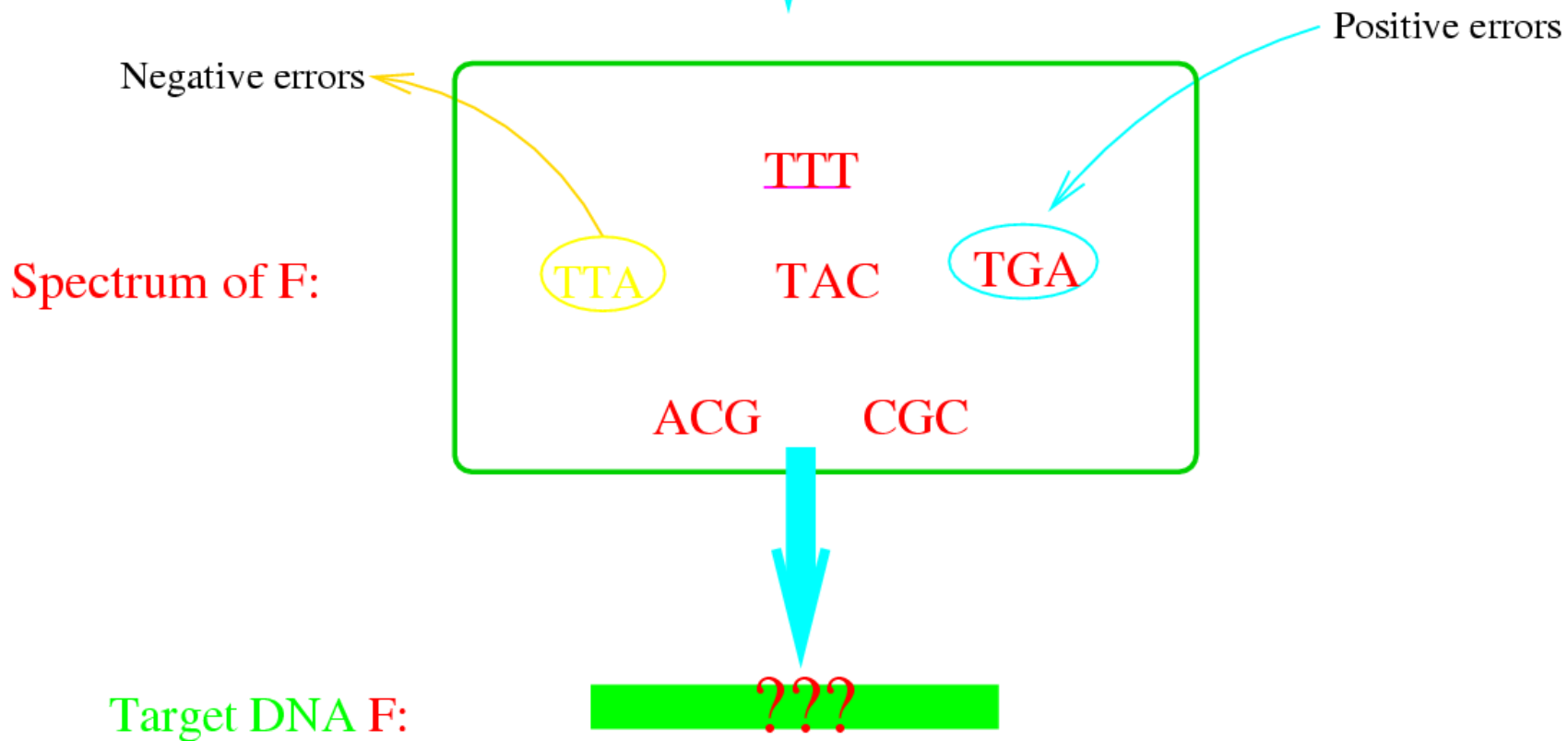


Errors: Positive (misread) / Negative (missing, repetition)



SBH Problem with Errors:

By the hybridization of F=TTTTACGC with C(3)



— : k-tuple repetitions

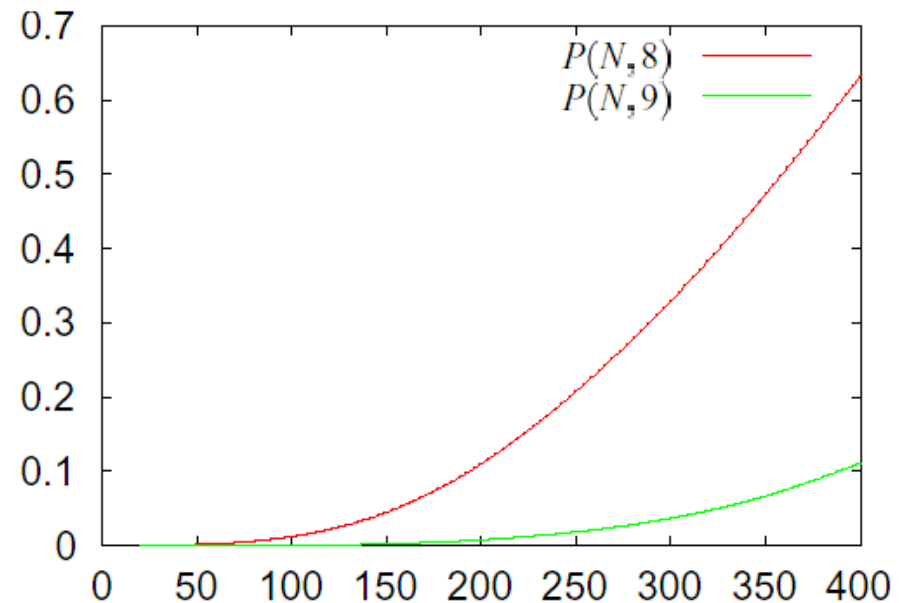


SBH Reconstruction

- In the case of error-free SBH experiments
 - A desired solution of SBH is just a feasible solution including **all k-tuple in the spectrum**
- For the general case
 - There is no additional information except spectrum and the length of target DNA
 - A feasible solution composed of a **maximum cardinality subset of the spectrum** shall be a reasonable desired solution

Uniqueness of Reconstruction

- Different sequences can have the same spectrum:
 - ACT, CTA, TAC
 - ACTAC
 - TACTA
- Non-uniqueness Probability

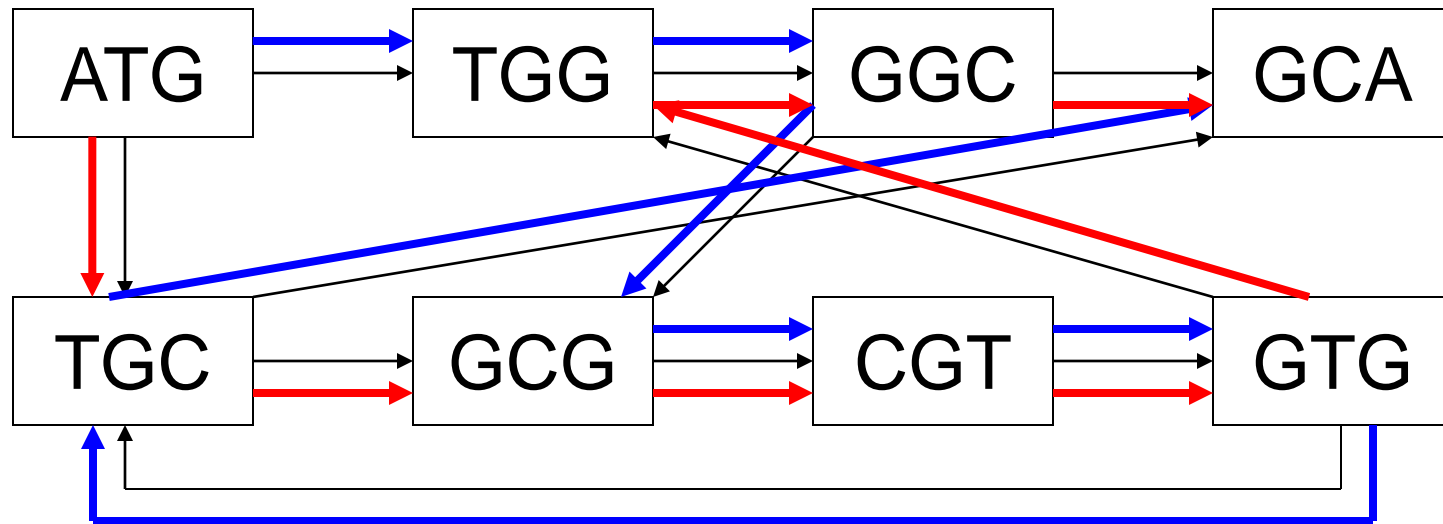


Complexity

- Ideal case (without repetitions and errors)
 - Equivalent to finding an Eulerian path in a corresponding graph (Pevzner, 1989)
 - A linear time algorithm (Fleischner, 1990)
- General case is **NP-hard** problem
 - Exact
 - Heuristics

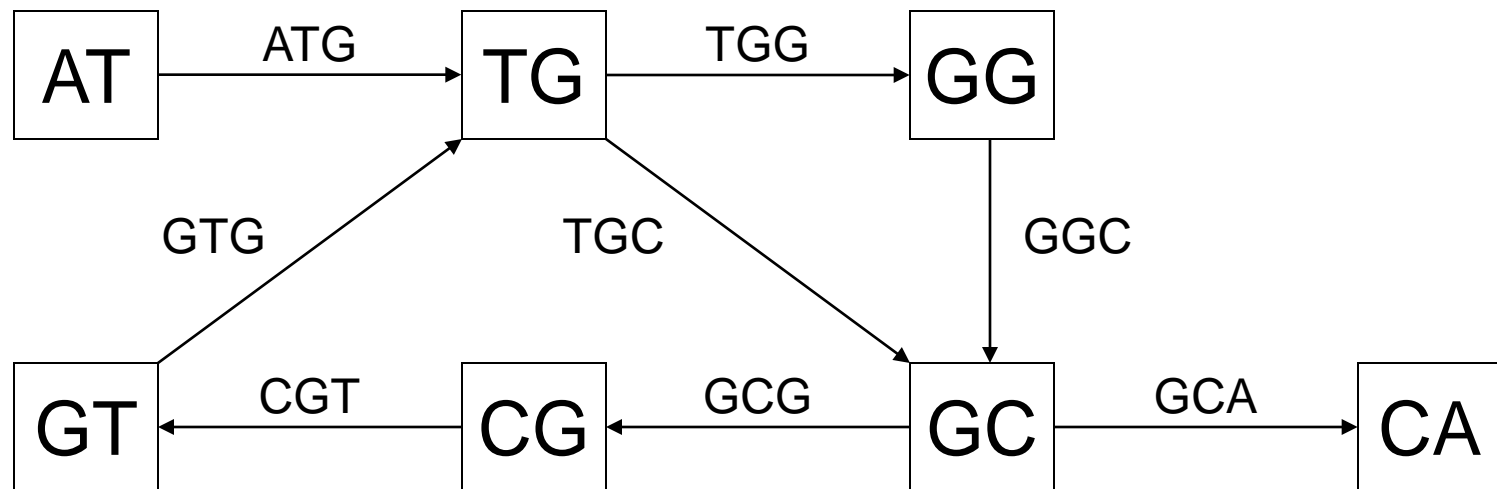
Hamiltonian Path

- {ATG, TGG, TGC, GTG, GGC, GCA, GCG, CGT}



de Bruijn Graph

- {ATG, TGG, TGC, GTG, GGC, GCA, GCG, CGT}



Algorithms

- Genetic Algorithm
 - Gonzalez-Gurrola, LC, Brizuela, CA, Gutierrez, E. A genetic algorithm for the shortest common superstring problem. LECT NOTES ARTIF INT 3315: 851-860 2004
- Tabu Search
 - Blazewicz, J, Formanowicz, P, Kasprzak, M, et al. Tabu search algorithm for DNA sequencing by hybridization with isothermic libraries. COMPUT BIOL CHEM 28 (1): 11-19 FEB 2004
- Probabilistic Method
 - Endo, Takaho A. Probabilistic nucleotide assembling method for sequencing by hybridization. Bioinformatics 20 (14): 2181-8 Sep 2004

Motivations

- Give some criteria which can determine **the most possible k -tuples at both ends** and in the middle of all possible reconstructions of the target DNA
 - These criterions greatly reduce ambiguities in the reconstruction of DNA
- Transform **the negative errors** into the positive errors
 - These means enables us to handle both types of errors easily
- Separate **the repetitions** from both type of errors

Lower Bound

- Estimate the number of k -tuples that does not occur in a solution
 - Adjacency matrix (connection matrix)
 - Give a lower bound of k -tuples that does not occur in all solutions from k -tuple i to j

$$p_{ij}^{n-k} = n_s - 2 - \sum_{t \neq i, j} \bar{a}_{it}^{(n-k-1)} \bar{a}_{tj}^{(n-k-1)} + \delta_{ij},$$



Extensions of SBH

- Positional SBH
 - Broude, N., Sano, T., Smith, C., and Cantor, C. 1994. Enhanced DNA sequencing by hybridization. *Proc. Natl. Acad. Sci. USA* 91, 3072–3076.
- SBH in rounds
 - Margaritis, D., and Skiena, S.S. 1995. Reconstructing strings from substrings in rounds. *36th Annual Symposium on Foundations of Computer Science (FOCS'95)*, 613–620.
- Gapped SBH
 - Preparata, F., Frieze, A., and Upfal, E. 1999. Optimal reconstruction of a sequence from its probes. *J. Comp. Biol.* 6, 361–368.

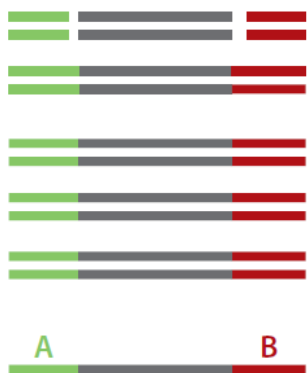
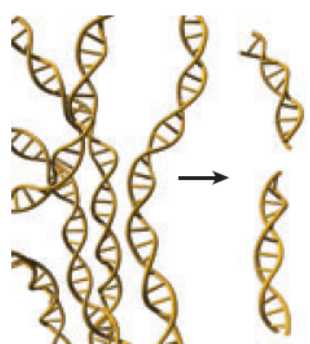


目录

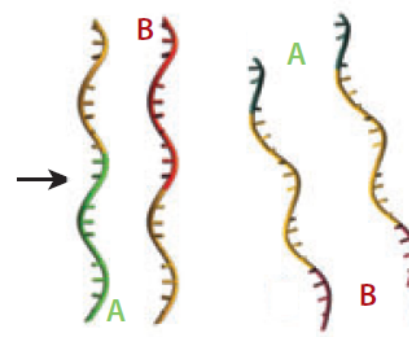
- 电泳测序技术
- 杂交测序技术
- **新一代测序技术**

a**DNA library preparation**

4.5 hours



Ligation

Selection
(isolate AB
fragments
only)

- Genome fragmented by nebulization
- No cloning; no colony picking
- sstDNA library created with adaptors
- A/B fragments selected using avidin-biotin purification

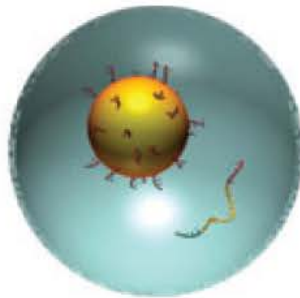
gDNA → sstDNA library

b**Emulsion PCR**

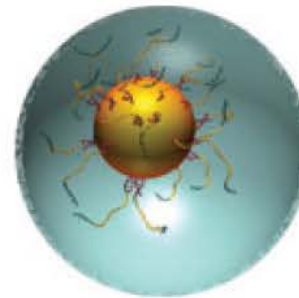
8 hours



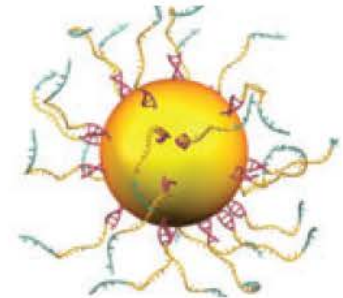
Anneal sstDNA to an excess of
DNA capture beads



Emulsify beads and PCR
reagents in water-in-oil
microreactors



Clonal amplification occurs
inside microreactors



Break microreactors and
enrich for DNA-positive
beads

sstDNA library

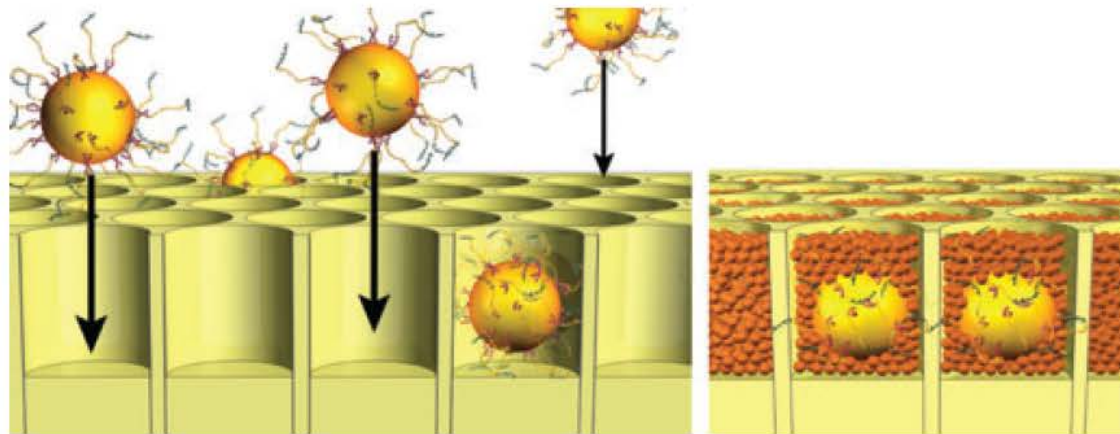
Bead-amplified sstDNA library

Roche 454

C

Sequencing

7.5 hours



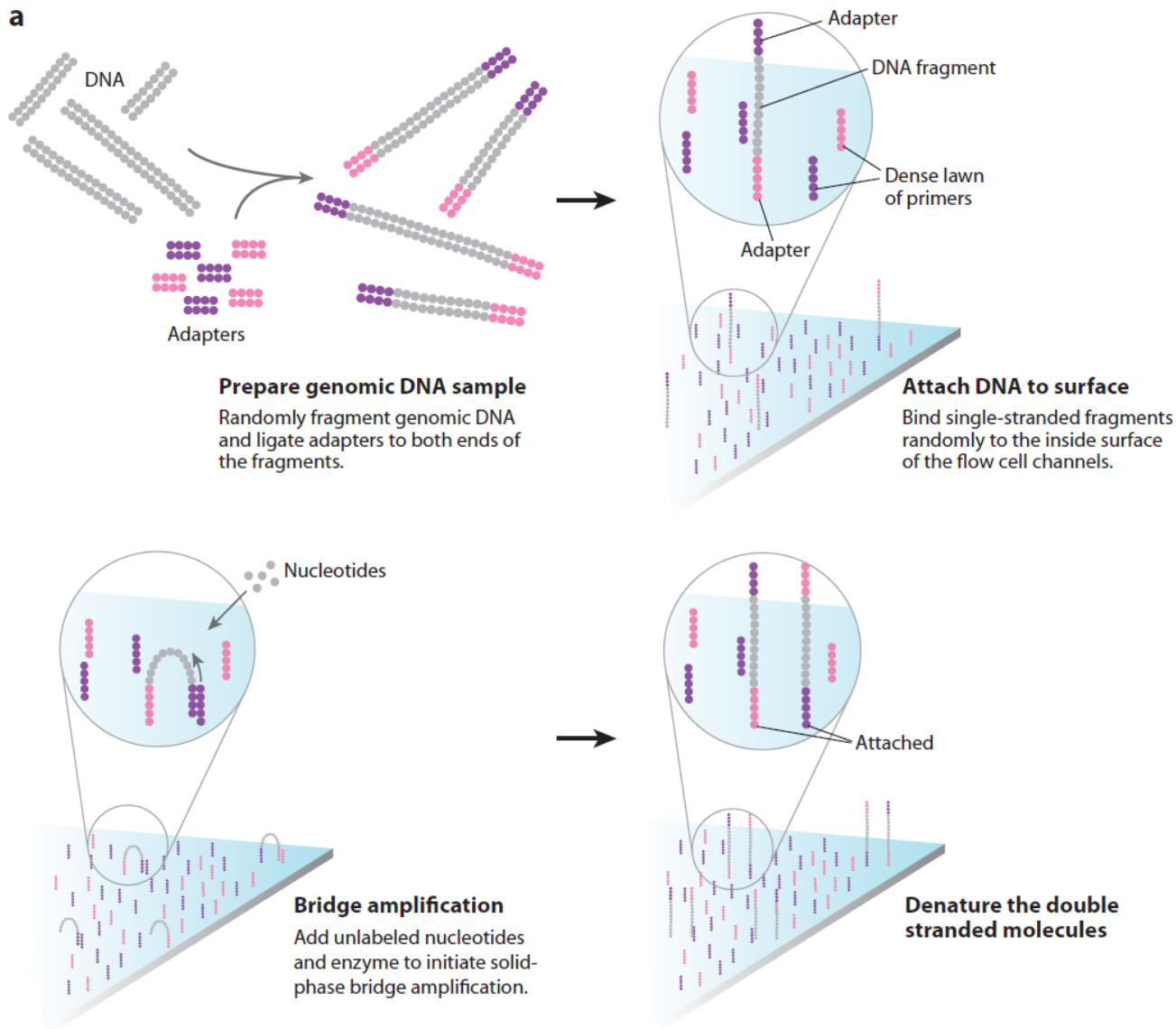
- Well diameter: average of 44 μm
- 400,000 reads obtained in parallel
- A single cloned amplified sstDNA bead is deposited per well

Amplified sstDNA library beads

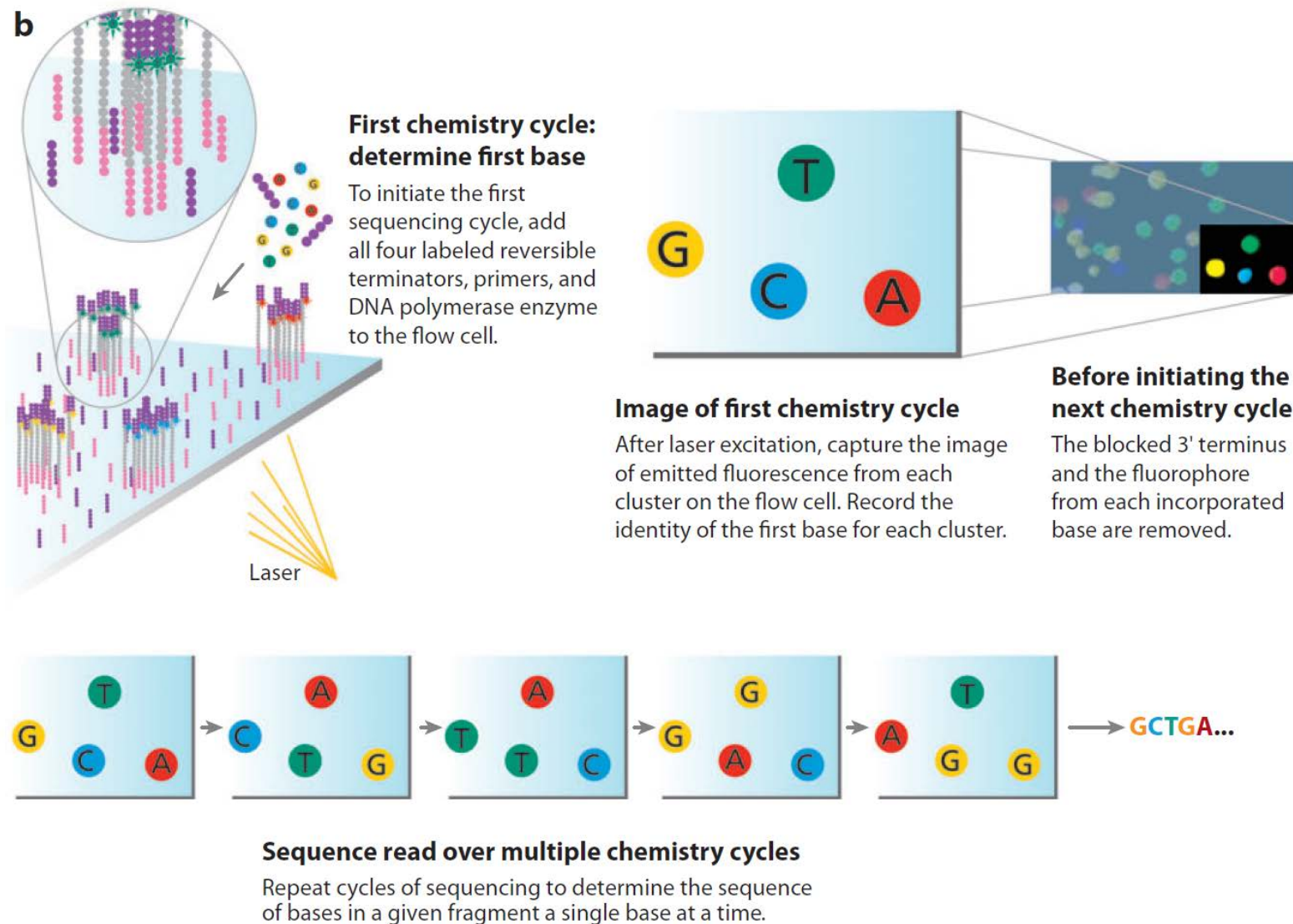


Quality filtered bases

Illumina Solexa



Illumina Solexa



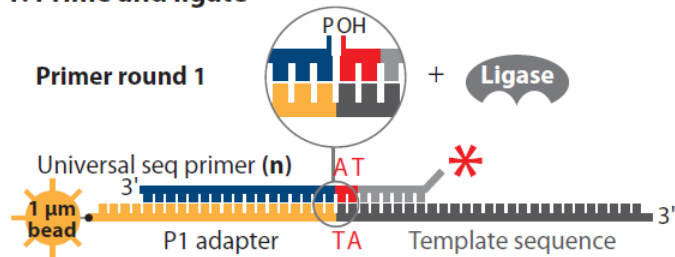
a

The diagram illustrates a DNA template with a cleavage site. The template sequence is 3' Tnnnzzz5', where 'T' is the base at the cleavage site. The bases 'T', 'n', 'n', 'n', 'z', 'z', 'z' are shown in different colors (blue, green, orange, red) and are connected by a vertical yellow bar. To the right, a matrix shows the possible base pairings for the 1st and 2nd bases. The 1st base is labeled '1st base' and the 2nd base is labeled '2nd base'. The matrix is a 4x4 grid of colored circles, with the 1st base on the vertical axis and the 2nd base on the horizontal axis. The colors correspond to the bases: A (blue), C (green), G (orange), and T (red). The matrix shows that the 1st base can be A, C, G, or T, and the 2nd base can be A, C, G, or T, resulting in 16 possible combinations.

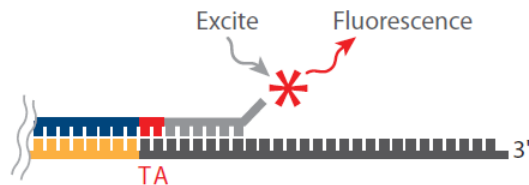
		2nd base				
		A	C	G	T	
1st base	A	●	●	●	●	A
	C	●	●	●	●	C
	G	●	●	●	●	G
	T	●	●	●	●	T

SOLiD

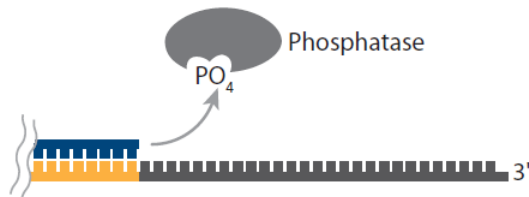
1. Prime and ligate



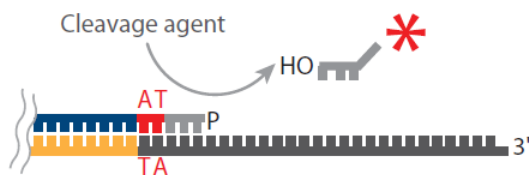
2. Image



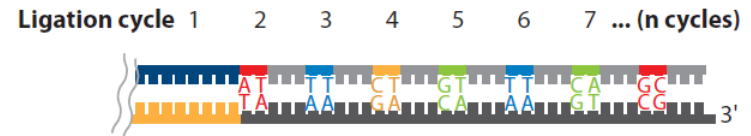
3. Cap unextended strands



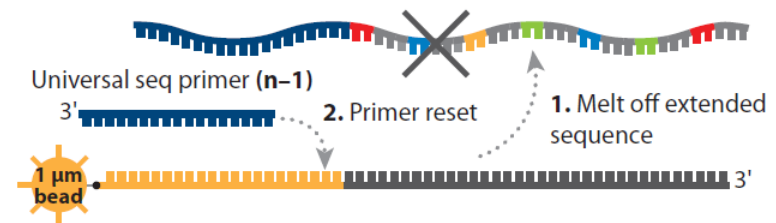
4. Cleave off fluor



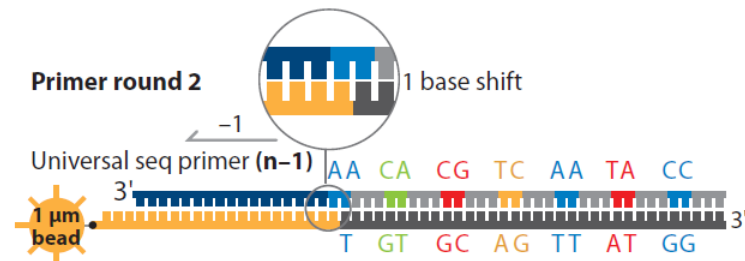
5. Repeat steps 1–4 to extend sequence



6. Primer reset

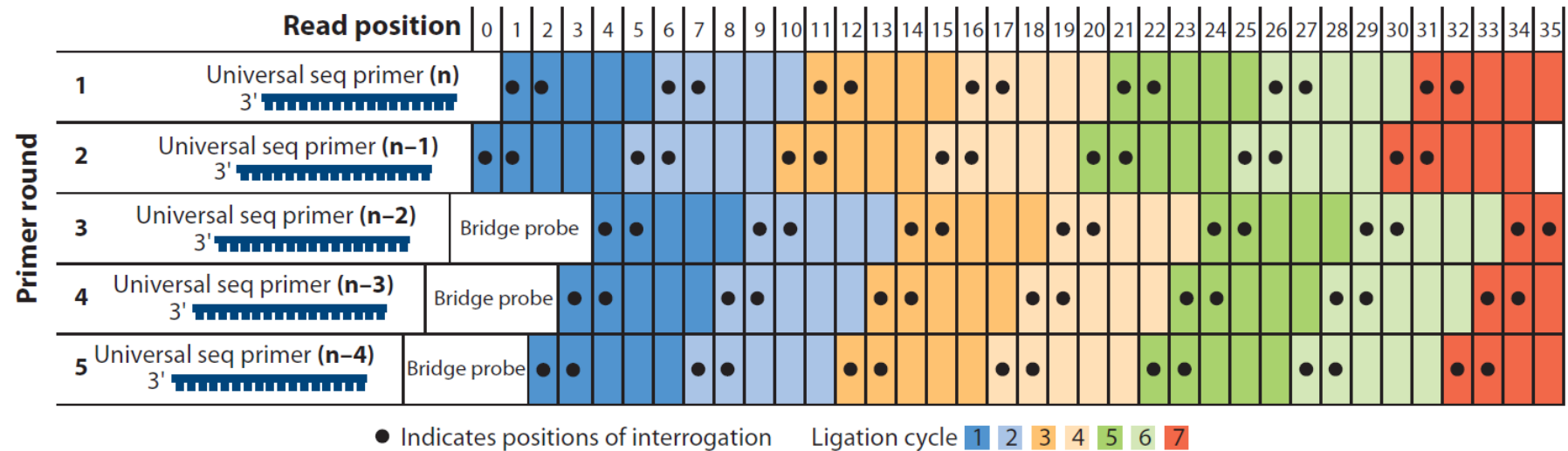


7. Repeat steps 1–5 with new primer



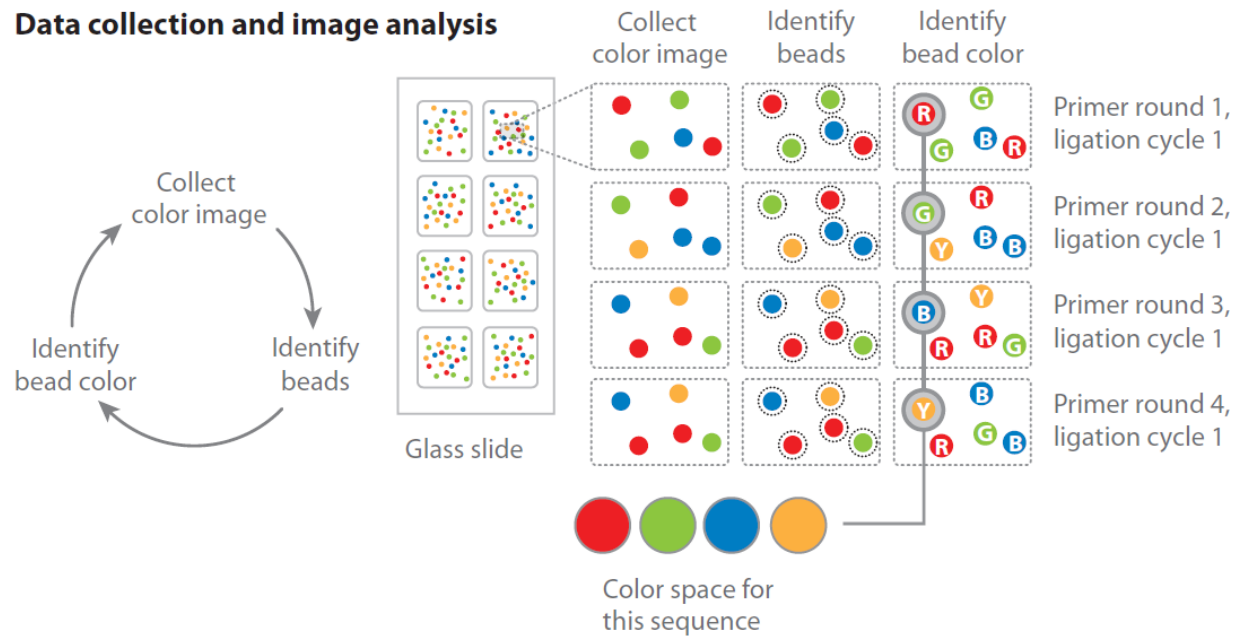
SOLiD

8. Repeat Reset with , $n-2$, $n-3$, $n-4$ primers

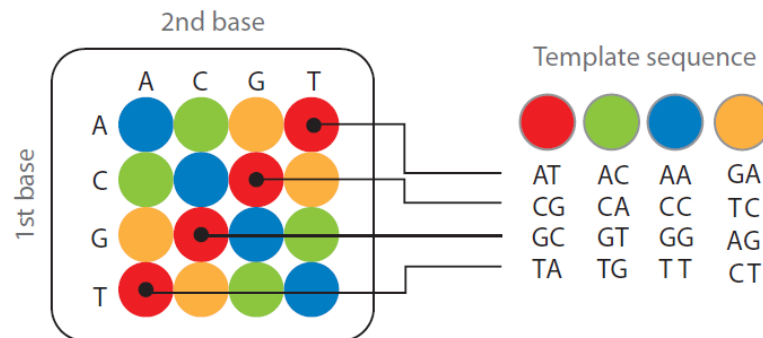


SOLiD

b Data collection and image analysis



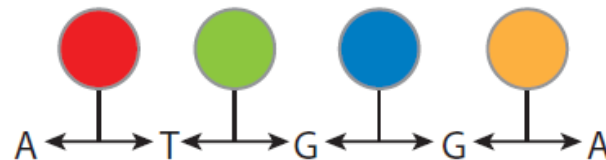
Possible dinucleotides encoded by each color



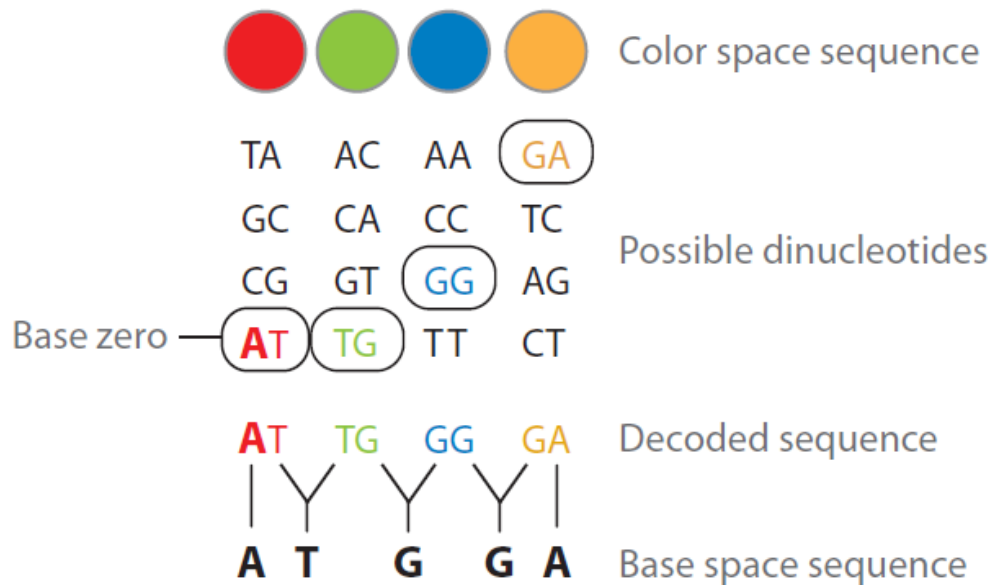
SOLiD

Double interrogation

With 2 base encoding each
base is defined twice



Decoding





Single Molecule Sequencing: HeliScope

- Direct sequencing of DNA molecules: no amplification stage
- DNA fragments are attached to array
- Potential benefits: higher throughput, less errors

Technology Summary

	Read length	Sequencing Technology	Throughput (per run)	Cost (1Mbp) *
Sanger	~800bp	Sanger	400kbp	500\$
454	~400bp	Polony	500Mbp	60\$
Solexa	75bp	Polony	20Gbp	2\$
SOLiD	75bp	Polony	60Gbp	2\$
Helicos	30-35bp	Single molecule	25Gbp	1\$

*Source: Shendure & Ji, *Nat Biotech*, 2008

Applications

- Sanger:
 - Small projects (less than 1Mbp)
- 454:
 - De-novo sequencing, metagenomics
- Solexa, SOLiD, Heliscope:
 - Gene expression, protein-DNA interactions
 - Resequencing



Terminologies

- **Read**: a sequence fragment that comes out of sequencer
- **Mate pair**: a pair of reads from two ends of the same insert fragment
- **Contig**: a contiguous sequence formed by several overlapping reads with no gaps.
- **Supercontig** (scaffold): an ordered and oriented set of contigs, usually by mate pairs.
- **Consensus sequence**: sequence derived from the multiple alignment of reads in a contig

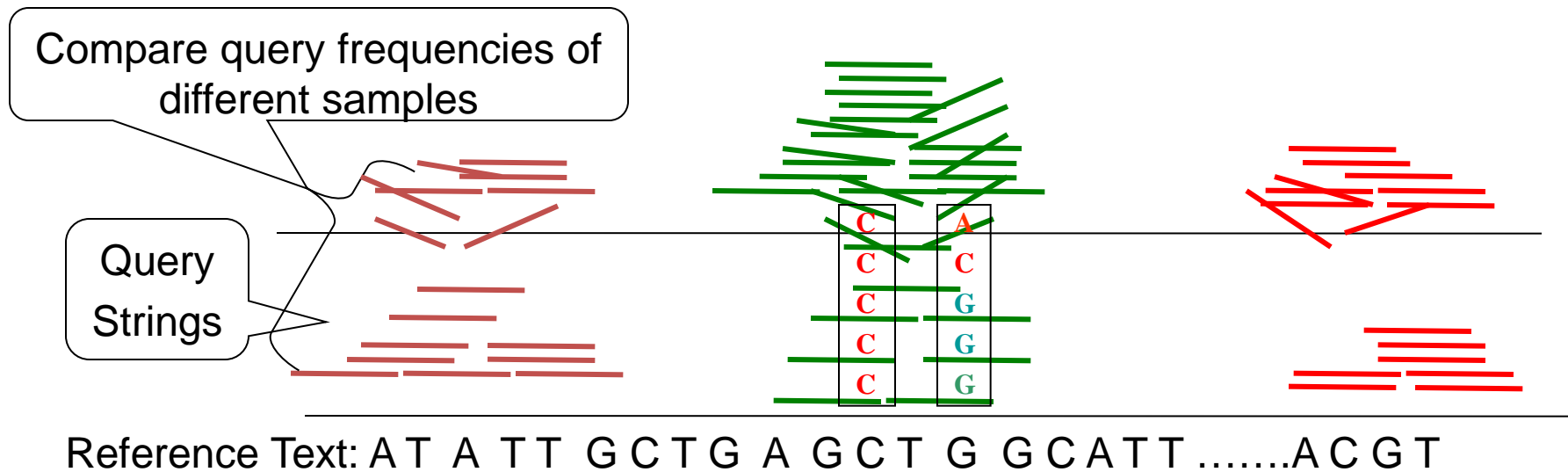


Analysis tasks

- Base calling / polymorphism detection
- Mapping to a reference genome
- *De novo* or assisted genome assembly

Mapping (Alignment)

- Genome re-sequencing
- Gene expression estimation
- String clustering for assembly or metagenomics





New Algorithm is need!

- BLAST is **too slow** because reads are
 - ① Short
 - ② Substituion only
 - ③ Same Length

Requirement

- Speed
- Sensitivity
- Memory usage

Search through the genome once for every query



Build index in advance to accelerate the mapping



Mapping with Errors

GCTGA GCT A
AT C TT GCT G

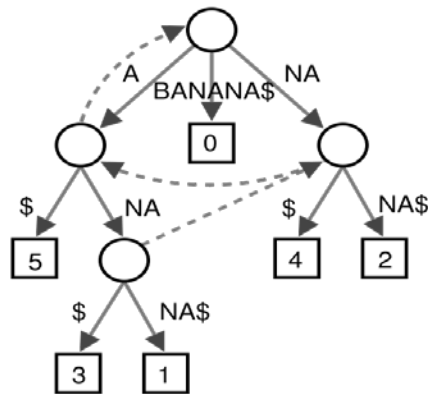
Reference -A T A TT GCTG A GCT G GCATT



How to build a index for string queries with errors

Exact String Matching

- **Suffix Tree** $O(n)$ Time/Space but large memory
- **Suffix Array** Save space, binary search
- **Burrow-Wheeler Transform**



Suffix tree >45x

6	\$
5	A\$
3	ANA\$
1	ANANA\$
0	BANANA\$
4	NA\$
2	NANA\$

Suffix array
(≥ 5 bytes per base)

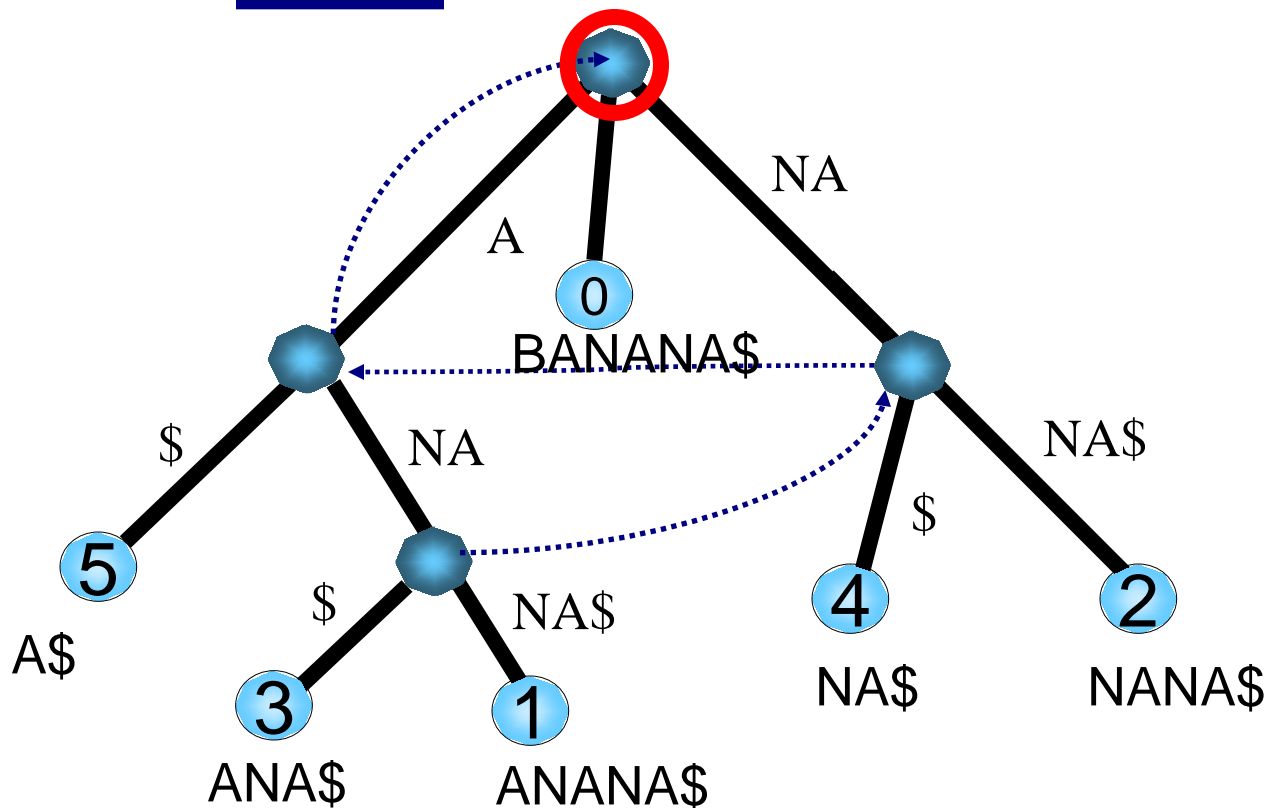
ANANA\$**B**
 ANA\$BAN**N**
 A\$BANAN**N**
 BANANA\$
 NANA\$B**A**
 NA\$BAN**A**
 \$BANANA**A**

BWT index
($\sim 1.65x$ bytes per base)

Suffix Tree

0 1 2 3 4 5 6
 $S = \text{BANANA\$}$

$P = \text{ANA}$



Burrows-Wheeler Transform

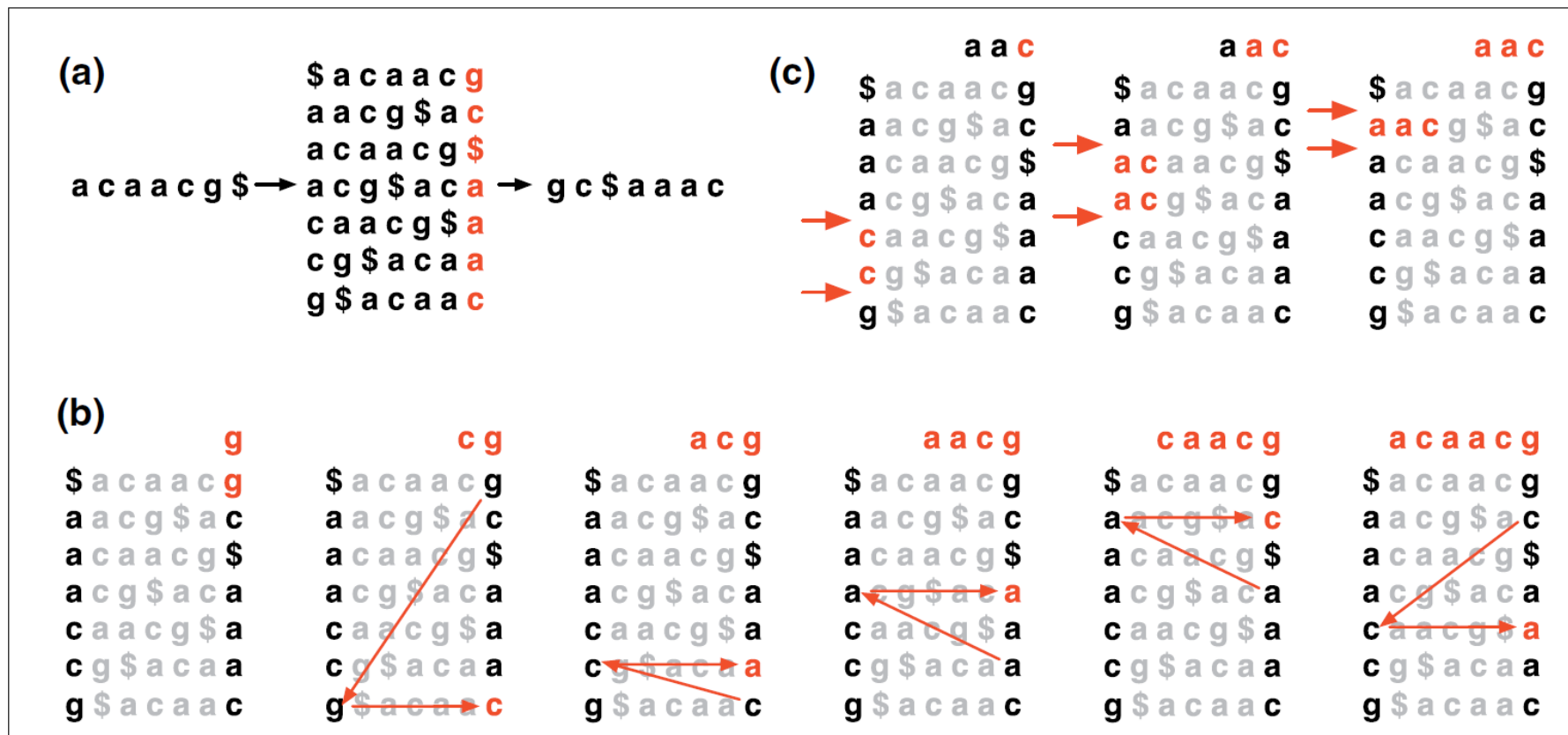


Figure 1
Burrows-Wheeler transform. (a) The Burrows-Wheeler matrix and transformation for 'acaaacg'. (b) Steps taken by EXACTMATCH to identify the range of rows, and thus the set of reference suffixes, prefixed by 'aac'. (c) UNPERMUTE repeatedly applies the last first (LF) mapping to recover the original text (in red on the top line) from the Burrows-Wheeler transform (in black in the rightmost column).



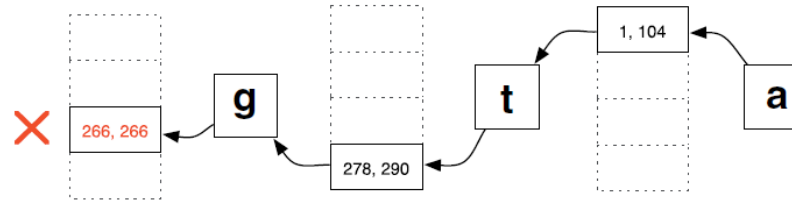
Advantages of BWT

- Easier to be compressed
- Easily to be reversed back
- Exact matching query (FM-index)

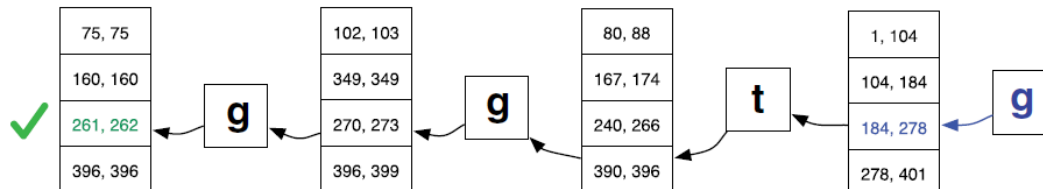
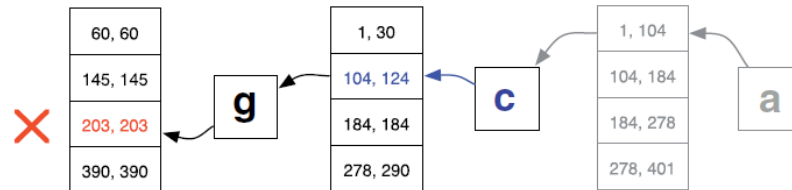
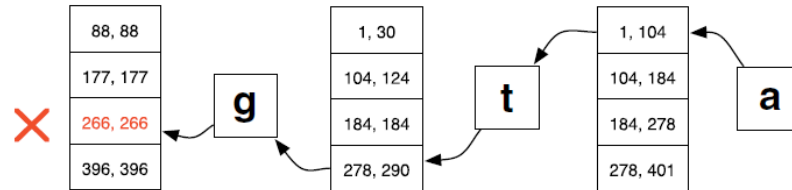


Bowtie

Exact



Inexact



Seed-based Methods

- Build index by **hashing** sub-sequences
- Find alignments only if there is a 12-13 bp exactly matched substring or subsequences.

Query String ———  ———

Reference —  — — — —  ———

Inexact Matching Seeds

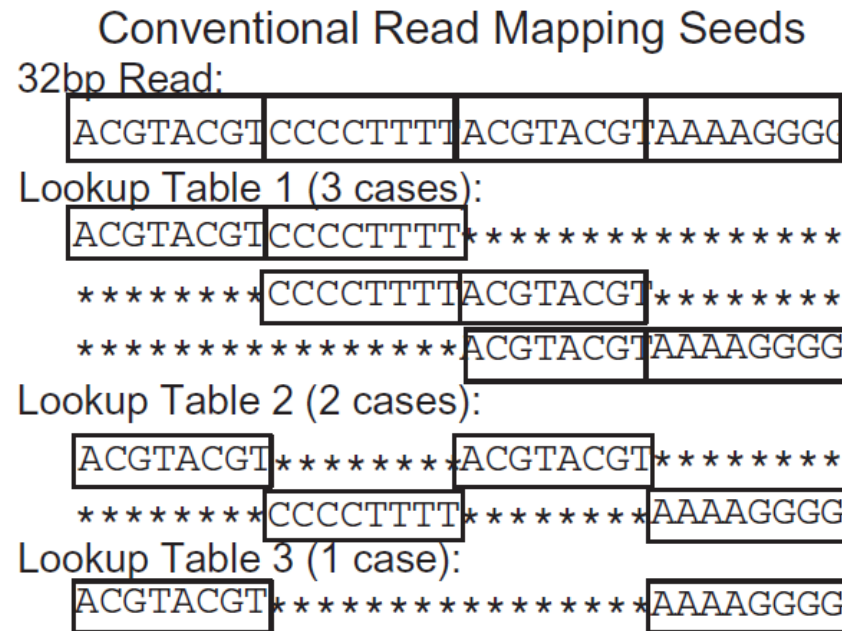


Fig. 1. Conventional seeds used by ELAND, SOAP and MAQ divide a 32 bp read into four substrings. For any alignment within two mismatches, at least one of six pairs of substrings will match exactly. This method requires three hash tables and six lookups for each read and direction (forward or reverse complement).

Single Periodic Spaced Seeds

Single Periodic Spaced Seed

32bp Read:

ACGTACGTCCCCTTTTACGTACGTAAAAGGGG

Lookup the Single Table (7 cases):

ACG*A**	TCC*C**	TTA*G**	CGT*A*****
*CGT*C**	CCC*T**	TAC*T**	GTA*A*****
GTA*G	CCC*T**	ACG*A**	TAA*A****
TAC*T**	CCT*T**	CGT*C**	AAA*G
****ACG*C**	CTT*T**	GTA*G**	AAA*G**
*****CGT*C**	TTT*A**	TAC*T**	AAG*G*
*****GTC*C**	TTT*C**	ACG*A**	AGG*G

Fig. 2. The single periodic spaced seed full sensitive to two mismatches over a 32bp read. For any alignment within two mismatches, at least one out of the seven subsequences will match exactly. This seed is composed of repeating the pattern (111*1**).

NGS Alignment

Table 1: Popular short-read alignment software

Program	Algorithm	SOLiD	Long ^a	Gapped	PE ^b	Q ^c
Bfast	hashing ref.	Yes	No	Yes	Yes	No
Bowtie	FM-index	Yes	No	No	Yes	Yes
BWA	FM-index	Yes ^d	Yes ^e	Yes	Yes	No
MAQ	hashing reads	Yes	No	Yes ^f	Yes	Yes
Mosaik	hashing ref.	Yes	Yes	Yes	Yes	No
Novoalign ^g	hashing ref.	No	No	Yes	Yes	Yes

^aWork well for Sanger and 454 reads, allowing gaps and clipping.

^bPaired end mapping. ^cMake use of base quality in alignment. ^dBWA trims the primer base and the first color for a color read. ^eLong-read alignment implemented in the BWA-SW module. ^fMAQ only does gapped alignment for Illumina paired-end reads. ^gFree executable for non-profit projects only.

Limitations

- The sample may contain sequence that is **absent** or **divergent** from the reference
- Reference sequences, particularly of higher eukaryotes, are **incomplete**, notably in telomeric and pericentromeric regions
- Samples under study may either have **no available reference** sequence or it may not be possible to define a single suitable reference

De novo Assembly

- Sequence assembly problem
 - Find the shortest common sequence of a set of reads
 - Given strings $\{s_1, s_2, \dots\}$ find the shortest string T such that every s_i is a substring of T
 - This is NP-hard
 - Need approximation algorithm

Greedy Algorithm

- Approximation algorithm for this is efficient, the greedy algorithm
 1. calculate pairwise alignments of all fragments
 2. choose two fragments with the largest overlap
 3. merge chosen fragments
 4. repeat step 2. and 3. until only one fragment is left



Greedy Algorithm

- Comments:
 - Greedy algorithm was the first successful assembly algorithm implemented
 - Used for organisms such as bacteria, single-celled eukaryotes
 - It has some efficiency limitations

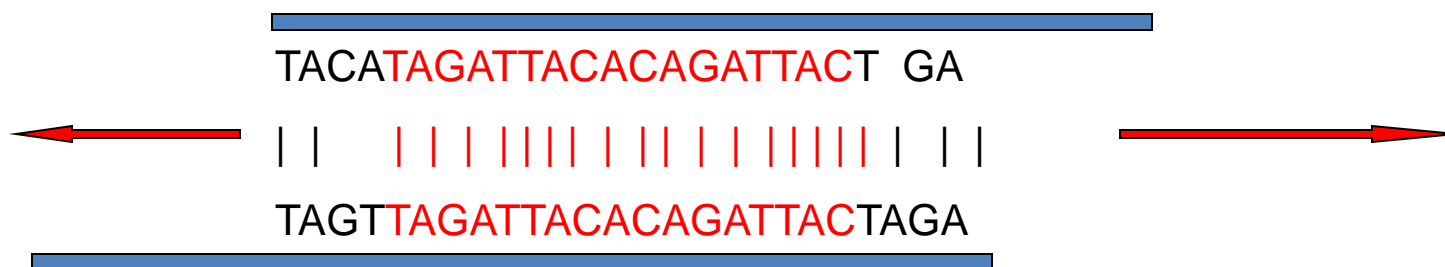


Overlap-layout-consensus

- Algorithm based on graph theory
- A graph is constructed
 - nodes are reads
 - edges represent overlapping reads
- Assemblers based on this approach
 - Arachne, Celera, Newbler, etc

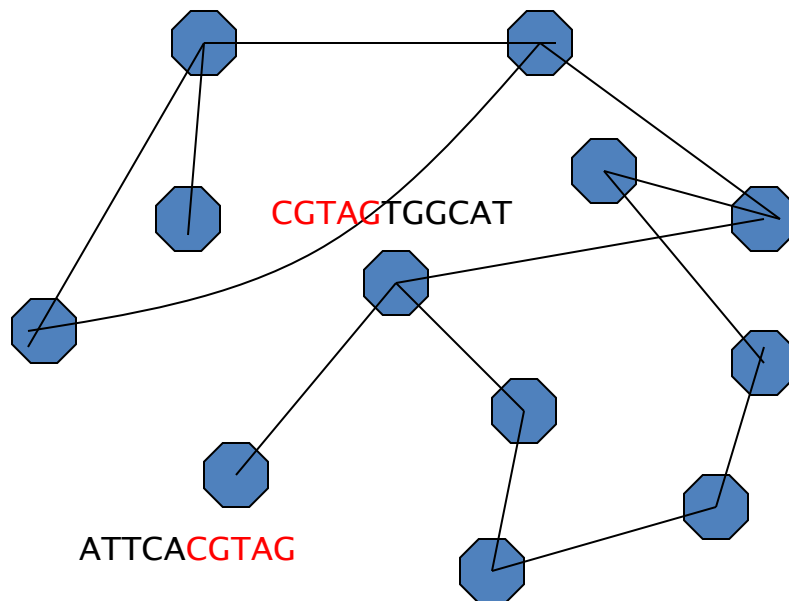
Step 1: Find Overlapping Reads

- Sort all k-mers in reads ($k \sim 24$)
- Find pairs of reads sharing a k-mer
- Extend to full alignment, throw away if not $> 95\%$ similar



Step 2: Construct overlap graph

- A graph is constructed:
 - Nodes are reads
 - Edges represent overlapping reads



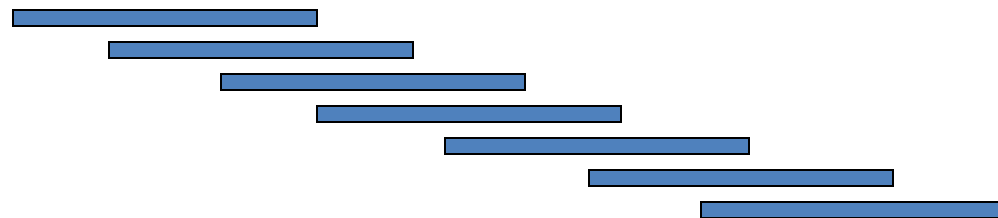
Overlap graph

Step 3: Find Contigs

- Terminology in graph theory:
 - **Simple path**: a path in the graph contains each node at most once.
 - **Longest simple path**: a simple path that cannot be extended.
 - **Hamiltonian path**: a path in the graph contains each node exactly once.

Step 4: Multiple alignment and consensus

- Recall: Now we got several contigs (i.e. several longest simple paths)
- Find the multiple alignments of these contigs, and get one consensus sequence as our final contig.



de Bruijn Graph

- Breaks reads into overlapping k-mers
- Source – $k-1$ prefix and destination is the $k-1$ suffix corresponding to an n-mer
- Basic problem is to find a path that uses all the edges
- Eulerian path – a path that visits all edges of a graph
- Eulerian path is more efficient



Challenge of de novo assembly for NGS

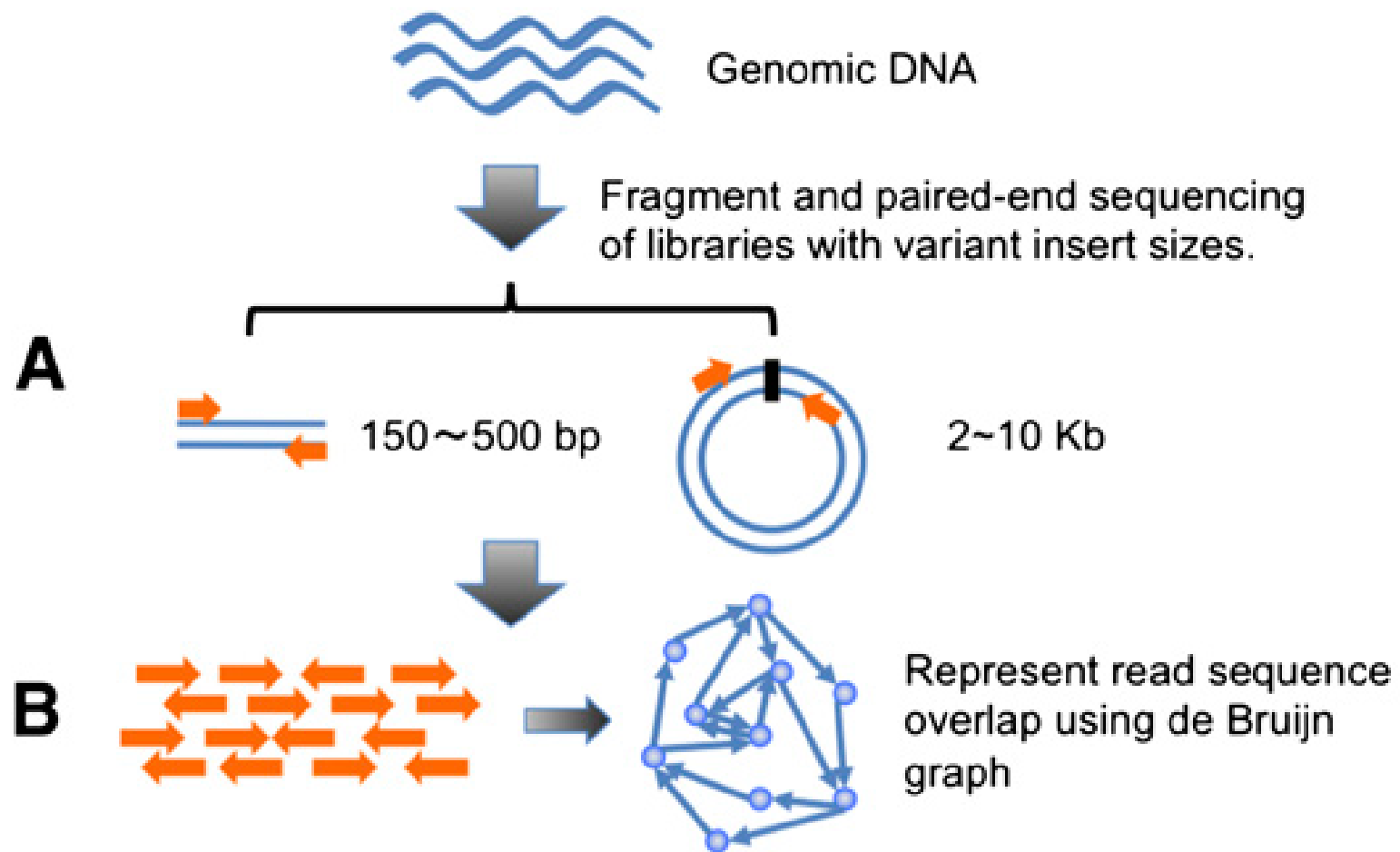
- Large amount of reads
- Short reads
- Repeats
- Sequencing errors
- Computing power
- Memory usage



NGS Assembler

- Velvet
- Edena
- SSAKE
- SHARCGS
- SHRAP
- ALLPATHS
- EULER-SR
-

Workflow



Workflow



Remove erroneous connections on the graph

C

(i) Clip tips



(ii) Remove low-coverage links



(iii) Resolve tiny repeats



(iv) Merge bubbles



D

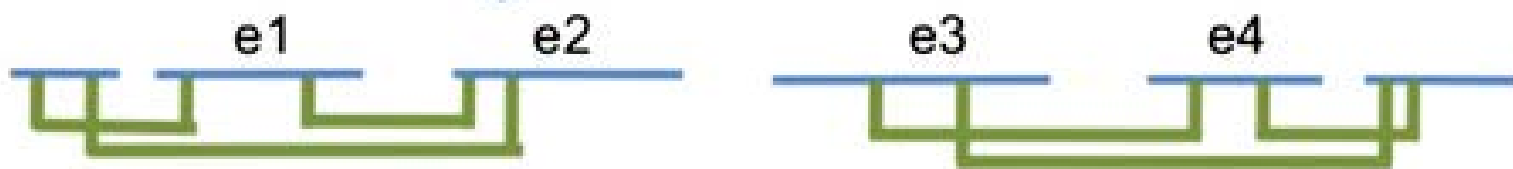


Break at repeat boundaries and output contigs

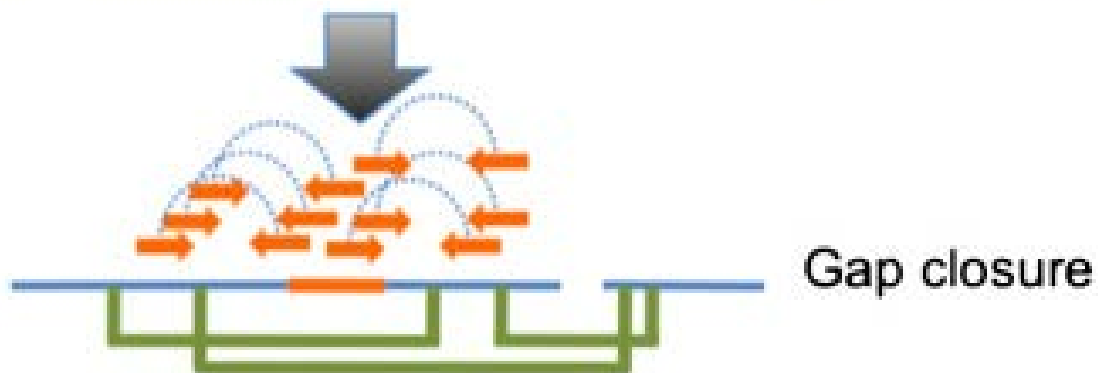
Workflow

Scaffold construction

E



F





Measures of Assembly

- N50
- Largest contig formed
- % bases in contigs \geq 1KB
- Total bases in contigs



Other Problems of NGS

- Base Calling & Quality Control
- Polymorphism detection
- Transcript assembly