Variation detection based on second generation sequencing data

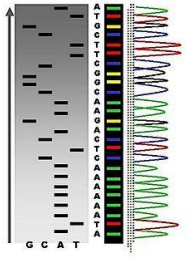
Xin LIU
Department of Science and Technology, BGI
liuxin@genomics.org.cn
2013.11.21

Outline

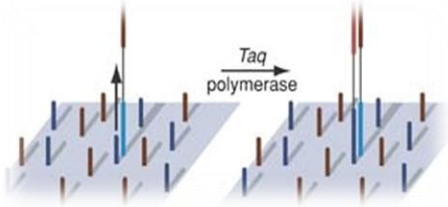
- Summary of sequencing techniques
- Data quality assessing and filtering
- Mapping the short reads
- Detection of SNPs
- Detection of SVs
- Detection of CNVs

华大基因研究院
BGI research

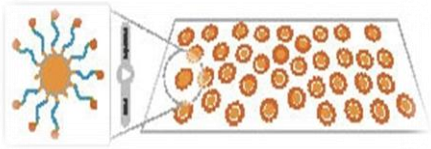
Sequencing technologies



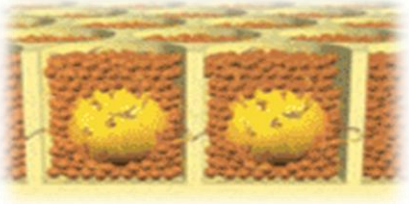
Sanger



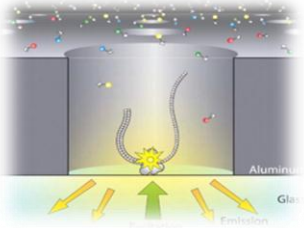
Illumina



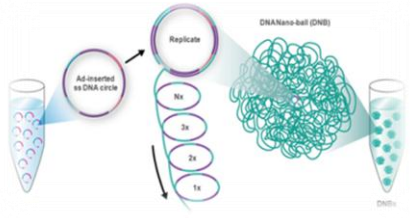
SOLiD



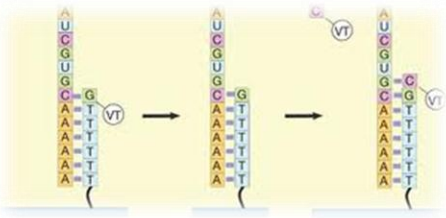
454



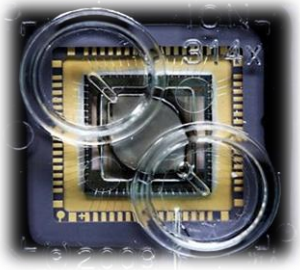
PacBio



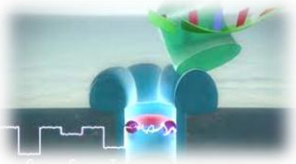
Complete Genomics



Helicos



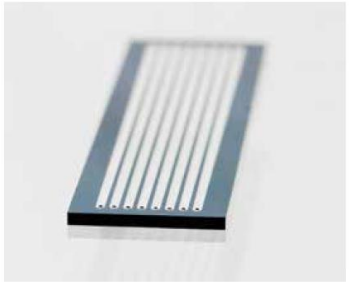
Ion Torrent



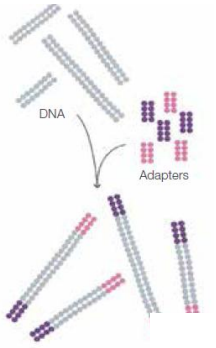
Oxford Nanopore

华大基因研究院
BGI research

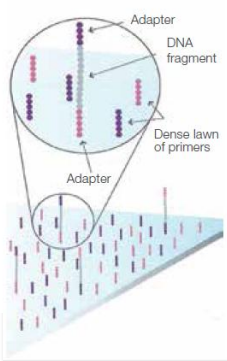
Illumina sequencing



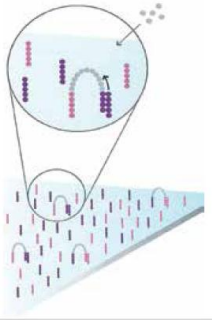
Several samples can be loaded onto the eight-lane flow cell for simultaneous analysis on an Illumina Sequencing System.



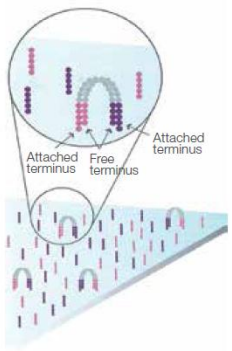
Randomly fragment genomic DNA and ligate a fragments.



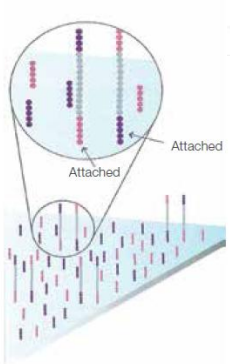
ded fragments randomly to the inside surface of the flow



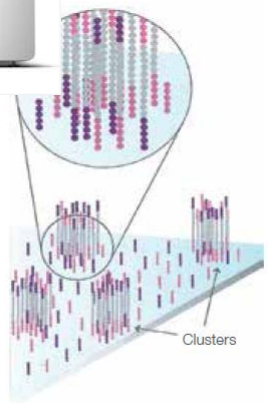
Add unlabeled nucleotides and enzyme to initiate solid-phase bridge amplification.



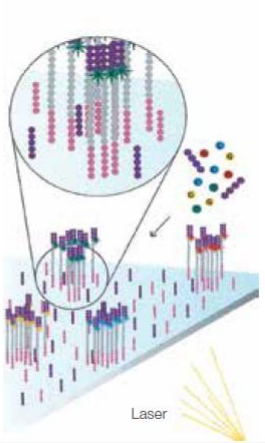
The enzyme incorporates nucleotides to build double-stranded bridge the solid-phase substrate.



Denaturation leaves single-stranded templates anchored to the substrate.

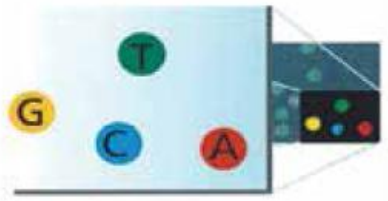


Several million dense clusters of double-stranded DNA are genera each channel of the flow cell.

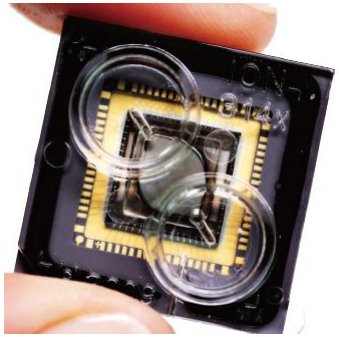


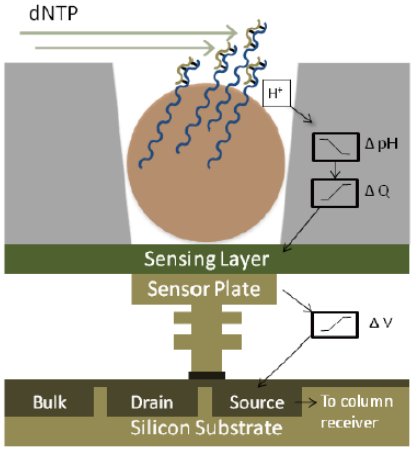
The first sequencing cycle begins by adding four labeled reversible terminators, primers, and DNA polymerase.

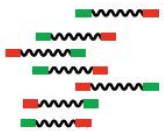
Technology Spotlight: Illumina® Sequencing



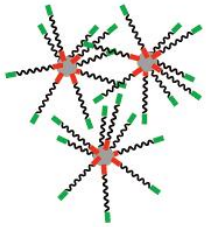
Ion Torrent sequencing





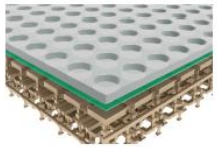


Library Preparation



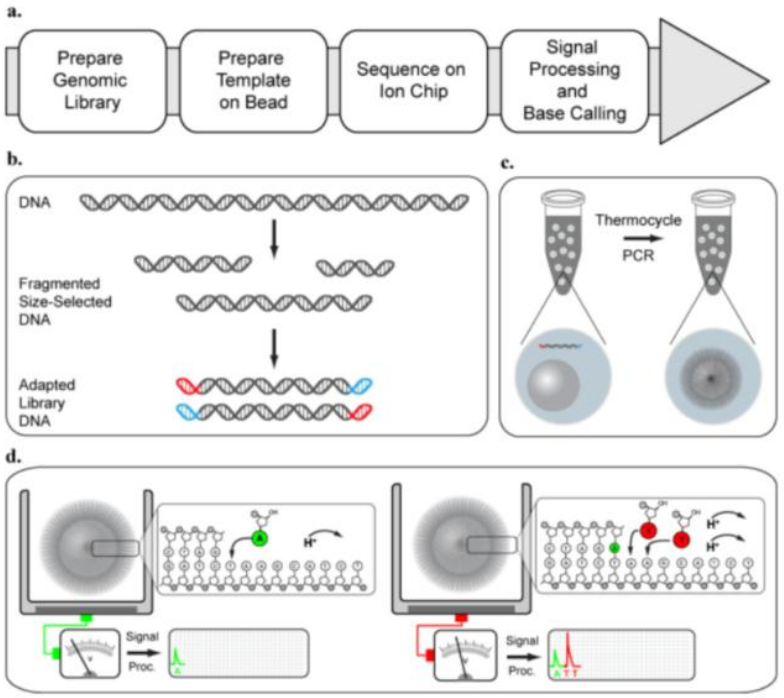
Template Preparation

Ion Semiconductor Sequencing Chip



Ion PGM

Sequencing



http://www.lifetechnologies.com

华大基因研究院
BGI research

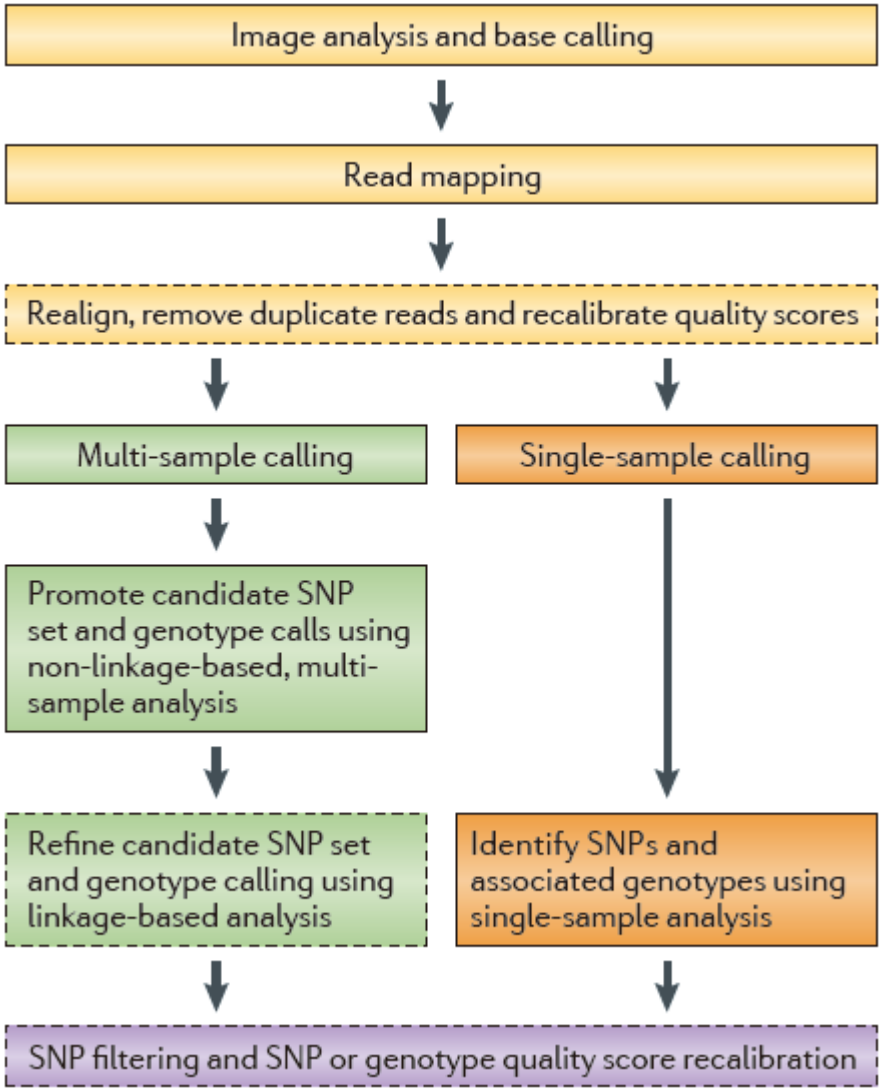
Sequencing techniques

Platform	Illumina MiSeq	Ion Torrent PGM	PacBio RS	Illumina GAIIx	Illumina HiSeq 2000
Instrument Cost*	$128 K	$80 K**	$695 K	$256 K	$654 K
Sequence yield per run	1.5-2Gb	20-50 Mb on 314 chip, 100-200 Mb on 316 chip, 1Gb on 318 chip	100 Mb	30Gb	600Gb
Sequencing cost per Gb*	$502	$1000 (318 chip)	$2000	$148	$41
Run Time	27 hours***	2 hours	2 hours	10 days	11 days
Reported Accuracy	Mostly > Q30	Mostly Q20	<Q10	Mostly > Q30	Mostly > Q30
Observed Raw Error Rate	0.80 %	1.71 %	12.86 %	0.76 %	0.26 %
Read length	up to 150 bases	~200 bases	Average 1500 bases**** (C1 chemistry)	up to 150 bases	up to 150 bases
Paired reads	Yes	Yes	No	Yes	Yes
Insert size	up to 700 bases	up to 250 bases	up to 10 kb	up to 700 bases	up to 700 bases
Typical DNA requirements	50-1000 ng	100-1000 ng	~1 µg	50-1000 ng	50-1000 ng

Quail, M. A., M. Smith, et al. (2012). "A tale of three next generation sequencing platforms: comparison of Ion Torrent, Pacific Biosciences and Illumina MiSeq sequencers." BMC Genomics **13: 341.**

华大基因研究院 BGI research

Flowchart of sequencing analysis



Sequencing data

De novo assembly

Resequencing

Whole genome assembly

Regional/partial assembly

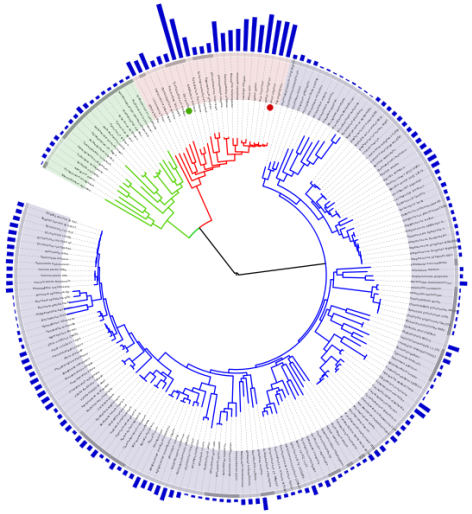
Whole genome resequencing

Target region resequencing

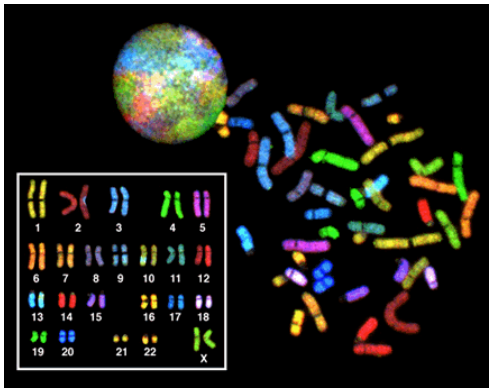
RAD/GBS

Nielsen, R., J. S. Paul, et al. (2011). "Genotype and SNP calling from next-generation sequencing data." Nat Rev Genet **12(6): 443-451.**

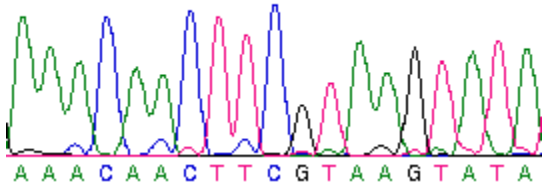
Different genomic variations



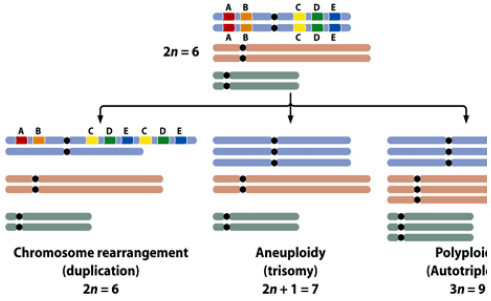
Genome size



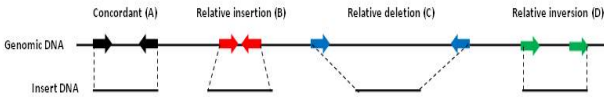
Karyotypes



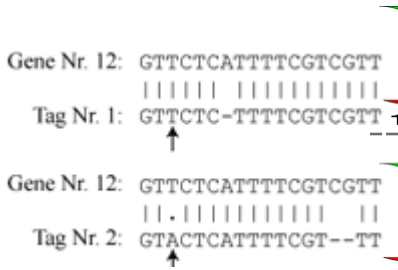
DNA sequences



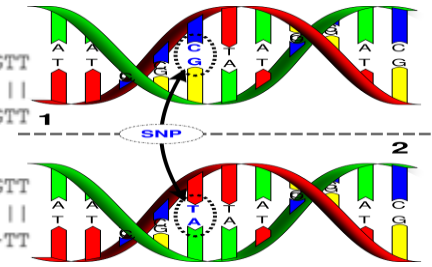
Chromosome rearrangements



Structural variations



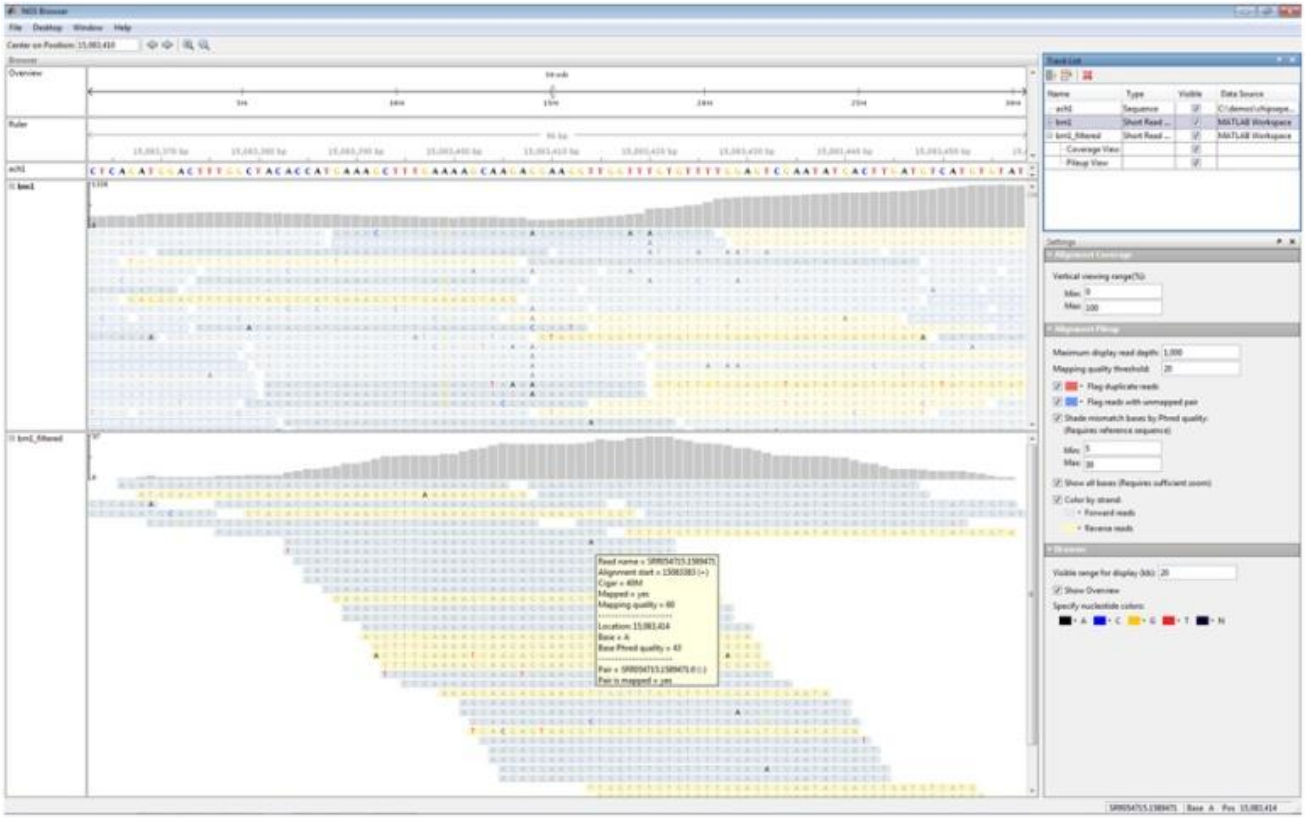
Indels

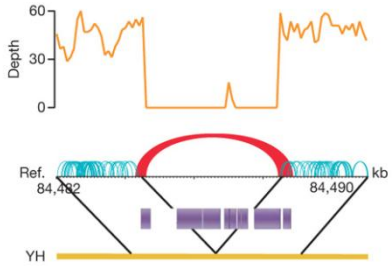


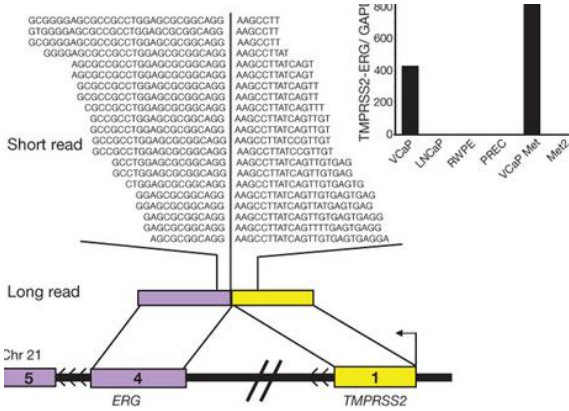
SNPs

华大基因研究院
BGI research

Detection of variations







Beginning, mapping/alignment

- Find the sequenced read's placement in reference genome
- Calculate the coverage and depth distribution of the sequenced reads
- Sequencing quality evaluation
- Important for variation detection

Previous mapping tools

		G	C	C	C	T	A	G	C	G
	0	-2	-4	-6	-8	-10	-12	-14	-16	-18
G	-2	1	-1	-3	-5	-7	-9	-11	-13	-15
C	-4	-1	2	0	-2	-4	-6	-8	-10	-12
G	-6	-3	0	1	-1	-3	-5	-5	-7	-9
C	-8	-5	-2	1	2	0	-2	-4	-4	-6
A	-10	-7	-4	-1	0	1	1	-1	-3	-5
A	-12	-9	-6	-3	-2	-1	2	0	-2	-4
T	-14	-11	-8	-5	-4	-1	0	1	-1	-3
G	-16	-13	-10	-7	-6	-3	-2	1	0	0

Needleman-Wunsch

- Global alignment algorithm
- An example: align COELACANTH and PELICAN
- Scoring scheme: +1 if letters match, -1 for mismatches, -1 for gaps

```
COELACANTH

P-ELICAN--
```

Smith-Waterman

- Modified to do local alignment

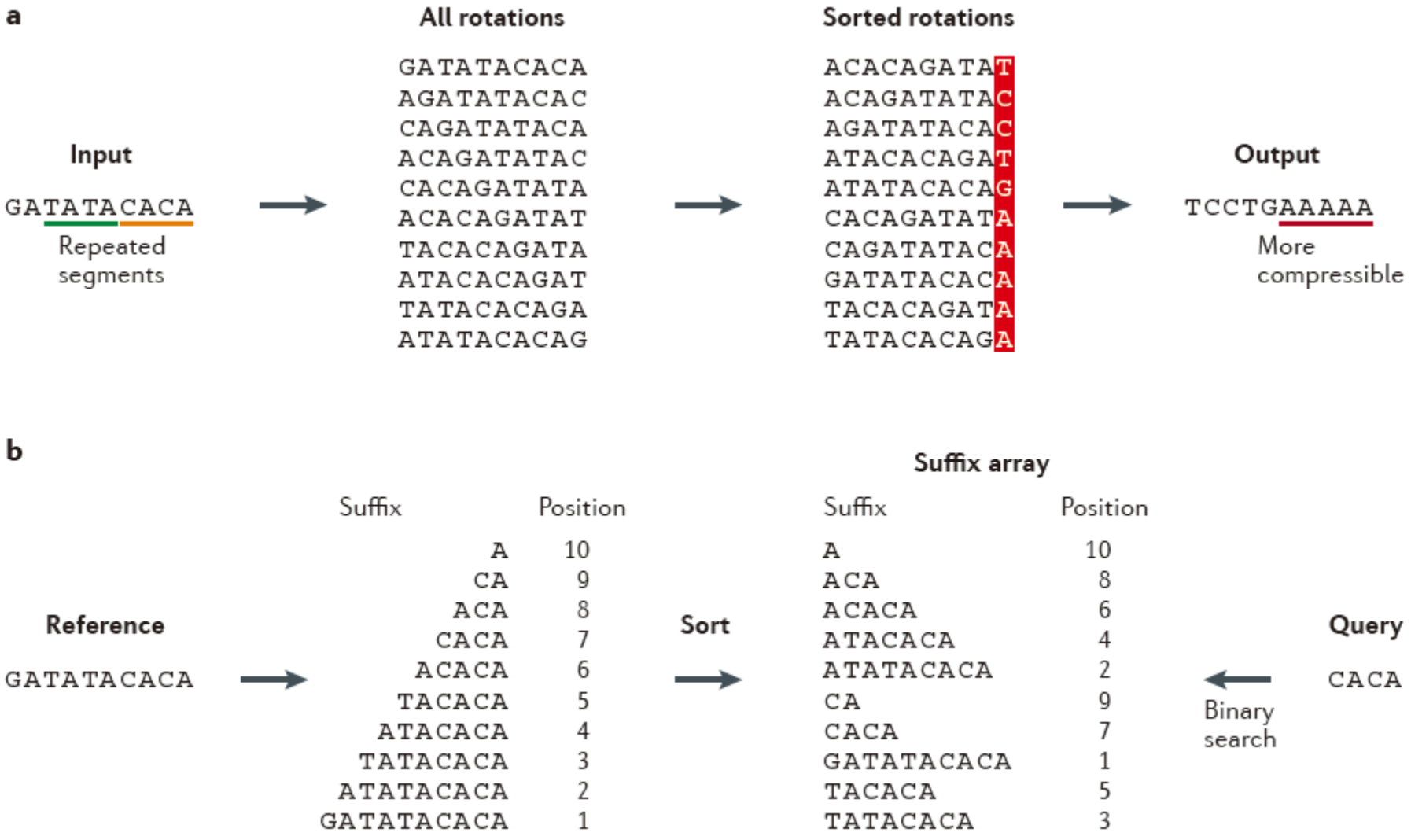
BLAST

- Three heuristic layers: seeding, extension, and evaluation
- Seeding – identify where to start alignment
- Extension – extending alignment from seeds
- Evaluation – Determine which alignments are statistically significant

SOAP

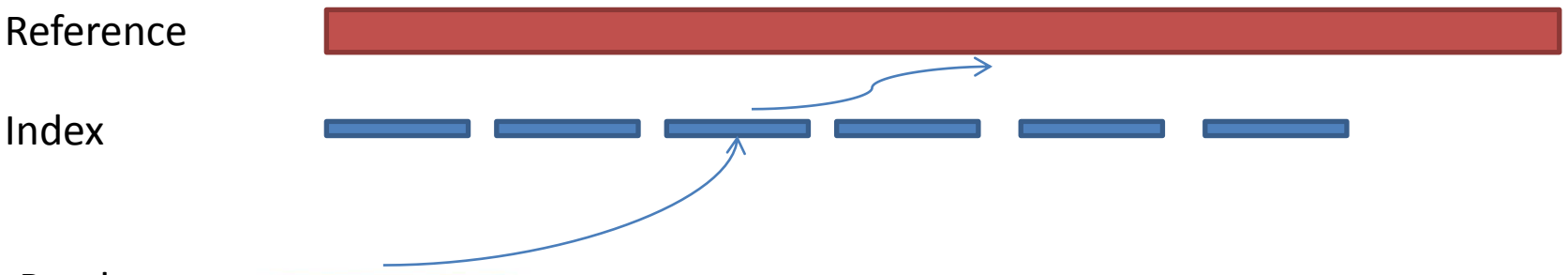
- Differences between traditional and next-generation sequencing technology
  - reads length
  - data capacity
- Algorithm change to meet the data characteristics of the sequencing technology
  - traditional aligner: global or local alignment; scoring matrix; dynamic programming and trace-back
  - Next-Gen aligner: Indexing & Bitwise operation
- Does blastall/blat still work?
- Short Oligonucleotide Alignment/Analysis Package

Index the genome



Algorithm of SOAP

- Comparing to BLAST, BLAT, short reads aligner applied looking-up method.
- Index of the reference were made in order to help the process of looking-up.
- Seed were first looked up by using index and then sequences were extended.



Features of SOAP

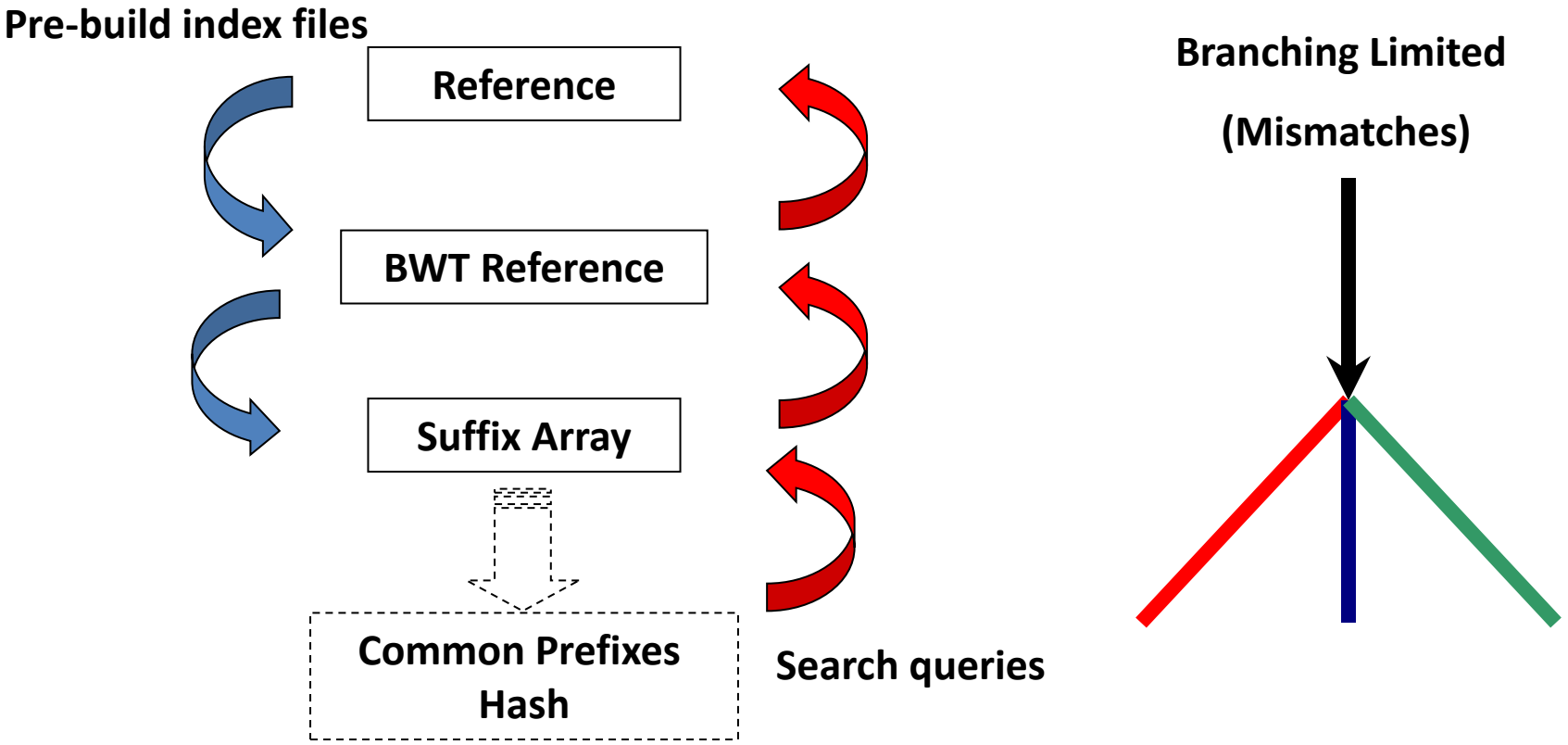
- Fast and efficient
- Mode of pair-end mapping
- Permit gaps within alignments
- Trim of reads permitted

Program	Time consumed (s)	Reads aligned (%)
blastn (−F F −W 11)	165 780	85.47
blastn (−F F −W 15)	150 660	84.66
Blat (−tileSize = 8)	22 032	85.07
Eland	166	88.53
Maq	458	88.39
Soap	134	88.46
Soap iterative	161	90.9
Soap iterative + gapped	486	91.15

Algorithm of SOAP2

- Indexing
  - Split read into parts, which used to anchor the exact matching region in the reference, excluding much of the unwanted region.
- 2way-BWT (Burrows-Wheeler transform) provide a excellent solution for the computing complexity
  - Memory effective (~7G memory need for 3G genome).
  - Fast indexing (2 minutes to finish 1M 35bp single end alignment).
- Thread Parallel Computing
  - Make fully use of process and save time.
- Bitwise operation
  - Encode each base into 2 binary bits, and use exclusive-or to check if two bases are the same.

Flowchart of mapping



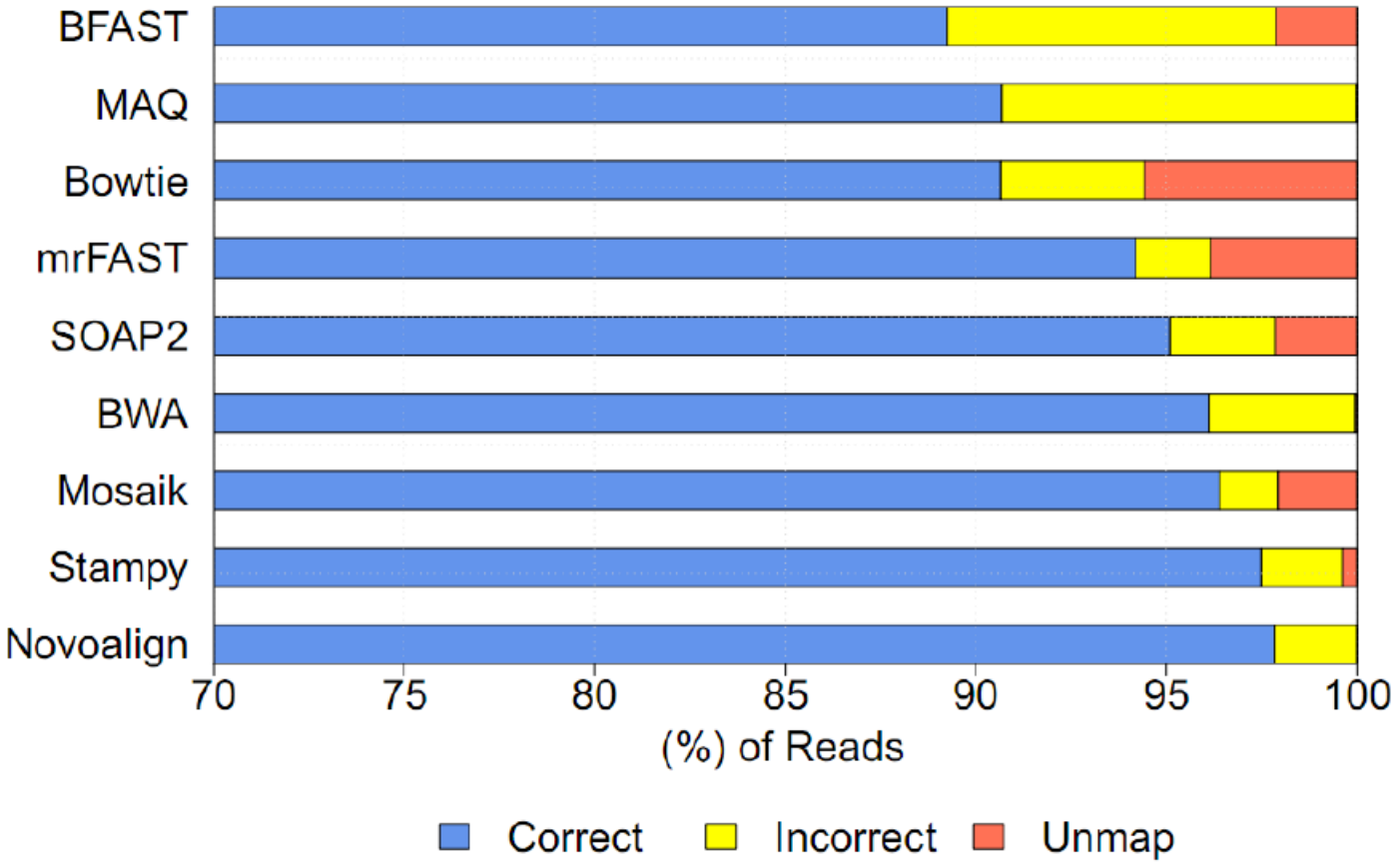
BWA

- Burrows-Wheeler Alignment tool (BWA), the read alignment package that is based on backward search with Burrows–Wheeler Transform (BWT);
- Allowing mismatches and gaps;
- BWA is ~10–20× faster than MAQ, while achieving similar accuracy;
- BWA outputs alignment in the SAM format
- Variant calling can be achieved by SAMtools software package

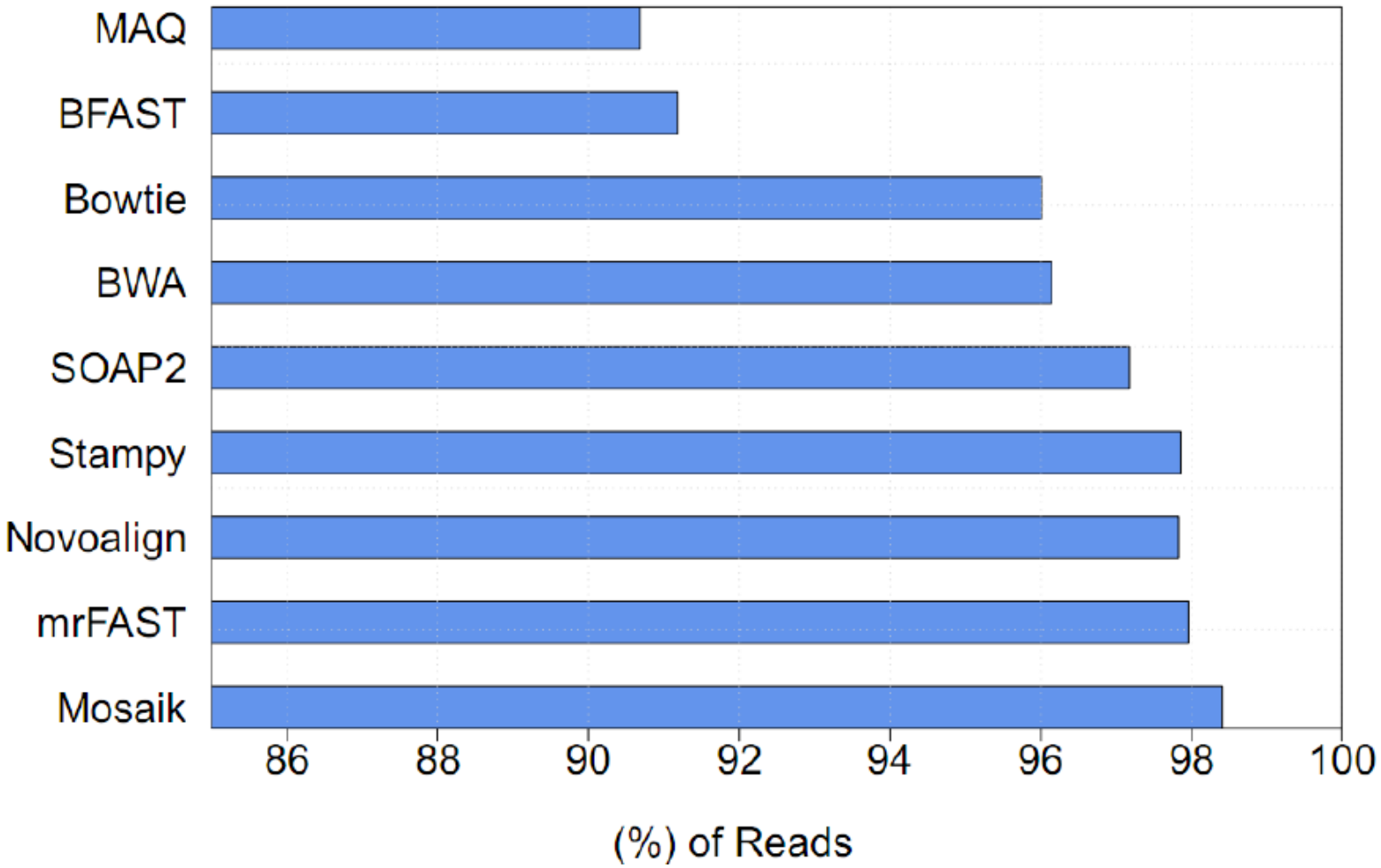
CPU time and RAM

Aligners	CPU-TIME (min)				RAM (G)			
	35	75	90	101bp	35	75	90	101bp
BWA	50	14	13	13	3.2	3.2	3.2	3.1
Bowtie	5	4	4	4	2.9	2.9	2.9	2.9
BFAST	300	360	240	360	16.75	16.75	16.75	16.75
MAQ	300	180	150	132	0.8	0.6	0.6	0.6
Mosaik	122	50	51	60	19	19	19	19
mrFAST	900	284	273	231	8	2	1.7	1.5
Novoalign	149	18	20	21	5.5	5.5	5.5	5.5
SOAP2	9	6	7	9	5.6	5.6	5.6	5.6
Stampy	206	109	60	59	2.7	2.7	2.7	2.7

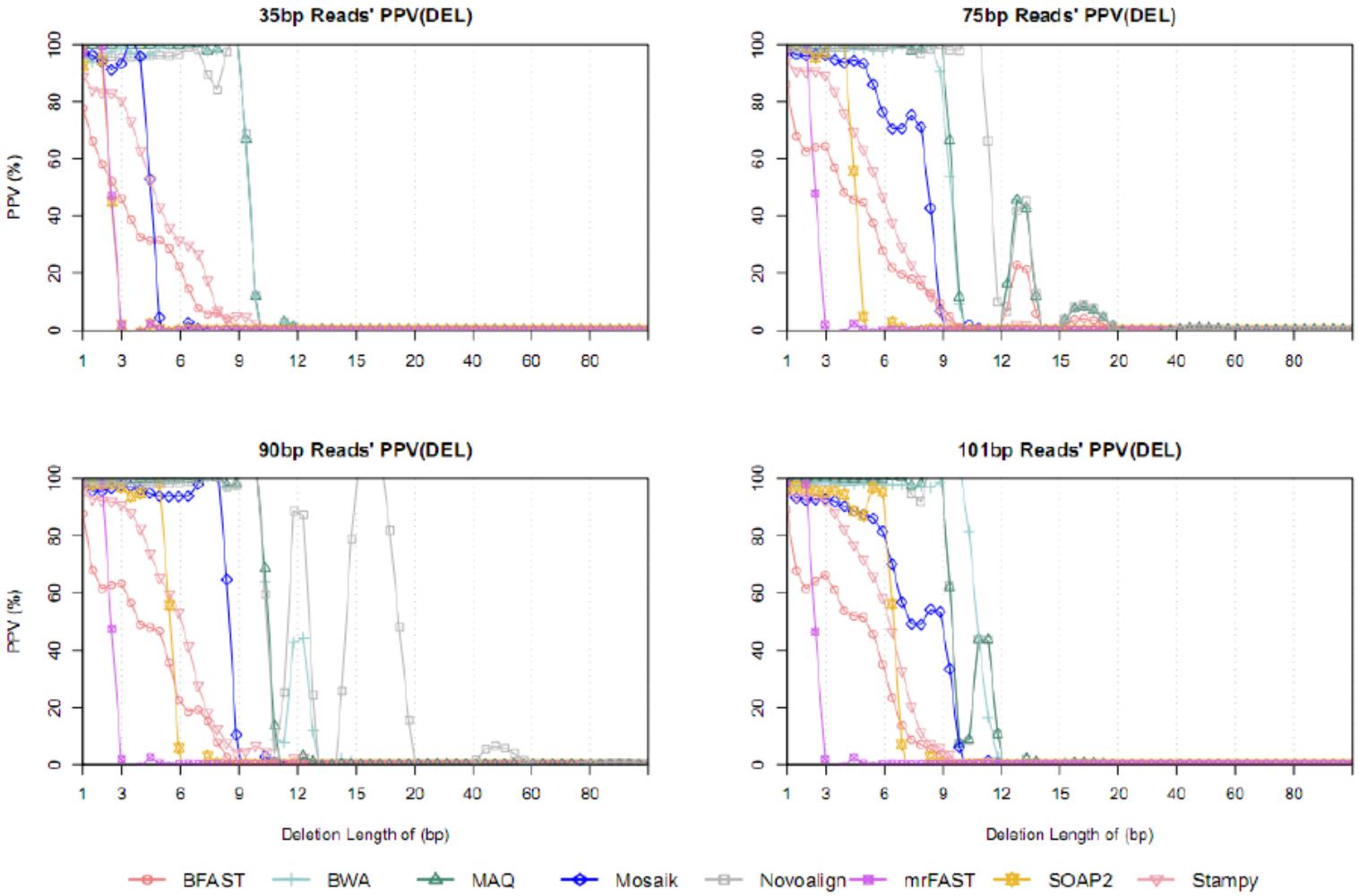
Sensitivity of mapping



Correctness of mapping



Alignment with insertion



Summary of mapping

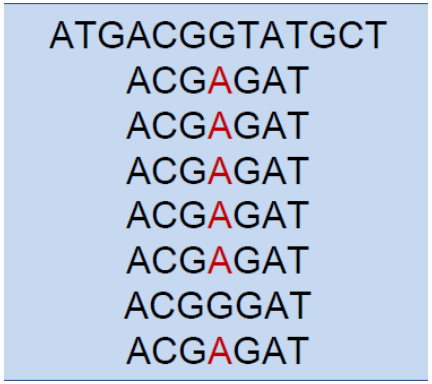
- Short reads mapping need specific aligners
- Many aligners are available and there are different features.
- No best aligner exists, and most of them are acceptable.
- Mapping is important in variation detection.

Importance of SNPs

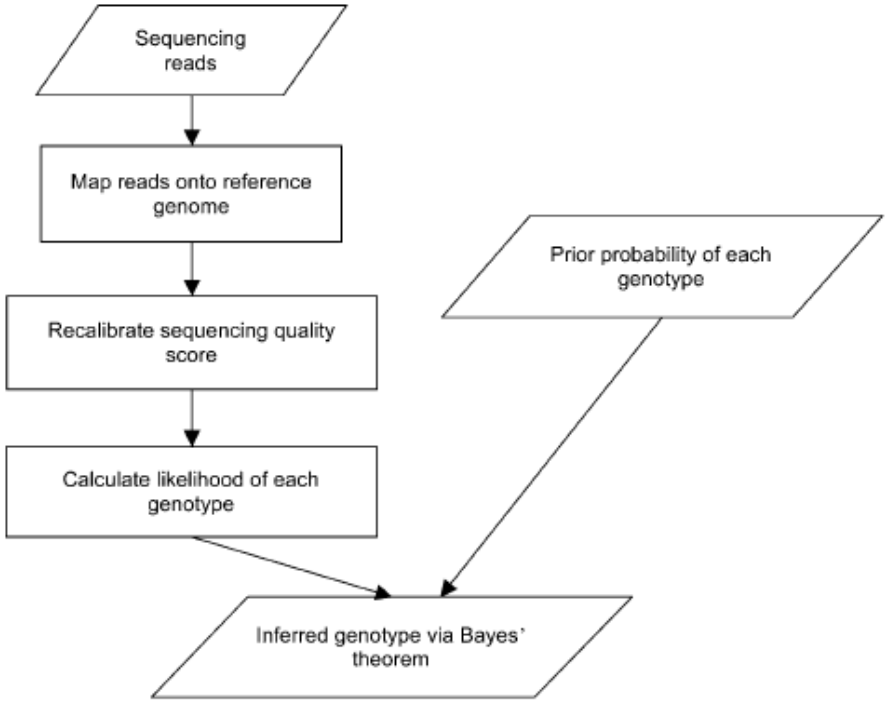
- Well-studied variation
- Better representing demographic history
- Method of detection is relatively mature
- Provides more information for follow-up studies
- Detect SNPs in individuals
- Detect SNPs in population

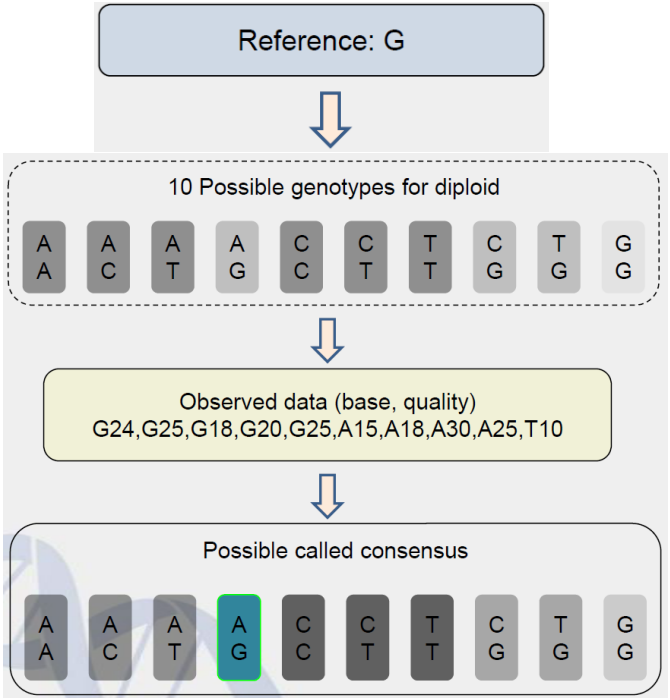
SOAPsnp to detect SNPs

- SOAPsnp was developed for consensus calling and SNP detection based on the Solexa sequencing technology.
- SOAPsnp takes Bayes's theorem as statistic model for SNP calling, it considers:
  - Sequencing quality
  - Likelihood calculation based on observed data
  - Experiment factors
  - Prior probability
  - Alignment uniqueness and accuracy
  - Using dbSNP as prior probability



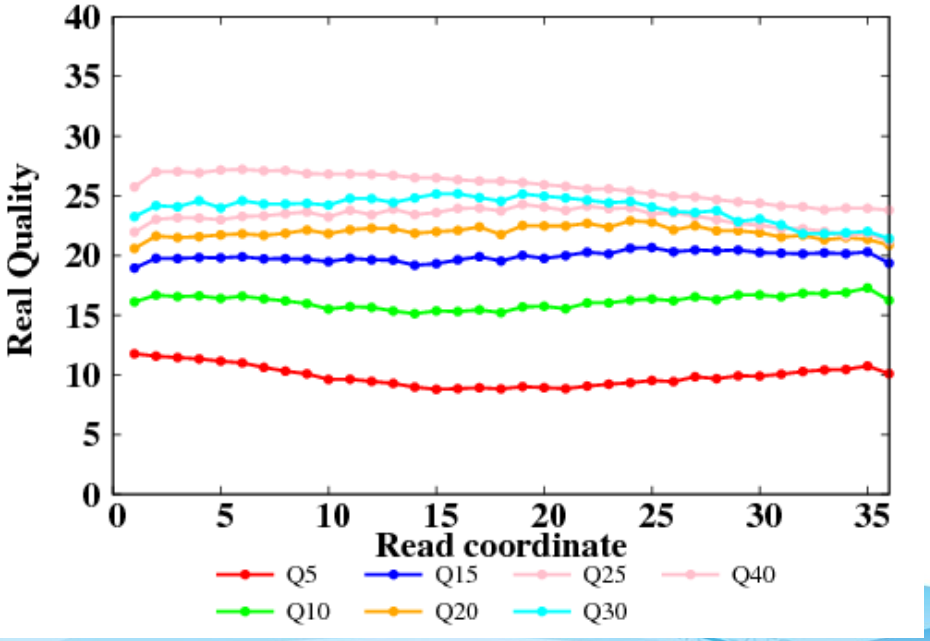
SOAPsnp


Sequencing reads
Map reads onto reference genome
Recalibrate sequencing quality score
Calculate likelihood of each genotype
Prior probability of each genotype
Inferred genotype via Bayes' theorem


Reference: G
10 Possible genotypes for diploid
AA AC AT AG CC CT TT CG TG GG
Observed data (base, quality)
G24,G25,G18,G20,G25,A15,A18,A30,A25,T10
Possible called consensus
AA AC AT AG CC CT TT CG TG GG

Sequencing errors

- SNP identification can be inferred by counting mismatch numbers.
- But, sequencing quality is important for distinguishing sequencing error from SNP, especially for Solexa sequencing.



Statistic model

$$P(T_i|D) = \frac{P(T_i)P(D|T_i)}{\sum_{x=1}^{S} P(T_x)P(D|T_x)}$$

- D: is the observed data in alignment.
- Prior(g): prior probability of a given genotypes
- P($D|x$): conditional probability to get the observed data D of a given genotype

Table 1. Prior probability of genotypes of a diploid genome

	A	C	G	T
A	3.33×10^{-4}	1.11×10^{-7}	6.67×10^{-4}	1.11×10^{-7}
C		8.33×10^{-5}	1.67×10^{-4}	2.78×10^{-8}
G			0.9985	1.67×10^{-4}
T				8.33×10^{-5}

Assuming that the reference allele is G, the homozygous SNP rate is 0.0005, the heterozygous SNP rate is 0.001, and the ratio of transitions versus transversions is 4.

Diploid Ti contain 10 types:
AA,CC,GG,TT,AC,AG,AT,CG,CT,GT;

P((Base,quality)|Ti) =
P((Base,quality)|Ti1) /2
+ P((Base,quality)|Ti2)
/2

P(D|Ti) = ∏P((Base, quality)|Ti)

Filtering SNPs

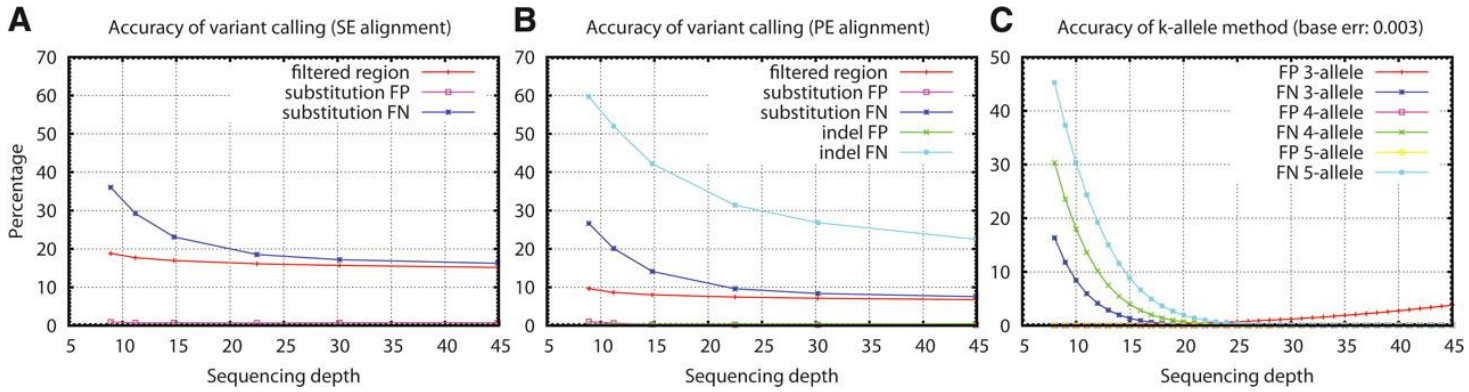
- By comparing the reference allele and the consensus genotype with maximum Bayesian likelihood, decisions of SNP status are made.

- But this is only the candidate SNPs, the accuracy is not reliable, especially when the mapped reads depth is low.

- In addition, we used some other measures to get confident SNPs, such as the minimum supporting reads number for each allele, exclude SNP predictions on repeat regions, and the rank-sum test for heterozygous SNP.

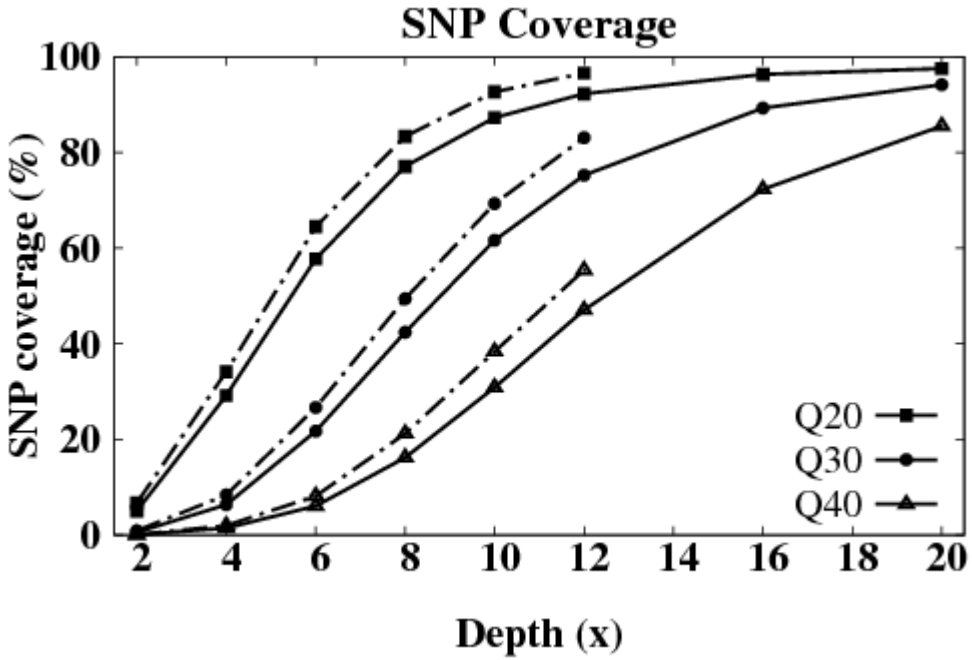
Other software to identify SNPs

- MAQ provides modules to identify SNPs. Error rates were calculated at each position thus SNPs were identified.



- Samtools, using sam/bam files, identifies SNPs in individual or populations.

Detect SNPs at different depth



In population studies, sequencing depth of each individual is always low. Then, how can we detect SNPs?

Detect SNPs in population

- Sequencing in population
  - Several individuals sequenced
  - Sequencing depth of each individual is relatively low (0.1-20X).
  - Total depth is high, several hundred times.

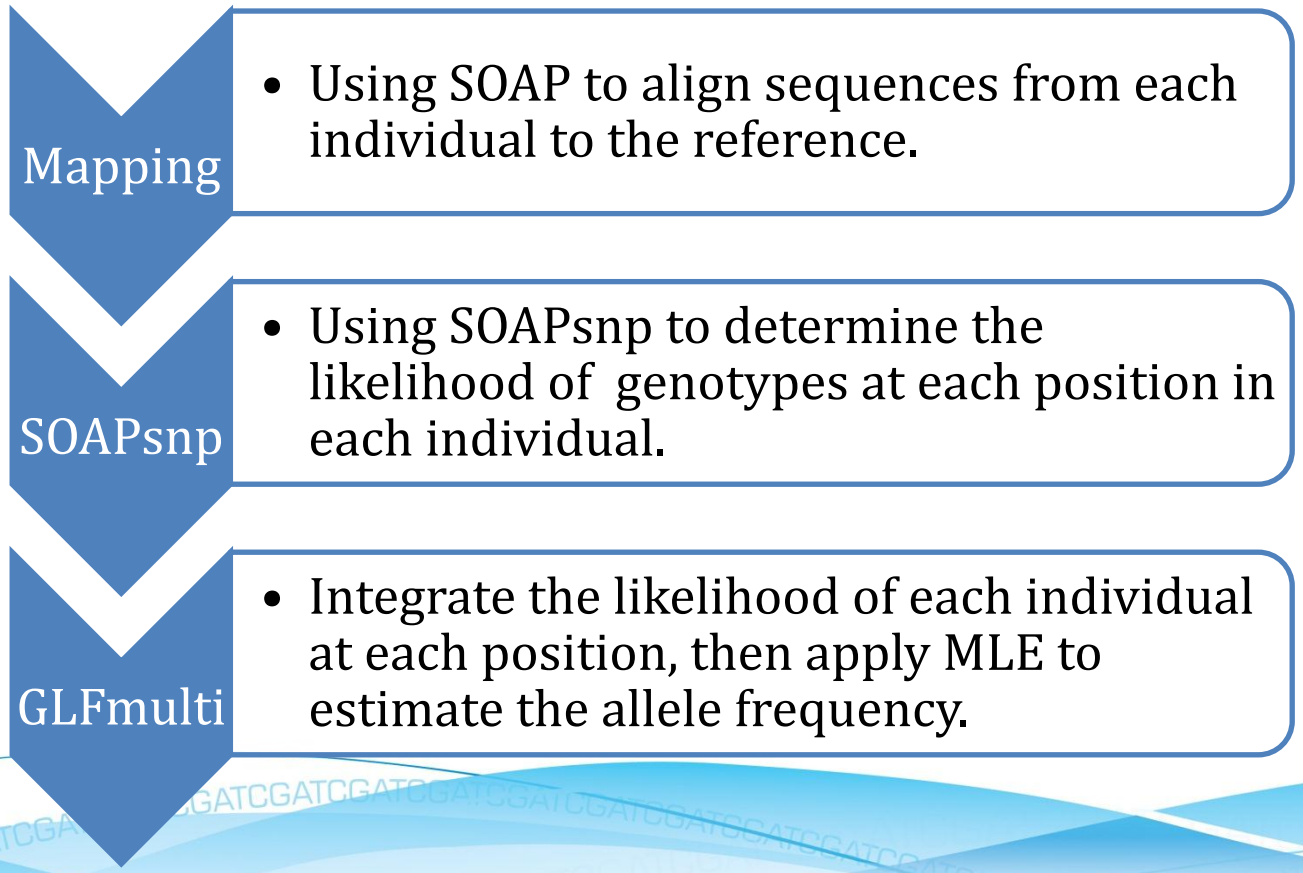
华大基因研究院
BGI research

Detect SNPs in population

Sequencing depth	Genome coverage ratio	Identified SNP ratio	Study purpose
Low (1-3X)	50%-80%	30%-50%	Rough population survey, infer population structure, phylogeny, and selection.
Middle (6-10X)	90%-99%	70%-90%	Whole population sequencing, suitable for further applications, such as molecular inbreeding, and functional genomics.
High (20-40X)	99.9%~100%	95%-99%	Complete map (de novo assembly) for each subspecies, line, or individual, suitable for all kinds of future applications.

GLFmulti

- To detect SNPs in population (applied in silkworm paper)

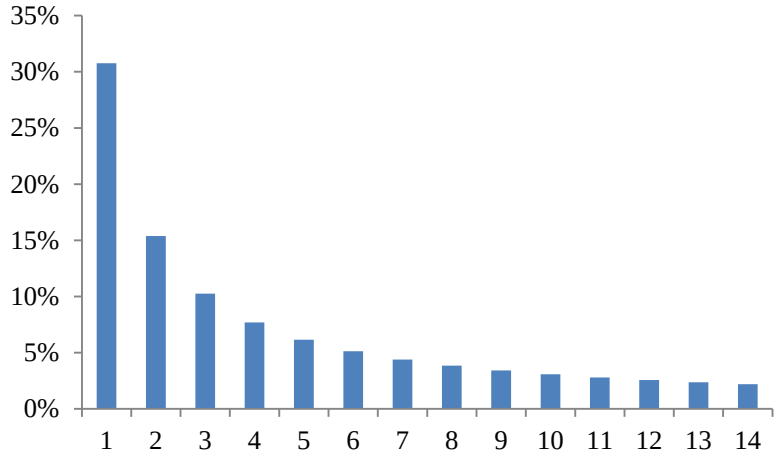


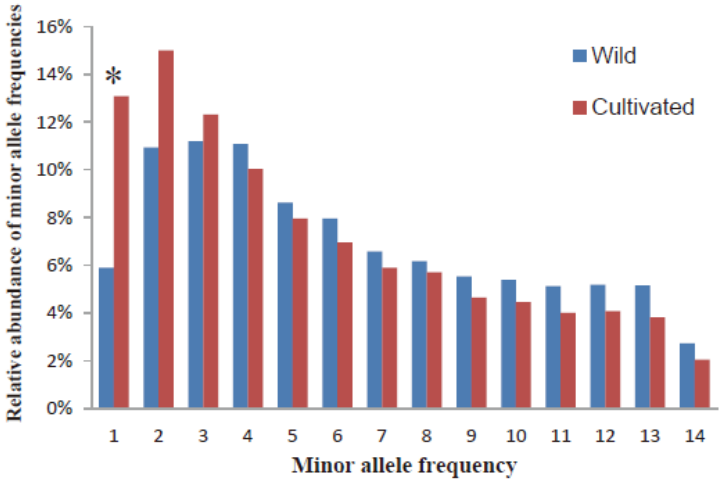
GLFmulti

- Frequency at each site with the maximum likelihood is given.
- Copy number, sequencing depth, quality score and minor allele count are integrated into one score.

GLFmulti

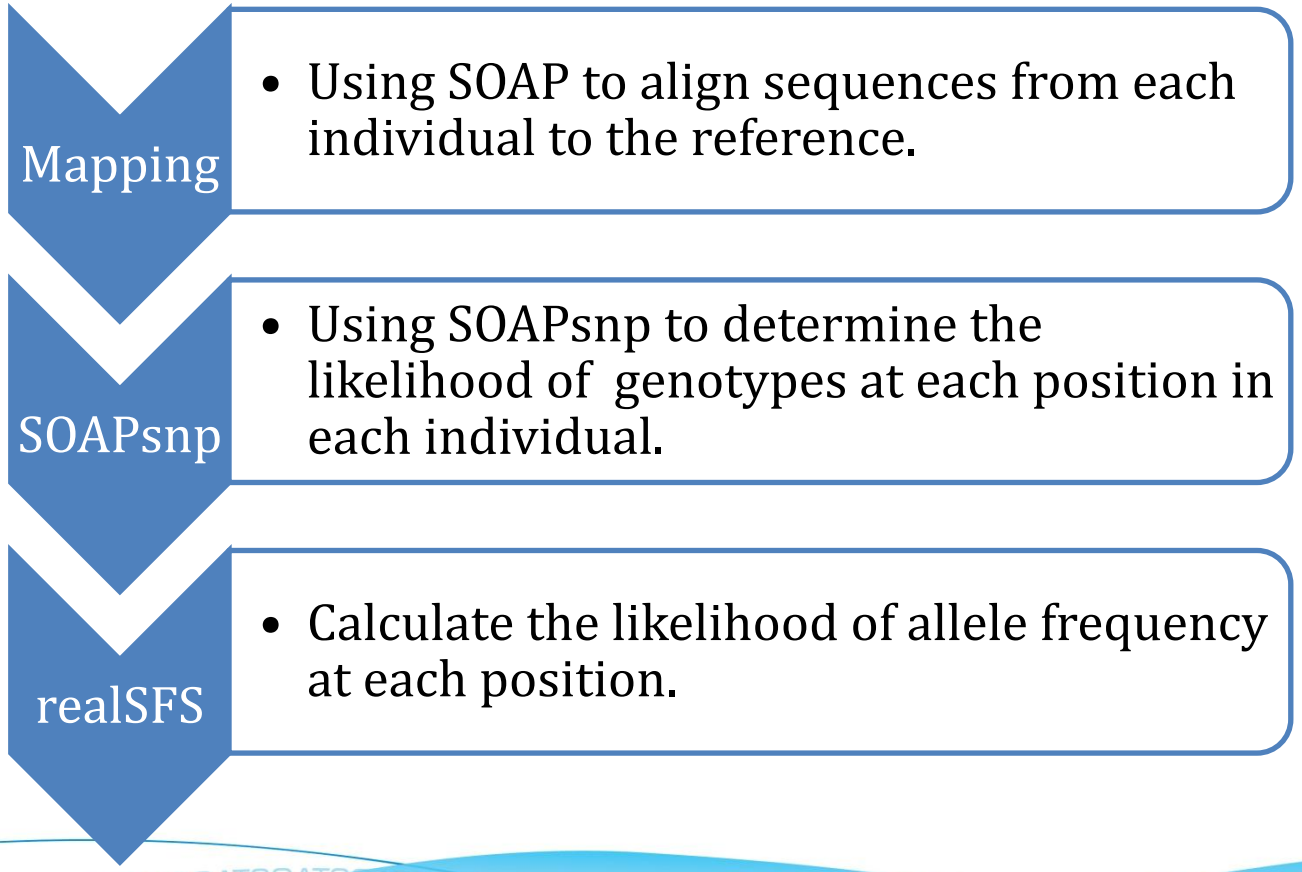
- SNPs were confidential, but SNPs at low frequency were underestimated.





realSFS

- To detect SNPs in population (applied in Tibetan paper)

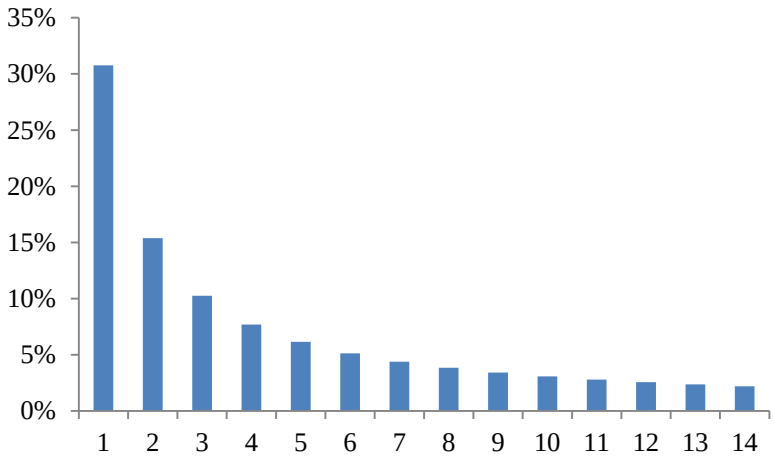


realSFS

1. Likelihood of different allele frequency.

	0	1	2	3	4	5	…
Individual1	P0	P1	P2	-	-	-	-
Individual2	P0*P0(2)	P1*P0(2)+P0*P1(2)	P2*P0(2)+P0*P2(0)+P1*P1(2)	P1*P2(2)+P2*P1(0)	P2*P2(2)	-	-
…	…	…	…	…	…	…	…

2. Prior probability of different allele frequency.



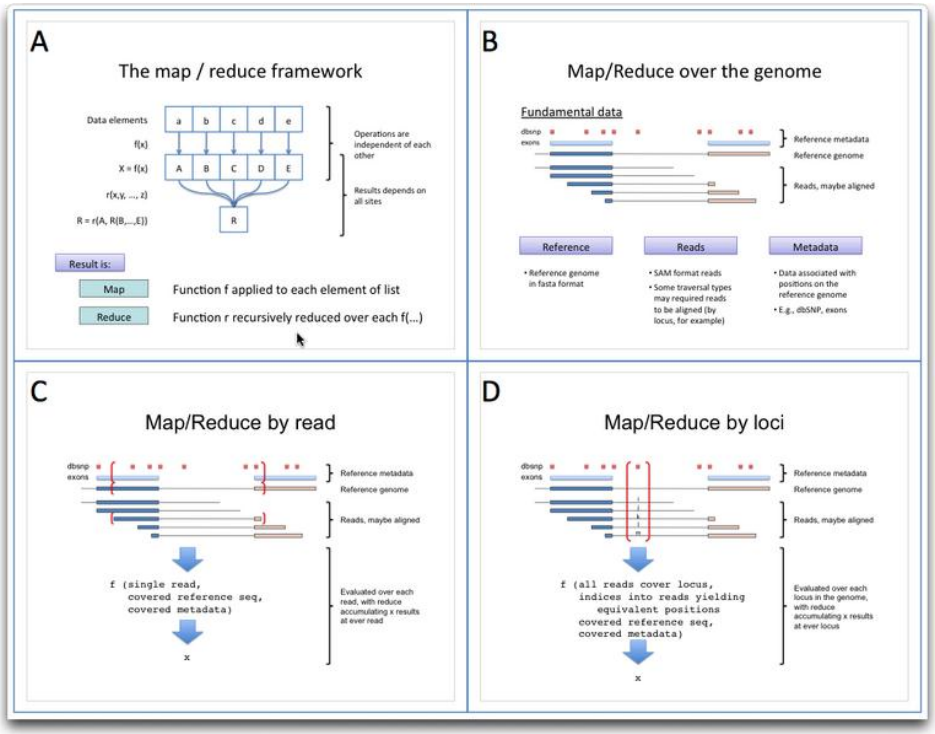
realSFS

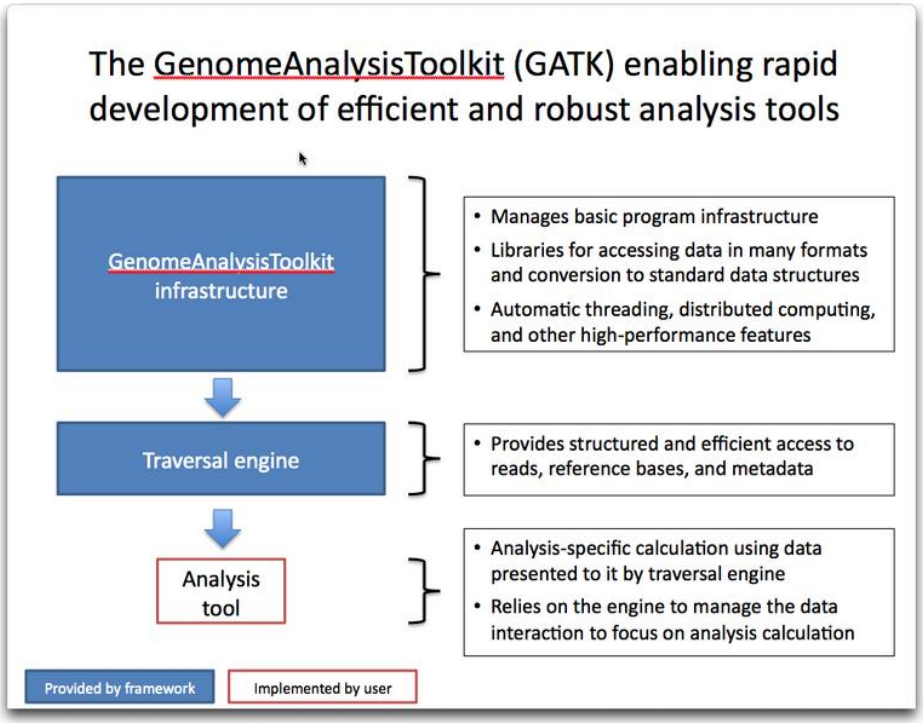
- Detect SNPs at low depth of each individual (lowest 0.4X depth).
- Relatively higher false positive ratio.

Other software to detect SNPs

- GATK: a widely used software/pipeline to detect variations, especially for human population (http://www.broadinstitute.org/gsa/wiki/index.php/Main_Page)
- SHORE: a pipeline used in 1000 genome project of *Arabidopsis* (http://sourceforge.net/apps/mediawiki/shore/index.php?title=SHORE_Documentation)

GATK


A
The map / reduce framework
Data elements
f(x)
X = f(x)
r(x,y, ..., z)
R = r(A, R(B,...,E))
Operations are independent of each other
Results depends on all sites
Result is:
Map
Function f applied to each element of list
Reduce
Function r recursively reduced over each f(...)
B
Map/Reduce over the genome
Fundamental data
dbsnp
exons
Reference metadata
Reference genome
Reads, maybe aligned
Reference
Reads
Metadata
• Reference genome in fasta format
• SAM format reads
• Some traversal types may required reads to be aligned (by locus, for example)
• Data associated with positions on the reference genome
• E.g., dbSNP, exons
C
Map/Reduce by read
f (single read,
covered reference seq,
covered metadata)
Evaluated over each read, with reduce accumulating x results at ever read
x
D
Map/Reduce by loci
f (all reads cover locus,
indices into reads yielding
equivalent positions
covered reference seq,
covered metadata)
Evaluated over each locus in the genome, with reduce accumulating x results at ever locus


The GenomeAnalysisToolkit (GATK) enabling rapid development of efficient and robust analysis tools
GenomeAnalysisToolkit infrastructure
• Manages basic program infrastructure
• Libraries for accessing data in many formats and conversion to standard data structures
• Automatic threading, distributed computing, and other high-performance features
Traversal engine
• Provides structured and efficient access to reads, reference bases, and metadata
Analysis tool
• Analysis-specific calculation using data presented to it by traversal engine
• Relies on the engine to manage the data interaction to focus on analysis calculation
Provided by framework
Implemented by user

SHORE

- SHORE is a data analysis and management application for short DNA/RNA reads produced by the various contemporary sequencing platforms.
- SHORE is designed to support different sequencing applications including genomic re-sequencing, ChIP-Seq, mRNA-Seq, sRNA-Seq and BS-seq.
- SHORE was developed for applications in *Arabidopsis thaliana* but has been successfully used with other genomes, including human, mouse, *D. melanogaster*, *C. elegans*, maize and several bacterial genomes.

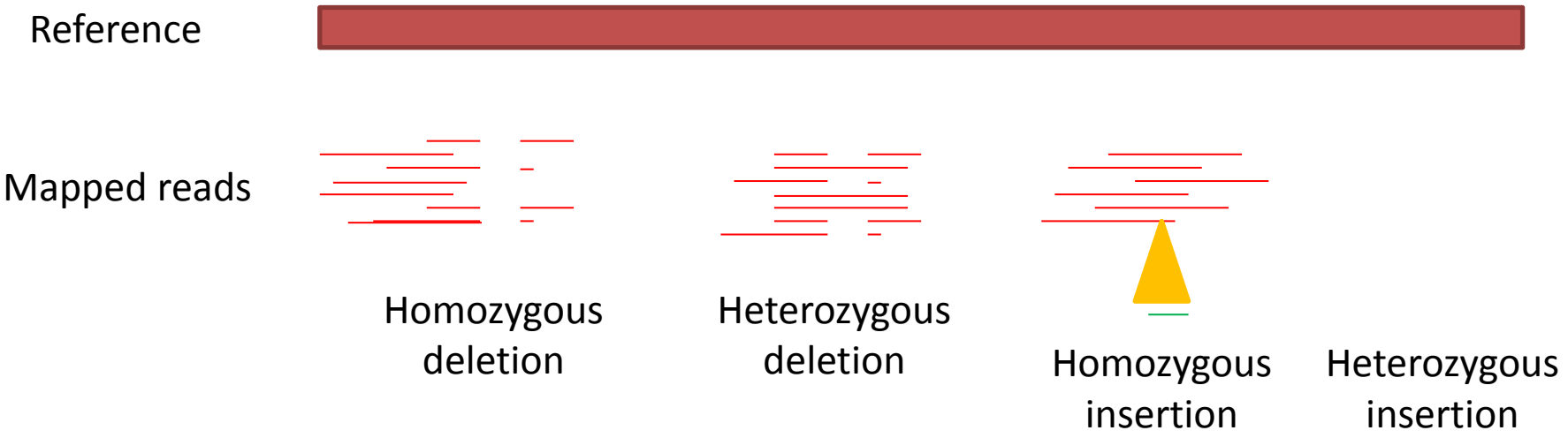
Summary of SNP detection

- Different methods can be applied in SNP detection in single individual.
- The main problem to cope with is the sequencing errors which would result in false positive.
- The variation calling would also depend a lot on the mapping result.
- Experimental validation is necessary.
- Detect SNPs in individual require higher depth.
- Detect SNPs in population can detect SNPs at lower individual depth.
- Statistic method is usually applied in SNP calling to prevent influence of sequencing errors.
- Different statistic models show different detection power.

Detection of indels

- After mapping, if the mapping permits gaps, those alignments with gaps can be the candidate for indels.
- In SOAPindel sequencing quality and mapping result were combined to deduce the probability of being an indel.

Detection of indel by SOAPindel



Other methods to detect indels

- Dindel: program for calling small indels from short-read sequence data
  - Extracts all indels from the read-alignments in the BAM file
  - Candidate InDels grouped into windows
  - For each window, Dindel will generate candidate haplotypes from the candidate indels and realign
  - Interpreting the output from Dindel

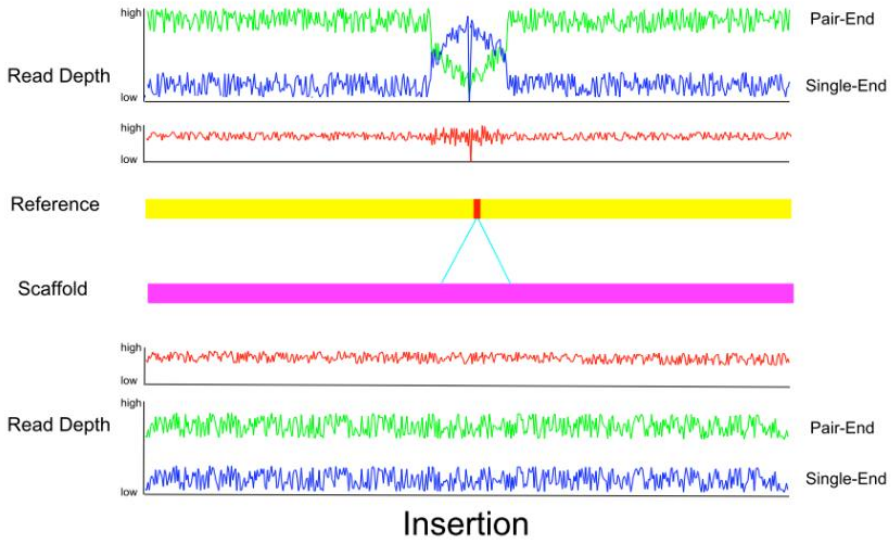
Detection of SVs

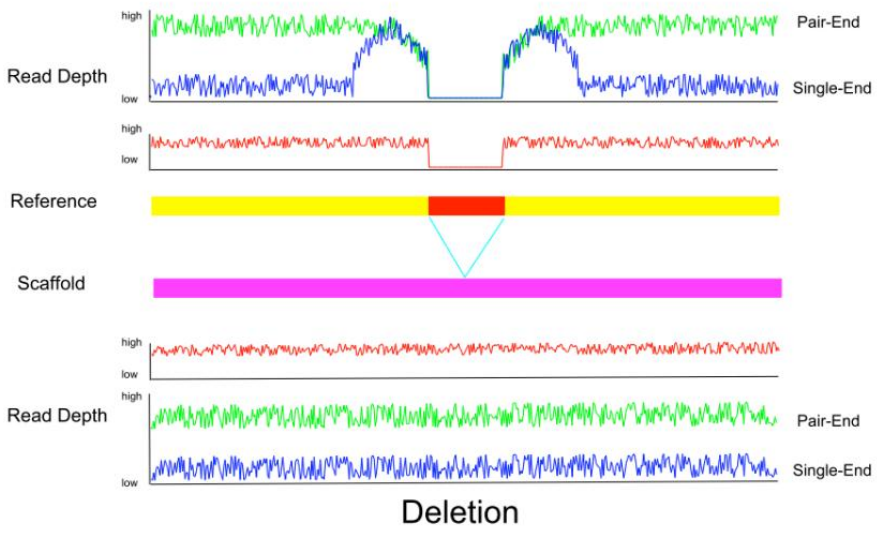
- Mainly there are three kinds of methods to detect structural variations:
  - Local *de novo* assembly
  - Pair end mapping
  - Split reads
- The SV detection based on assembly is believed to be more accurate.

华大基因研究院
BGI research

Local *de novo* assembly

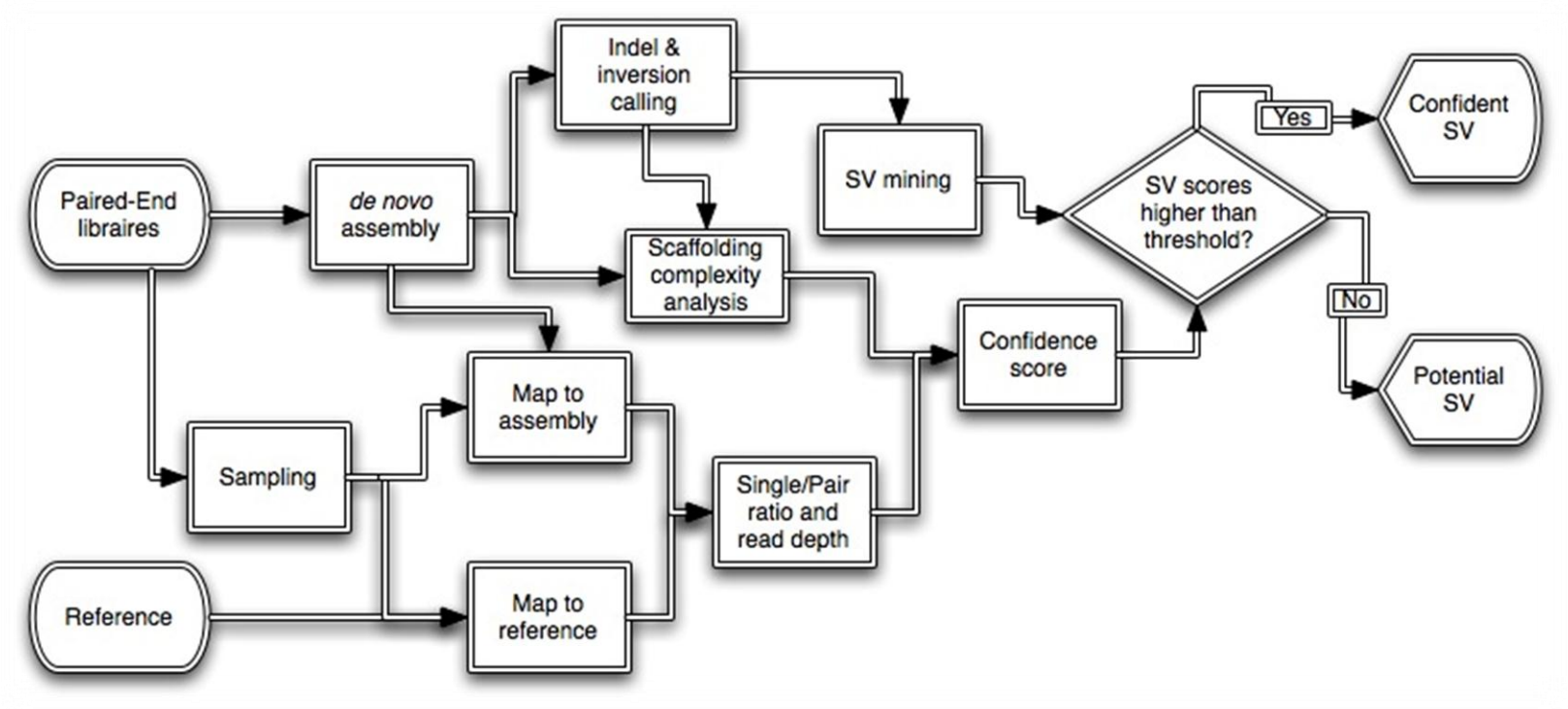
1. Assemble pair-end reads into scaffold.
2. Align the scaffold to reference genome.



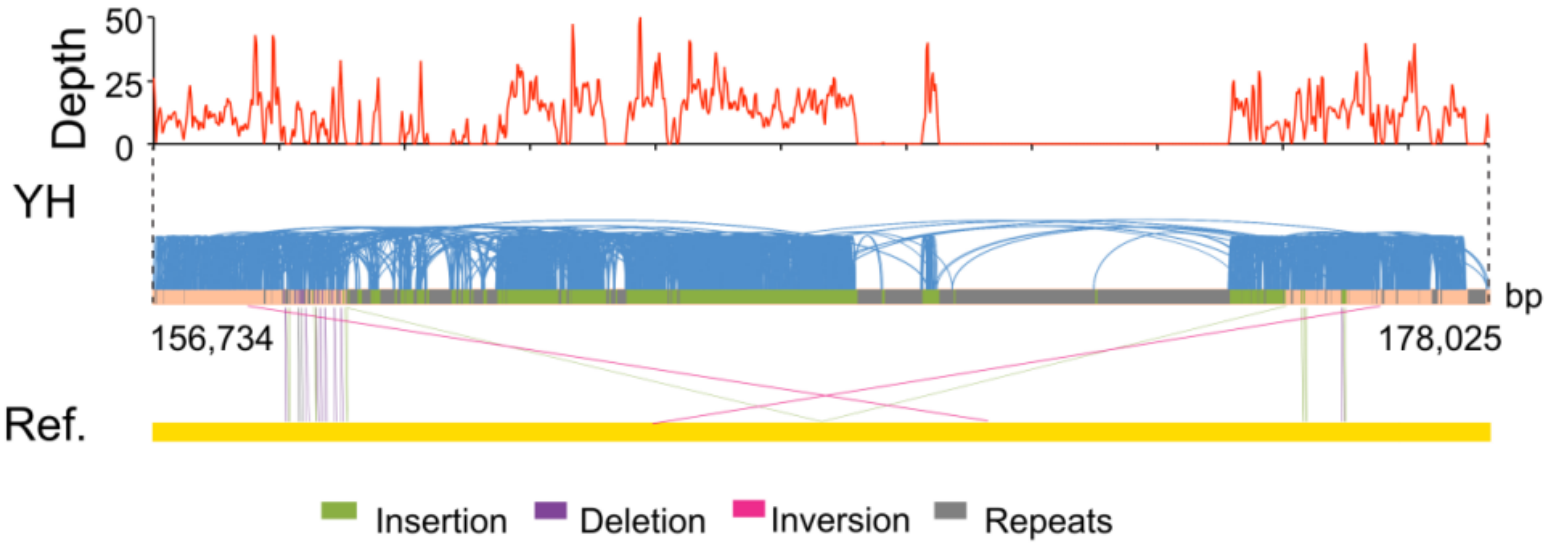


SOAPsv

Workflow of SOAPsv:



Complex structure variants can be detected:



Output format:

```
Insertion       scaffold41829   231     573     chr01   4023036 4023036 342 CAAACTATTCTTAATTAATAGATAAACTGCCATGCCGCAT
Insertion       scaffold1607    1979    2129    chr03   20248959        20248959        150 CTAGAACCCCCGCGGGGGCAGACC
Deletion        scaffold28683   35      35      chr04   33346626        33346656        30      GTTAAGGAAATGATTTATTG
Deletion        scaffold39065   797     797     chr06   14536368        14536526        158     TGAACCAGAGCTTGCACATC
```

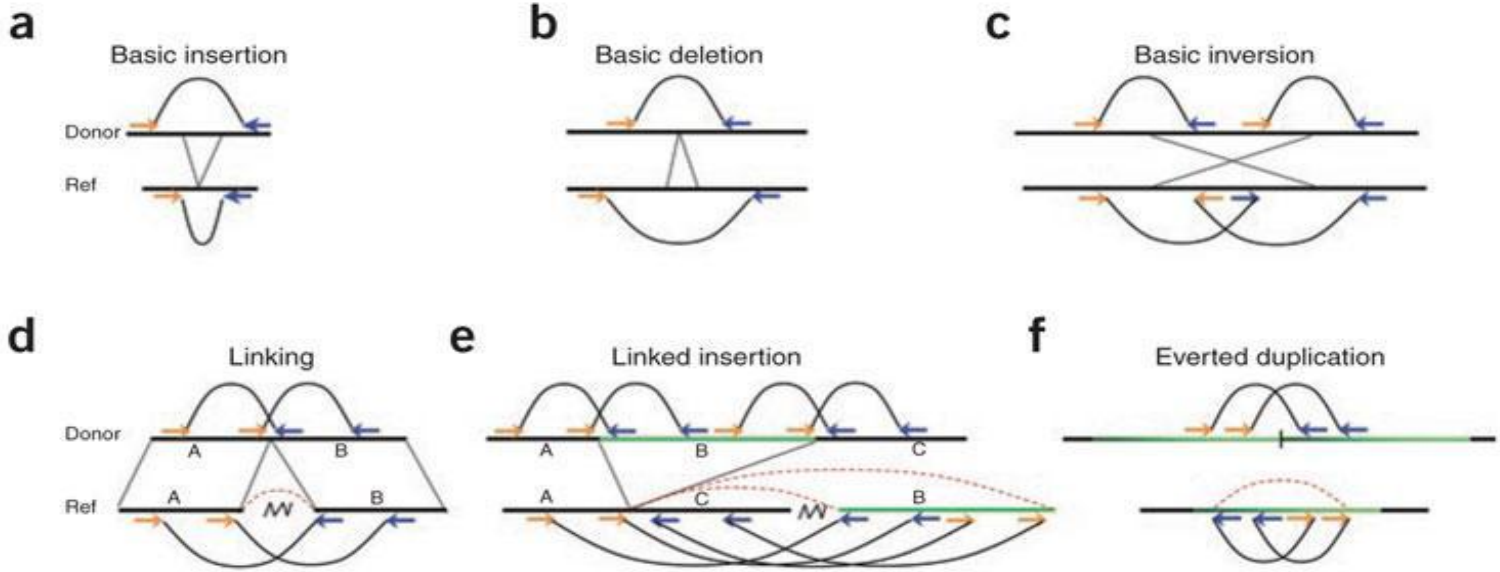
(#Type, scaffold, start, end, refChr, start, end, length, sequence)

Reference:

Li, Y., Zheng, H., Luo, R., Wu, H., Zhu, H., Li, R., ... & Wang, J. (2011). Structural variation in two human genomes mapped at single-nucleotide resolution by whole genome de novo assembly. *Nature biotechnology*, *29*(8), 723-730.

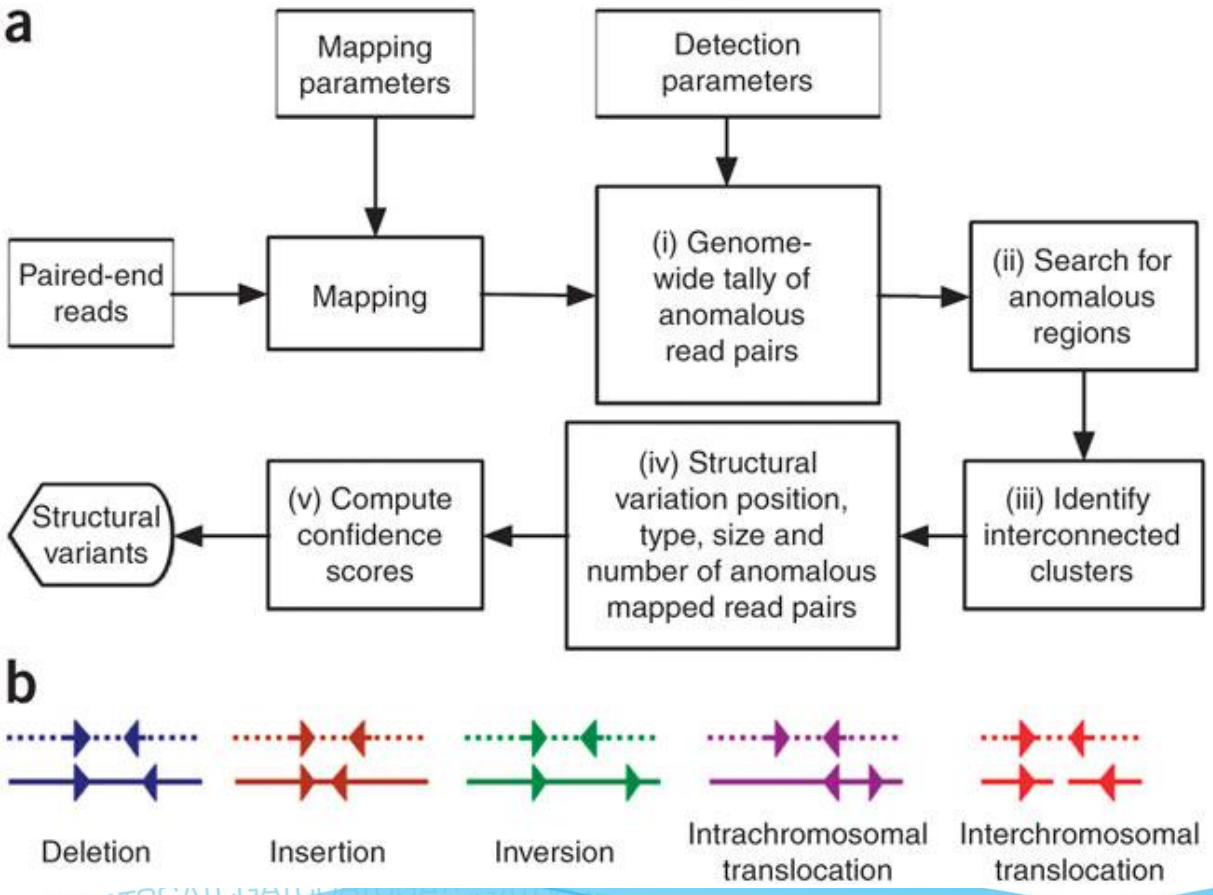
Pair end mapping

1. Map the reads to the reference genome.
2. Detect structure variants by discordant reads.

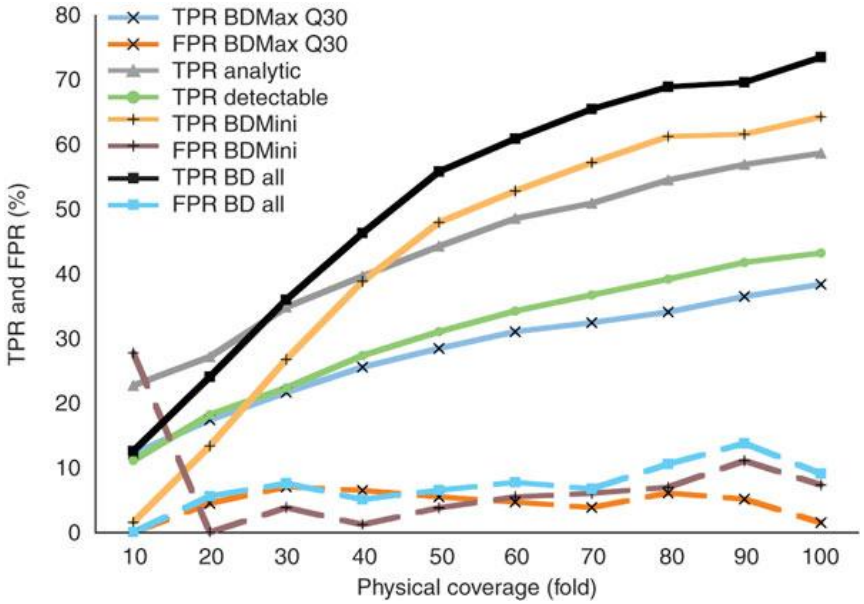


BreakDancer

Workflow of BreakDancer:



Performance of BreakDancer



Output format:

```
1 62767 10+0- 1 63126 0+10- INS -13 36 10 NA|10 1.00 BreakDancerMini-0.0.1 q10
1 59257 5+1- 1 60164 0+5- DEL 862 99 5 nA|2:tB|1 0.56 BreakDancerMax-0.0.1 c4
1 10000 10+0- 2 20000 7+10- CTX -296 99 10 tB|10 1.00 BreakDancerMax-0.0.1 t1
```

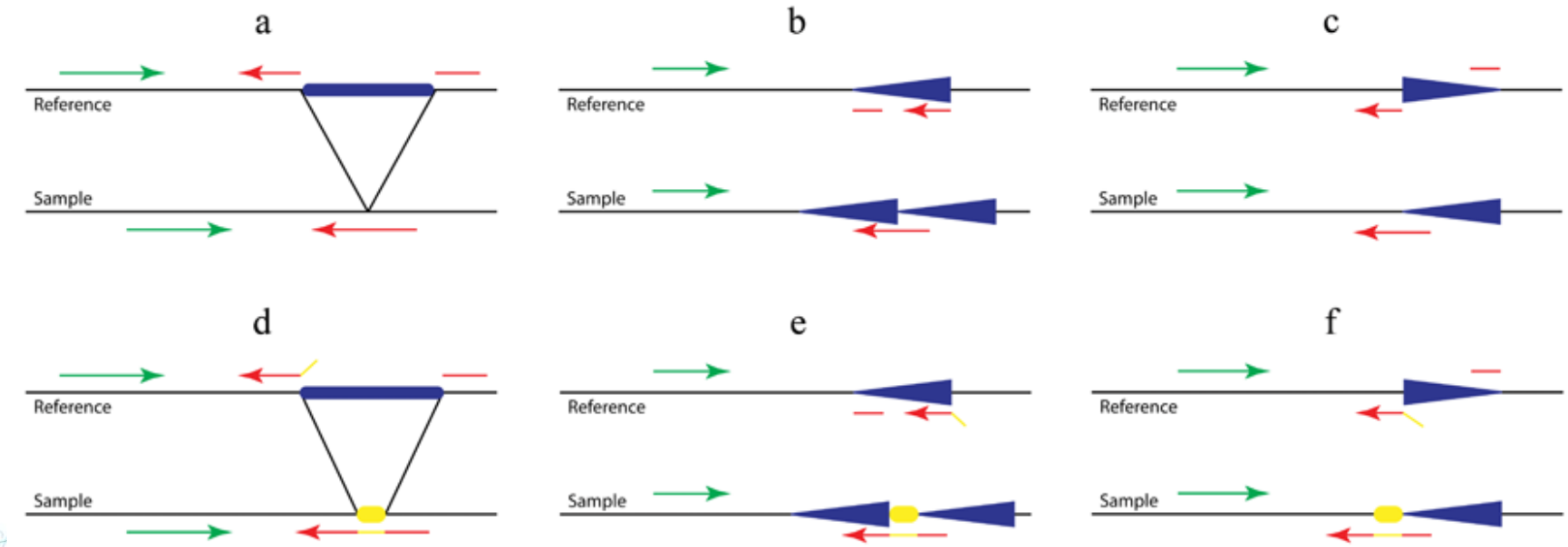
(#chromosome, position, orientation, chromosome, position, orientation, type, length, score …)

Reference:

Chen, K., Wallis, J. W., McLellan, M. D., Larson, D. E., Kalicki, J. M., Pohl, C. S., ... & Mardis, E. R. (2009). BreakDancer: an algorithm for high-resolution mapping of genomic structural variation. Nature methods, 6(9), 677-681.

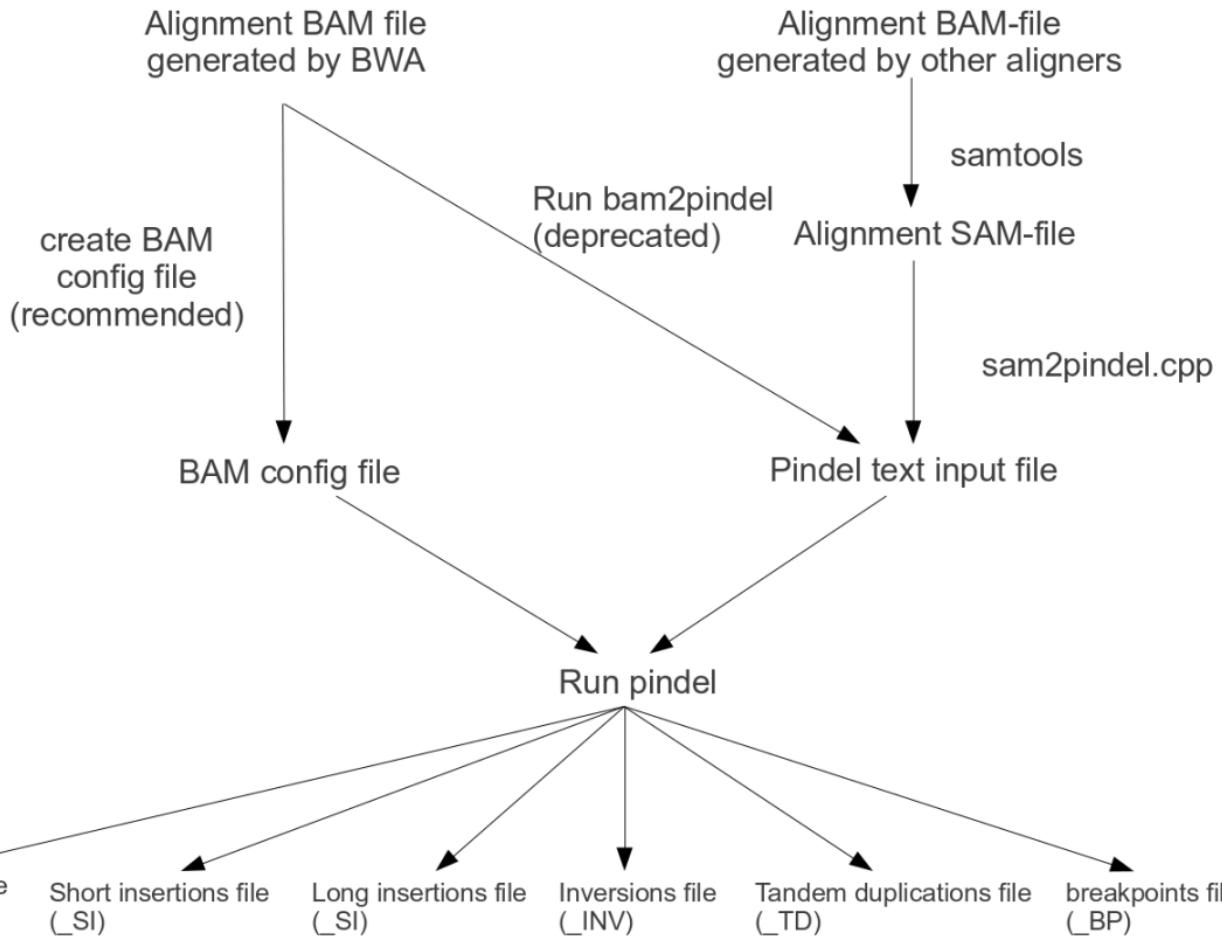
Split reads

1. Map the reads to reference genome.
2. Select those paired reads that mapped with indels or of which only one end can be mapped.
3. Uses the mapped reads to determine the anchor point on the reference genome and the direction of the unmapped

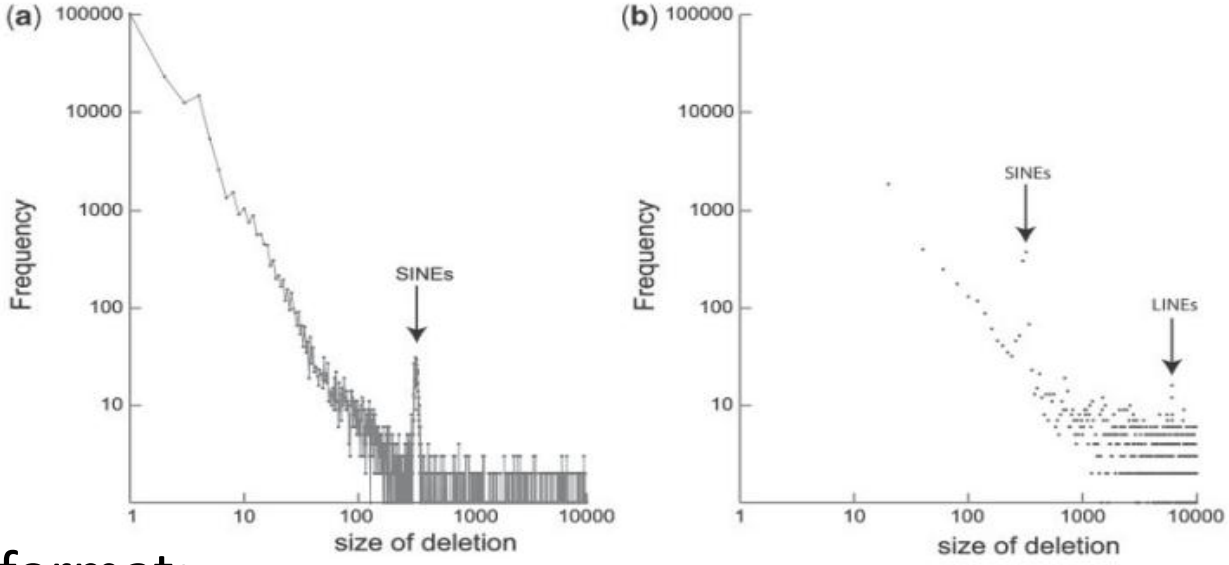


Pindel

Workflow of Pindel:



SLINEs and LINEs can be detected by Pindel:



Output format:

```
##################################################################################################################
9       D 43    NT 0 "" ChrID chr01     BP 26563        26607   BP_range 26563  26607   Supports 9      9       + 9
AGGTCATCGTAGATGCCATCATCAACAGGTACCACCGTCCAATTCCCAATTCCTTTCAGTTGCGCACTTCAATTGTCCAATTCACTTTTTTTtcaat<33>ctgtcCAATCACCCCCACC
                                                                                TTCACTTTTTTT                CAATCACCCTCACC
                                                                              TCCAATTCACTTTTTTT              CAATCACCCTCACC
                                                                            TCAATTGTCCAATTCACTTTTTTT         CAATCACCCTCACC
                                                                          TTGCGCACTTCAATTGTCCAATTCACTTTTTTT   CAATCACCCTCACC
                                                                        CCTTTCAGTTGCGCACTTCAATTGTCCAATTCACTTTTTTT CAATCACCCTCACC
                                                                      TTCCTTTCAGTTGCGCACTTCAATTGTCCAATTCACTTTTTTT CAATCACCCTCACC
```

(#index, type, length, insertion length, chrID, border of event, range of unclear breakpoint, support number)

Reference:

Ye, K., Schulz, M. H., Long, Q., Apweiler, R., & Ning, Z. (2009). Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads. *Bioinformatics*,*25*(21), 2865-2871.

华大基因研究院
BGI research

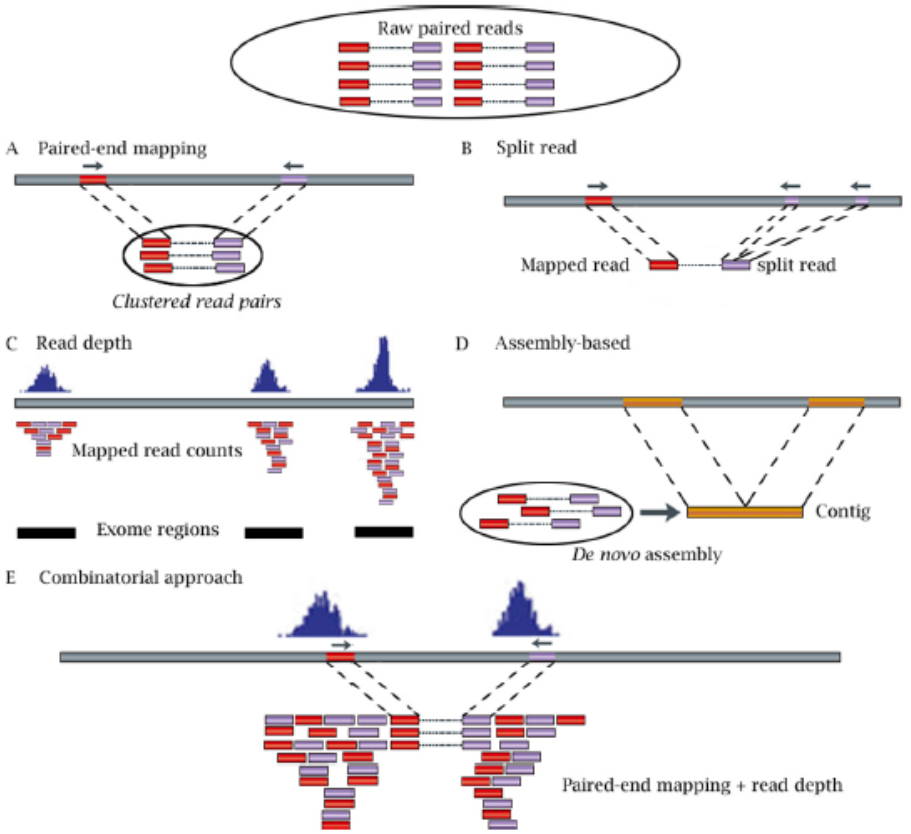
Comparison

Tools	SOAPsv	BreakDancer	Pindel
Main detectable length range	1 bp-50 kbp	>10bp	1bp-30kbp
Detectable SV types			
Insertions	Yes	Yes	Yes
Deletions	Yes	Yes	Yes
Inversions	Yes	Yes	Yes
Complex	Yes	Yes	No
Precision of breakpoints	Single base	A short ambiguous range	Single base
Genotypes of SV events	Yes	No	Yes
False-positive rate in simulated data	1.20%	9.1–10.3%	<2%
False-negative rate in simulated data	9.60%	26–32%	~20%

*Reference: Li, Y., Zheng, H., Luo, R., Wu, H., Zhu, H., Li, R., ... & Wang, J. (2011). Structural variation in two human genomes mapped at single-nucleotide resolution by whole genome de novo assembly. *Nature biotechnology*, *29*(8), 723-730. (Table2)

The CNV detection

- Depth of Coverage (DOC):
  - Number of reads in a region
    - Uniform depth distribution
    - Biased for GC etc.
- Paired End Mapping (PEM):
  - Proper pairing when mapping
    - Limit by insert sizes
    - Inversions/Translocations
- Split Reads (SR)
  - The unaligned reads
    - Pinpoint the location of CNV
- Assembly based (AS)



Zhao et al. BMC Bioinformatics 2013, 14(Suppl 11):S1

华大基因研究院 BGI research

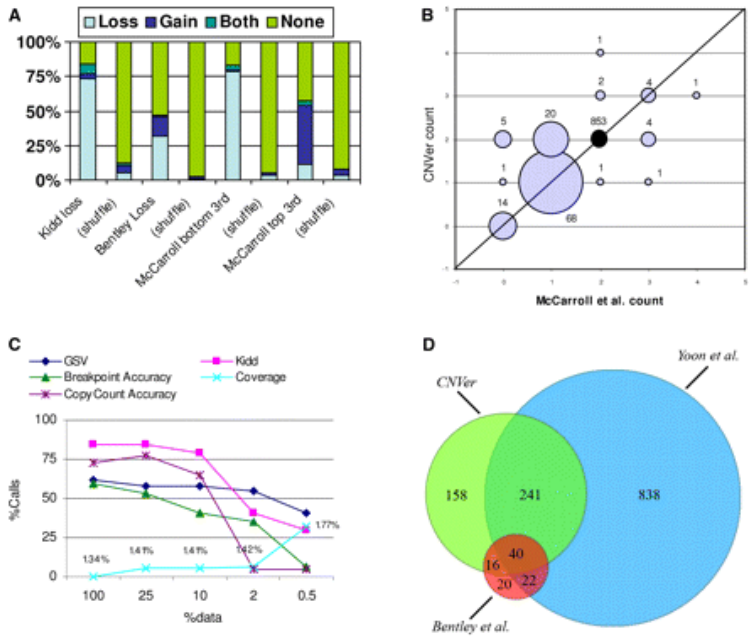
Summary of CNV detect tools

Method	URL	Language	Input	Comments
PEM-based				
BreakDancer	http://breakdancer.sourceforge.net/	Perl, C++	Alignment files	Predicting insertions, deletions, inversions, inter- and intra-chromosomal translocations
PEMer	http://sv.gersteinlab.org/pemer/	Perl, Python	FASTA	Using simulation-based error models to call SVs
VariationHunter	http://compbio.cs.sfu.ca/strvar.htm	C	DIVET[a]	Detecting insertions, deletions and inversions
commonLAW	http://compbio.cs.sfu.ca/strvar.htm	C++	Alignment files	Aligning multiple samples simultaneously to gain accurate SVs using maximum parsimony model
GASV	http://code.google.com/p/gasv/	Java	BAM	A geometric approach for classification and comparison of structural variants
Spanner	N/A	N/A	N/A	Using PEM to detect tandem duplications
SR-based				
AGE	http://sv.gersteinlab.org/age	C++	FASTA	A dynamic-programming algorithm using optimal alignments with gap excision to detect breakpoints
Pindel	http://www.ebi.ac.uk/~kye/pindel/	C++	BAM /FASTQ	Using a pattern growth approach to identify breakpoints of various SVs
SLOPE	http://www-genepi.med.utah.edu/suppl/SLOPE	C++	SAM/FASTQ/MAQ[b]	Locating SVs from targeted sequencing data
SRiC	N/A	N/A	BLAT output	CalibratingSV calling using realistic error models
AS-based				
Magnolya	http://sourceforge.net/projects/magnolya/	Python	FASTA	Calling CNV from co-assembled genomes and estimating copy number with Poisson mixture model
Cortex assembler	http://cortexassembler.sourceforge.net/	C	FASTQ/FASTA	Using alignment of *de novo* assembled genome to build de Bruijn graph to detect SVs
TIGRA-SV	http://gmt.genome.wustl.edu/tigra-sv/	C	SV calls[c] + BAM	Local assembly of SVs using the iterative graph routing assembly (TIGRA) algorithm

Method	URL	Language	Input	Combination
NovelSeq	http://compbio.cs.sfu.ca/strvar.htm	C	FASTA/SAM	PEM+AS
HYDRA	http://code.google.com/p/hydra-sv/	Python	Discordant paired-end mappings	PEM+AS
CNVer	http://compbio.cs.toronto.edu/CNVer/	Perl, C++	BAM/aligned positions	PEM+RD
GASVPro	http://code.google.com/p/gasv/	C++	BAM	PEM+RD
Genome STRiP	http://www.broadinstitute.org/software/genomestrip/genome-strip	Java, R	BAM	PEM+RD
SVDetect	http://svdetect.sourceforge.net/	Perl	SAM/BAM/ELAND	PEM+RD
inGAP-sv	http://ingap.sourceforge.net/	Java	SAM	PEM+RD
SVseq	http://www.engr.uconn.edu/~jiz08001/svseq.html	C	FASTQ/BAM	PEM+SR
Nord et al.	N/A	N/A	N/A	RD+SR

Software to detect CNV

- CNVer http://compbio.cs.toronto.edu/CNVer/



Notes about CNV detection

- Repeat regions would have great impact.
- Mapping depth should be carefully inspected especially when the pair-end mapping was done.
- It is always difficult to give the actual copy numbers.

Summary of variation detection

- Different models are available in SNP calling of a single individual, and good methods should take care of both sequencing and mapping quality.
- Indels can be easily detected by interpreting the gapped alignment, and the accuracy do depend greatly on the mapping.
- Using assembly to find SVs is more accurate in practice than using pair-end information.
- Depth of coverage was applied in CNV detection.

Thanks!