The Abdus Salam
International Centre for Theoretical Physics

United Nations
Educational, Scientific and
Cultural Organization

IAEA
International Atomic Energy Agency

2255-2

2nd Conference on Systems Biology and New Sequencing Techniques"
(2-4 November)
preceded by Introductory Lectures on "Quantitative Approaches to Biological Problems"
(31 October - 1 November)

31 October - 4 November, 2011

FROM DNA SEQUENCING TO GENOMES: THE ASSEMBLY CHALLENGE

Giuseppe Narzisi, PhD

Cold Spring Harbor Laboratory - NY
USA

FROM DNA SEQUENCING TO GENOMES: THE ASSEMBLY CHALLENGE

Giuseppe Narzisi, PhD

Scientific Informatics Analyst

CSH Cold Spring Harbor Laboratory

Goals

- Understand issues and challenges of genome assembly
- State-of-the-art assemblers
- Theory vs. practice: dealing with real data
- Sequence Assemblers: black-box vs. white-box
- How to evaluate sequence assemblers
- Do not trust sequence assemblers!

- ***Think of this tutorial as a mini-course on sequence assembly.***

Outline

① **Introduction**

- From DNA Sequencing to genome sequences

② **DNA Sequence Assembly**

- Formulation and statistics

③ **Sequence Assembly Problem**

- Computational Complexity

④ **Assembly Paradigms**

- The art of solving a difficult puzzle

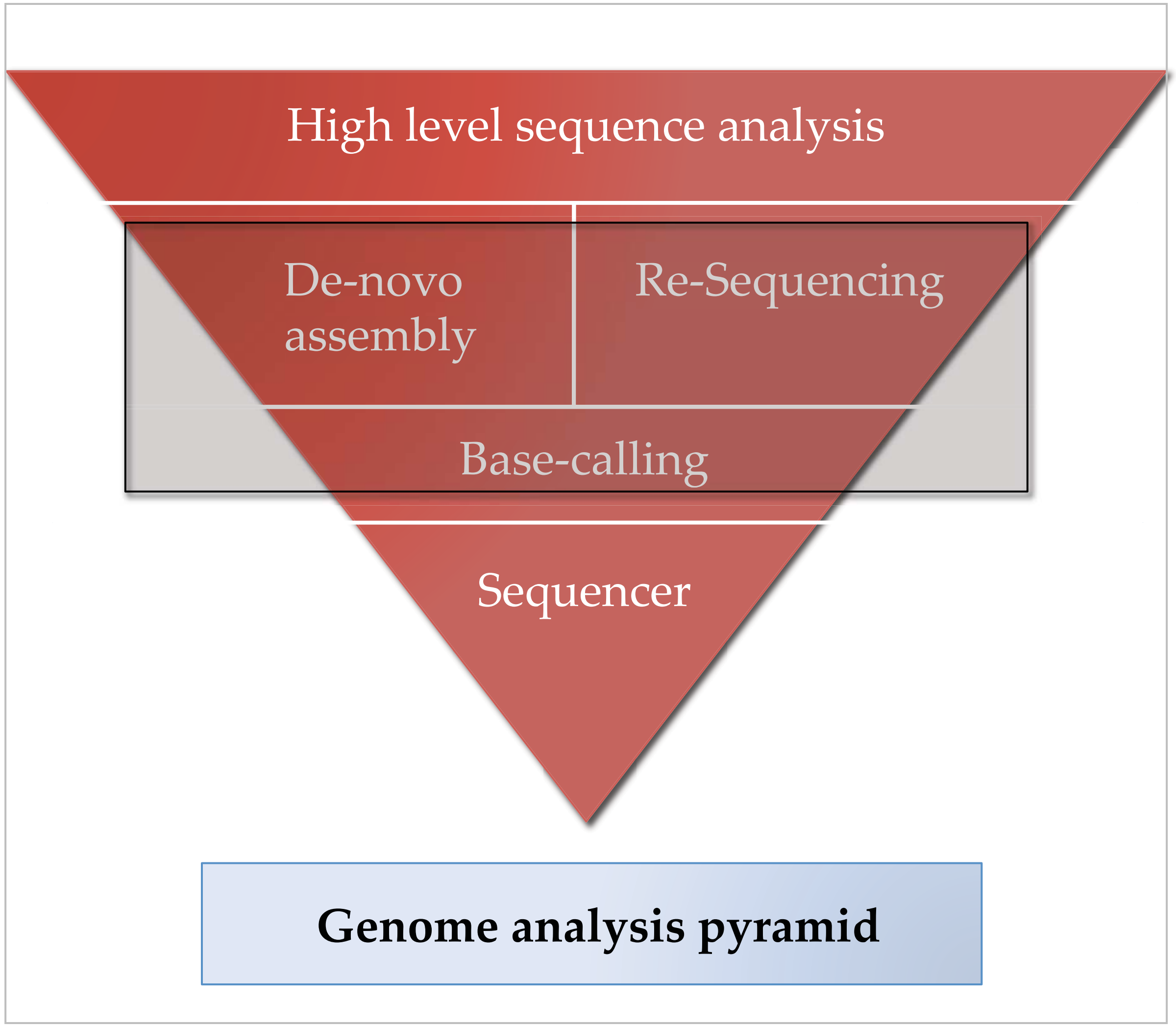
⑤ **Assembly Quality**

- How to evaluate an assembly

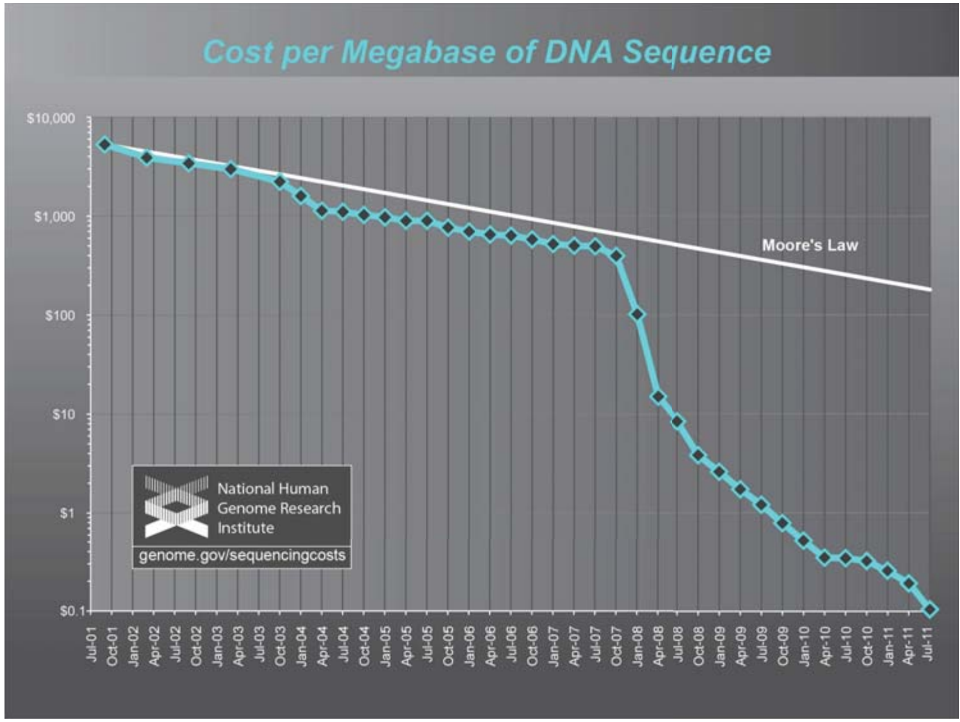
INTRODUCTION

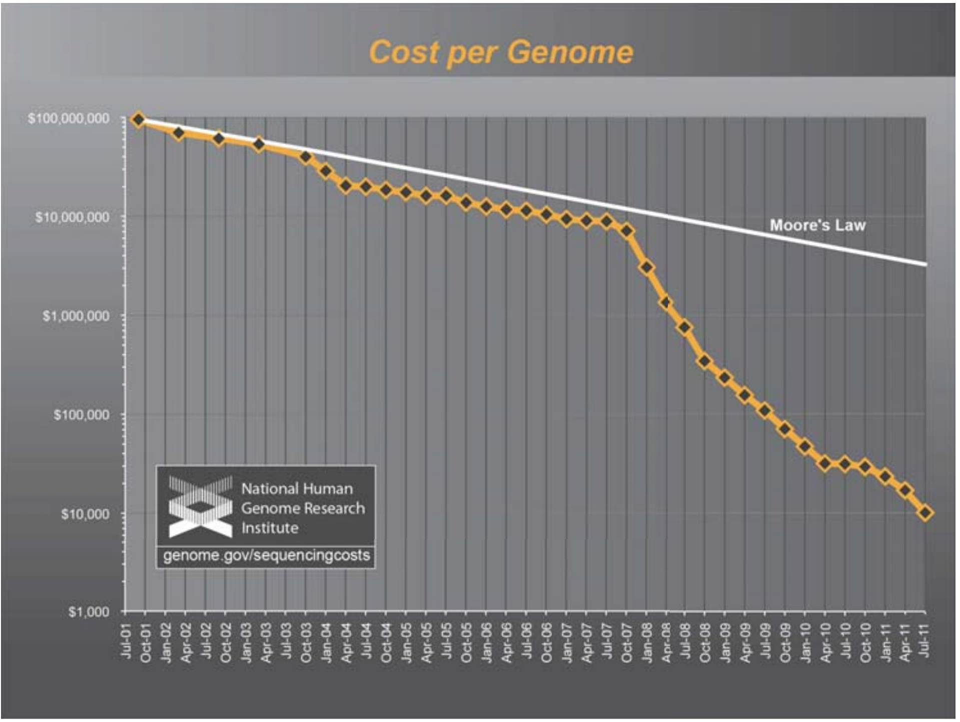
From DNA Sequencing to genome sequences

What is needed for clinical sequence analysis



DNA Sequencing costs



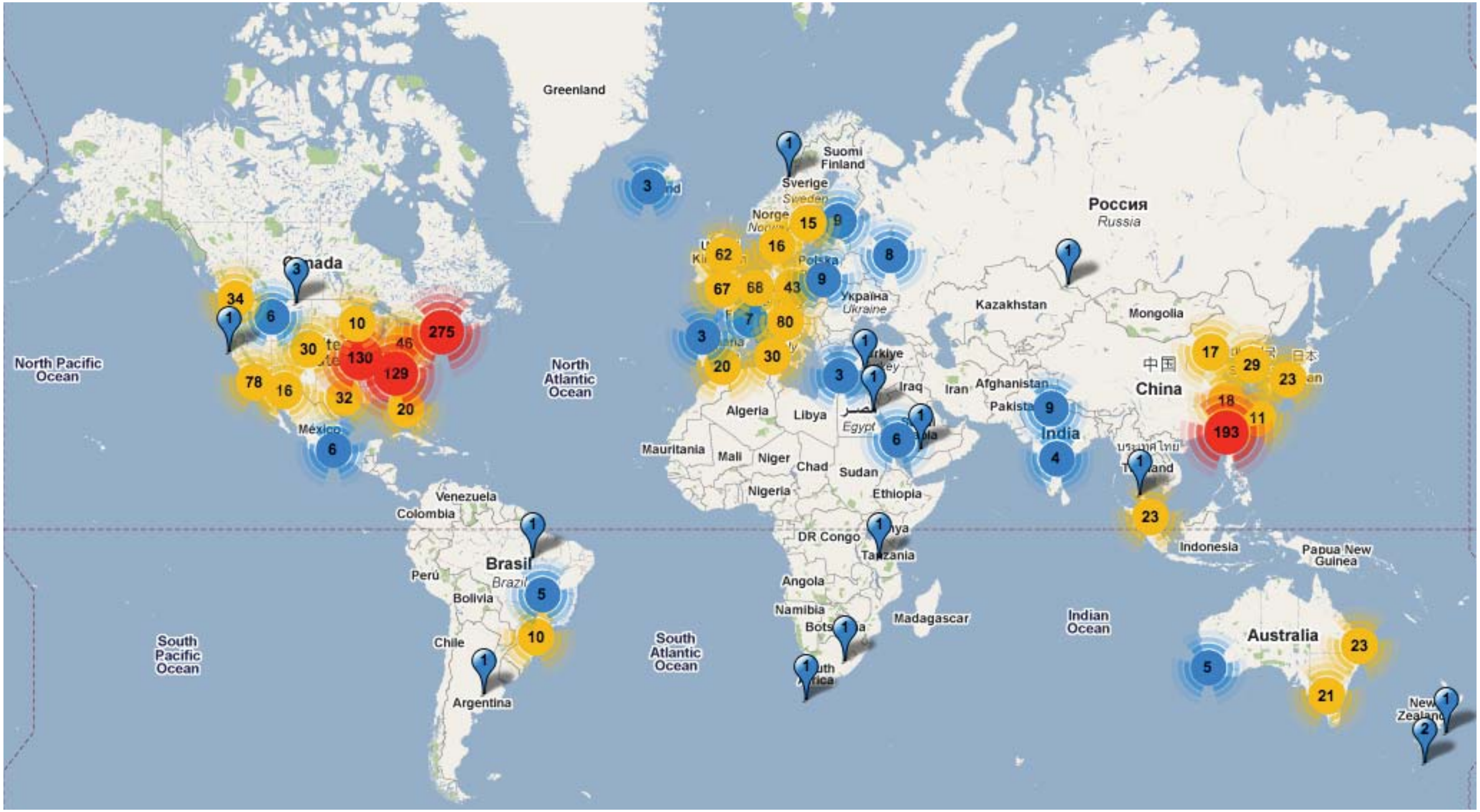


- Costs associated with DNA sequencing performed at sequencing centers funded by the **National Human Genome Research Institute** (NHGRI).

- *"Although sequencing technologies improve, the analysis of these data continues to lag far behind"*

[Kristensen, **Genome Biology** 2011]

Sequencing Centers



Next Generation Genomics: World Map of High-throughput Sequencers

http://pathogenomics.bham.ac.uk/hts/

The assembly challenge!

HiSeq 2000 from Illumina, Inc

Library Preparation
~2 h [15 min hands-on (Nextera)]
< 6 h [< 3 h hands-on (TruSeq)]

Cluster Generation
~5 h (<10 min hands-on)

Sequencing by Synthesis
~1.5 to 11 days

CASAVA
2 days (30 min hands-on)

200 GBp in 8 Days

≈ **50x coverage** of a human genome of **100Bp sequence reads**

⇓

No error-free (haplotypic) genome assembly (computational) method exist yet!

History

[1990]: The Human Genome Project was launched through funding from the US National Institutes of Health (NIH) and Department of Energy.

[1998] A new private venture was launched to sequence the human genome named Celera Genomics.

[2000] Public and private enterprises both announced the completion of the draft genomes

[2001] Celera's effort appeared in *Science*; International Human Genome Sequencing Consortium (IHGSC)'s effort published in *Nature*.

[2003] The IHGSC announces the gold-standard reference (99.99% accuracy).

History (continued)

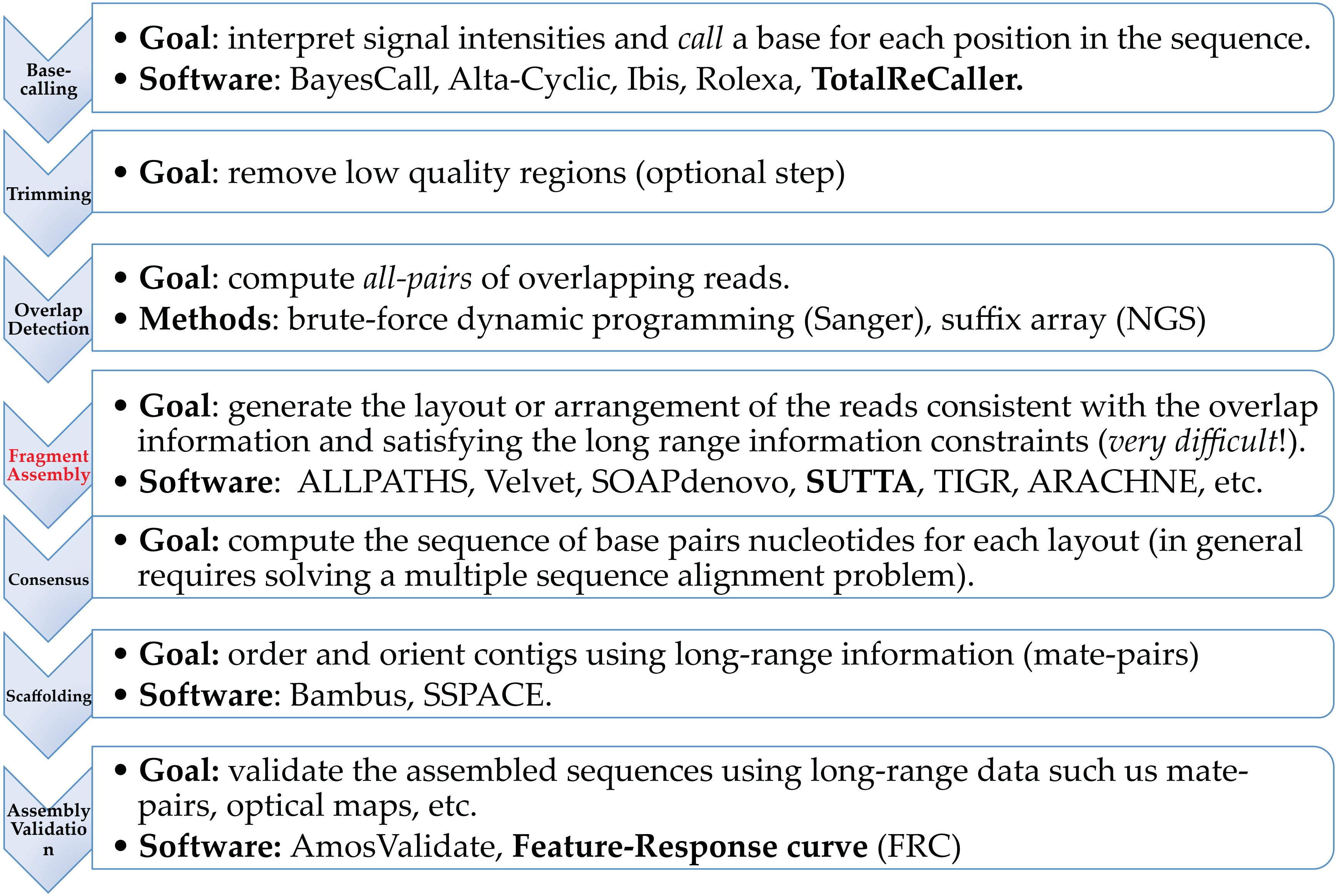
[2007] The Craig Venter Institute published an updated version of the human genome. This new sequence revealed more than 4.1 million DNA variants, encompassing 12.3Mb.

[2008] The first human genome (James D. Watson) sequenced by next-generation technologies is published.

[2009] First human genome assembled using next-generation short read data from Illumina, Inc (ABySS assembler).

[2010] The second human assembly using next-generation short read data is published using the assembler ALLPATHS-LG from Broad Institute.

Assembly pipeline



Base-calling

- **Goal**: interpret signal intensities and *call* a base for each position in the sequence.
- **Software**: BayesCall, Alta-Cyclic, Ibis, Rolexa, **TotalReCaller.**

Trimming

- **Goal**: remove low quality regions (optional step)

Overlap Detection

- **Goal**: compute *all-pairs* of overlapping reads.
- **Methods**: brute-force dynamic programming (Sanger), suffix array (NGS)

Fragment Assembly

- **Goal**: generate the layout or arrangement of the reads consistent with the overlap information and satisfying the long range information constraints (*very difficult*!).
- **Software**: ALLPATHS, Velvet, SOAPdenovo, **SUTTA**, TIGR, ARACHNE, etc.

Consensus

- **Goal:** compute the sequence of base pairs nucleotides for each layout (in general requires solving a multiple sequence alignment problem).

Scaffolding

- **Goal:** order and orient contigs using long-range information (mate-pairs)
- **Software**: Bambus, SSPACE.

Assembly Validation

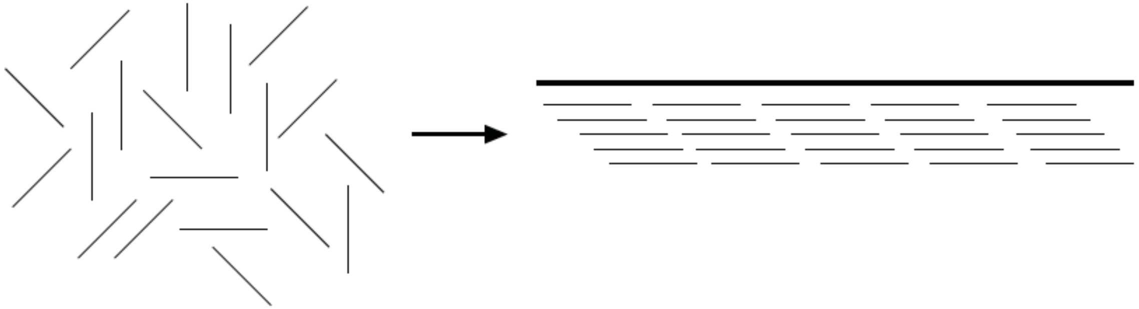
- **Goal:** validate the assembled sequences using long-range data such us mate-pairs, optical maps, etc.
- **Software:** AmosValidate, **Feature-Response curve** (FRC)

DNA SEQUENCE ASSEMBLY

Formulation and statistics

Shotgun sequence assembly

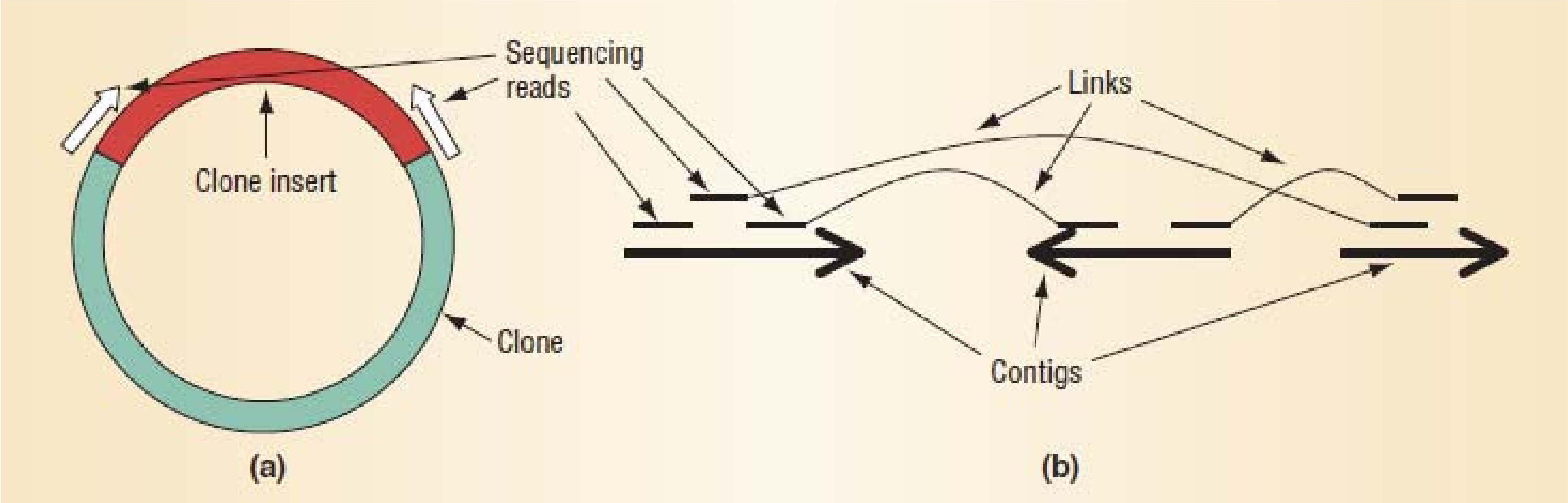
- DNA sequence is sheared into a large number of small fragments.



- **Assume**: If two sequence reads share the same string of letters (*overlap*), then they might have originated from the same genomic location.
- **Goal**: Join the sequences together using a computer program called assembler (similar to solving a jigsaw puzzle).
- **Add-ons**: Use long-range data to resolve complex genomic structures.

Paired-end and Mate-Pairs

Forward-reverse constraints



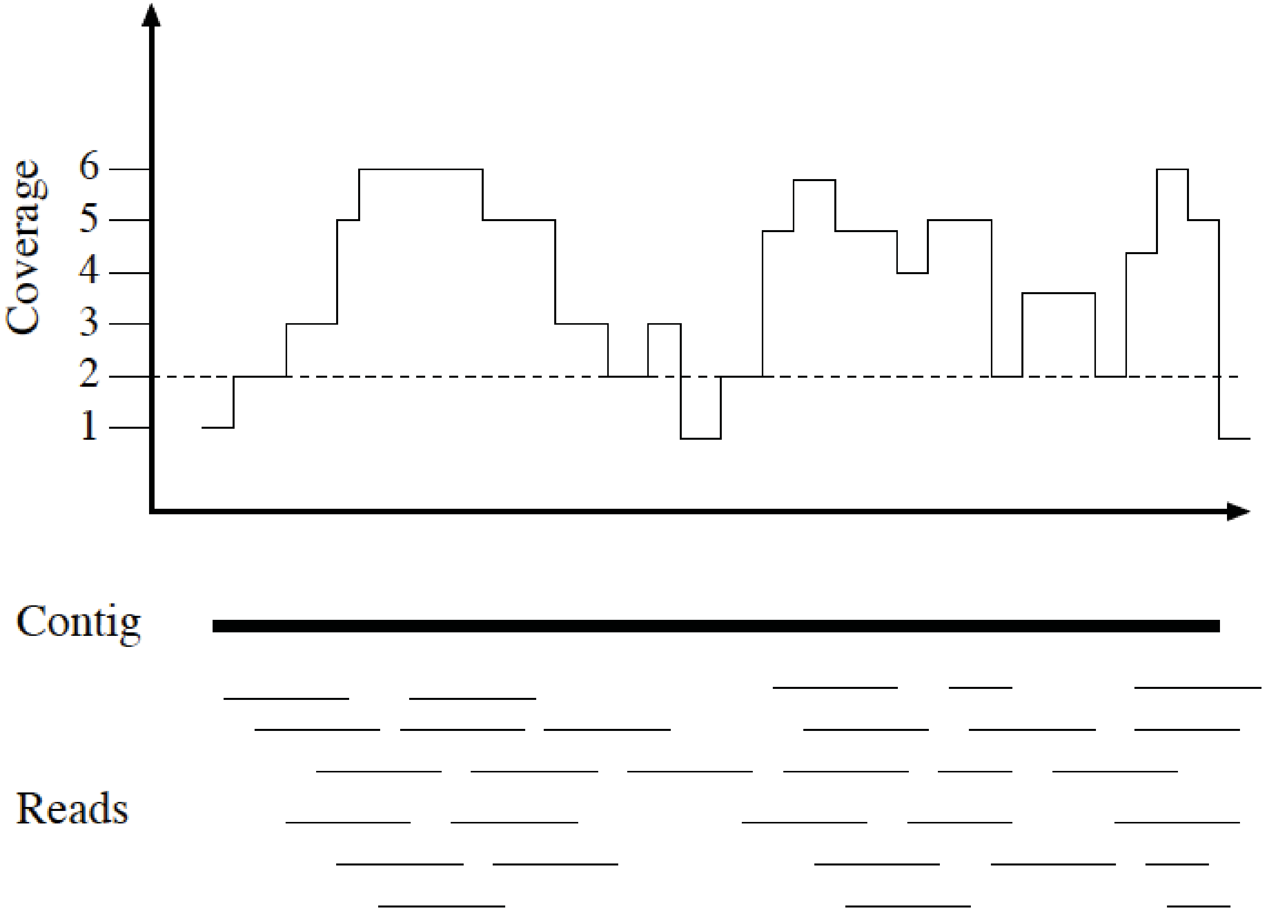
Pop et al. Genome Sequence Assembly: Algorithms and Issues. Computer (2002)

- **Properties**:
  - The sequence ends are facing towards each other (paired-end) or away from each other (mate-pairs).
  - The distance between the two fragments is known, within certain experimental error (μ ± σ).
- **Libraries**:
  - 200bp, 300bp, 1kbp, 5kbp, 10kbp

Coverage

Imagine raindrops on a sidewalk:

as the fragments are being sequenced, the randomness of the shearing process leads to cover successively more new sections of the original DNA.



Read coverage illustration (inspired by a lecture given by Michael Schatz in 2006 at the University of Hawaii).

Lander-Waterman statistics

Lander and Waterman. **Genomics**, 1988

- Consider a genome of length *G* that has been uniformly randomly sampled to collect *N* fragments each one of length *L*.
  - *G* = Genome length (in bp).
  - *L* = Average length of a fragment (in bp).
  - *N* = Number of fragments.
  - ***c* = *LN*/*G* (Coverage).**
  - *T* = number of base pairs two fragments must have in common to ensure their overlap (overlap parameter).
  - $\sigma = 1 - \theta \quad (\theta = T/L)$

1X~ (1 times) coverage of the human genome requires:

$$N = \frac{cG}{L} = \frac{3 \times 10^9}{500} = 6 \quad million \quad reads!$$

10X~ coverage requires *N* = 60 million reads !

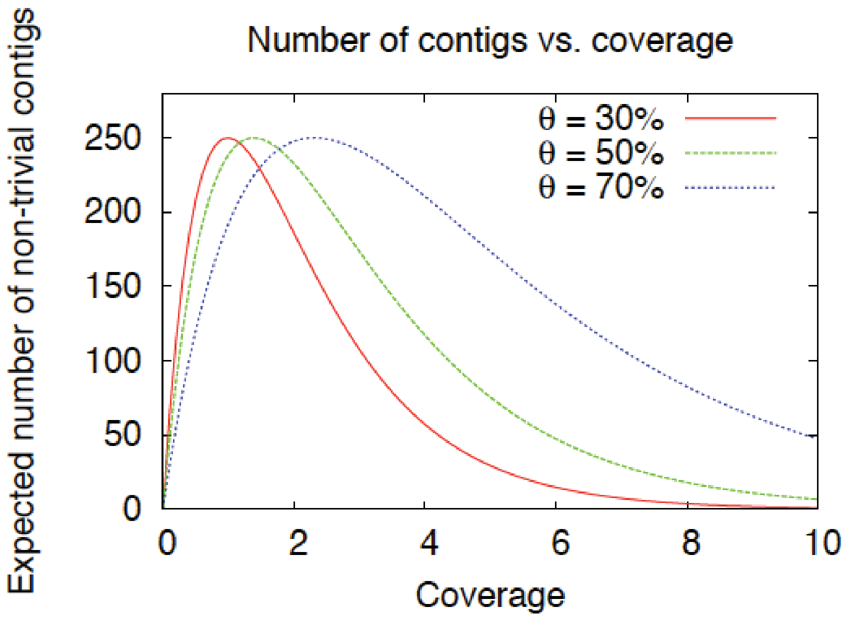
Contig statistics

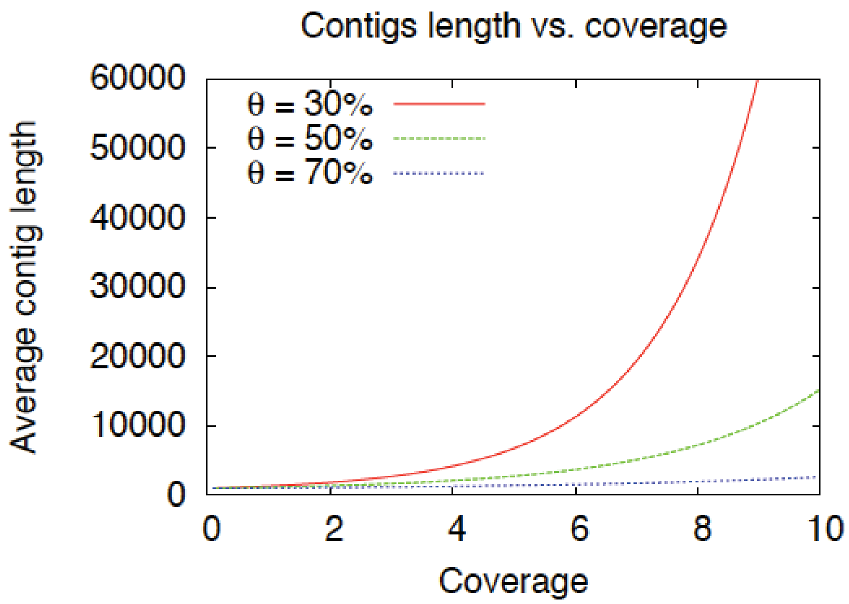
- If we model the "arrival" of *N* fragments of length *L* along a genome of length *G* as a Poisson process then the expected number of non-trivial contigs and their size is:

$$E[\ \#\quad non-trivial\quad contigs\ ] = Ne^{-(c\sigma)} - Ne^{-(2c\sigma)}$$

$$E[\ contig\quad size\ ] = L\left[\frac{e^{(c\sigma)}-1}{c} + (1-\sigma)\right]$$

non-trivial contig = contig with 2 or more reads.



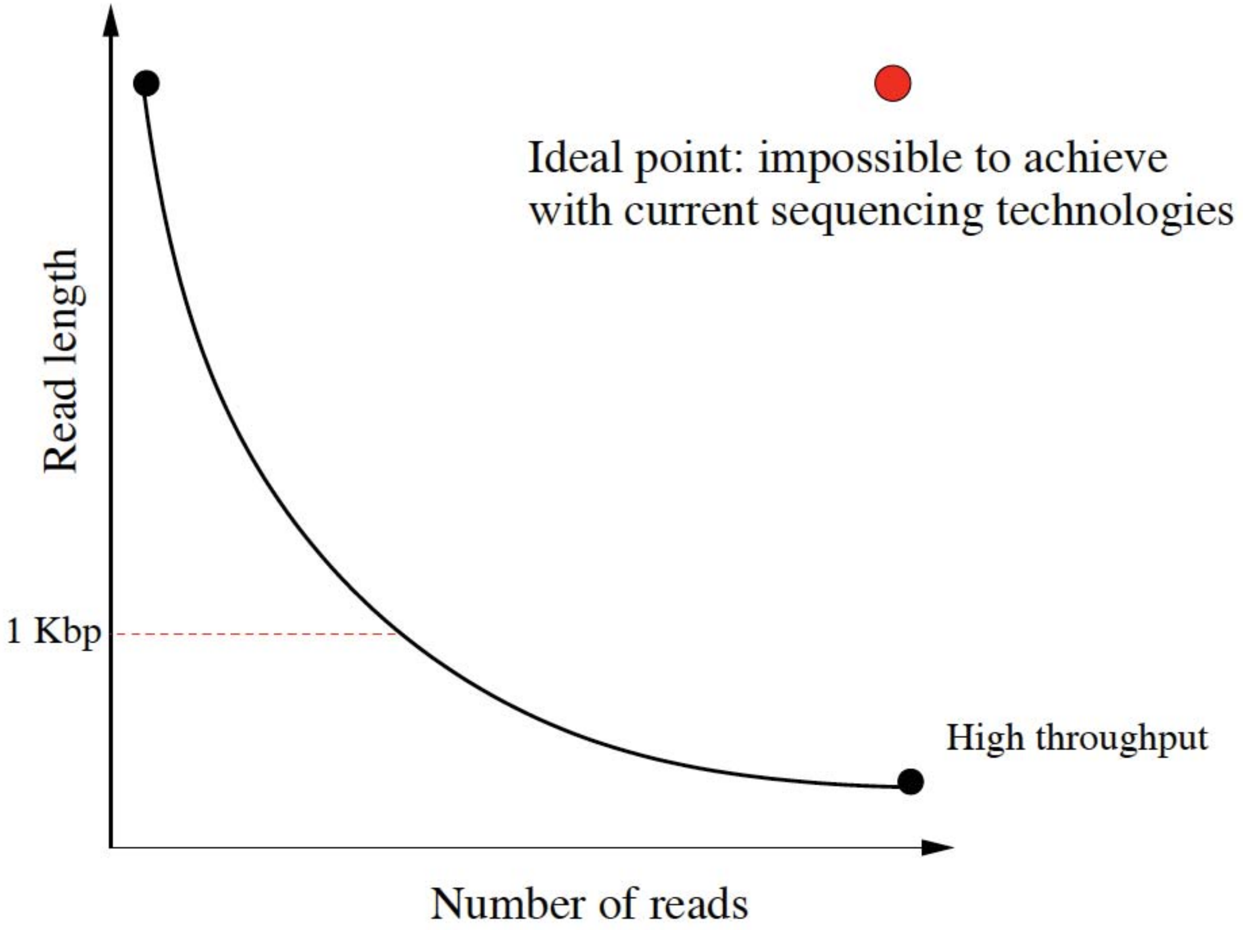


Read length tradeoff

Ideal: very long Reads (but currently no technology generates reads longer then 1Kb).

Solution: high throughput sequencing technologies (high coverage).

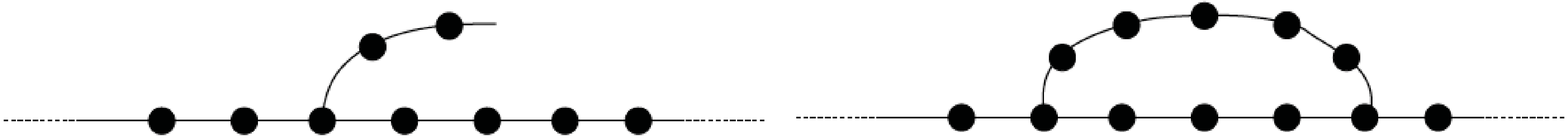
Problem: repeats!



- Trade-off between read length and coverage.

Challenges of new sequencing technology

- Short read lengths (up to 500 bps).
- Very high coverage (200X)
- Lots of data (requires distributed system approach).
- Dead-ends and Bubbles



- For short reads the required overlapping length represents a significant part of the read length.
- The **effective coverage** is more informative: $c_E = \dfrac{N(L-K)}{G}$

- *S. aureus* ($L = 35$, $G = 2.82$ Mbp, $N = 3.86$ Millions):
  - Raw coverage $c = LN/G = 48$X
  - Effective coverage ($K = 21$) $cE = N(L-K)/G = 14$X

SEQUENCE ASSEMBLY PROBLEM

Computational Complexity

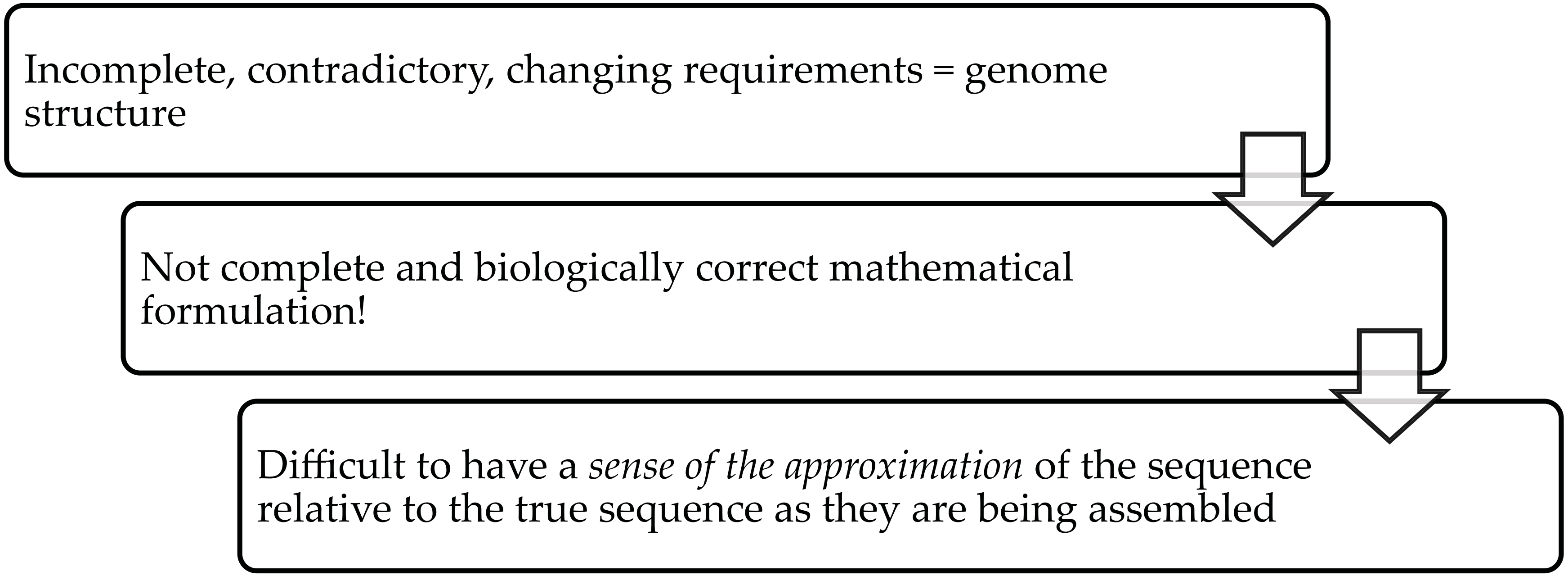
Why is de-novo assembly so difficult?

1. **NP-complete**: natural reduction to the *Shortest Superstring Problem* (easy for totally random DNA sequences).
2. **Genomic structures**: repeated regions, rearrangements, segmental duplications etc.
3. **Sequencing-Technology Dependent**:
    1. algorithms must change to accommodate changes to read-length or nature and availability of long-range information.
    2. Sequencing machine have different error profiles

The Sense of the Approximation

A wicked problem in search for a correct solution

- A **wicked** problem is a problem that is difficult or impossible to solve because of *incomplete, contradictory,* and *changing* requirements that are often difficult to recognize.



Incomplete, contradictory, changing requirements = genome structure

Not complete and biologically correct mathematical formulation!

Difficult to have a *sense of the approximation* of the sequence relative to the true sequence as they are being assembled

Shortest Superstring Problem

First approximation

- *Given a set of strings* $\{f_1, f_2, \ldots, f_n\}$ *find the shortest string* R *(reconstruction) such that* $\forall i$, f_i *is a substring of* R.

- **First issue**: NP-complete problem! [Gallant et al. 1980]

- **Second issue**: it does not correctly model the assembly problem:

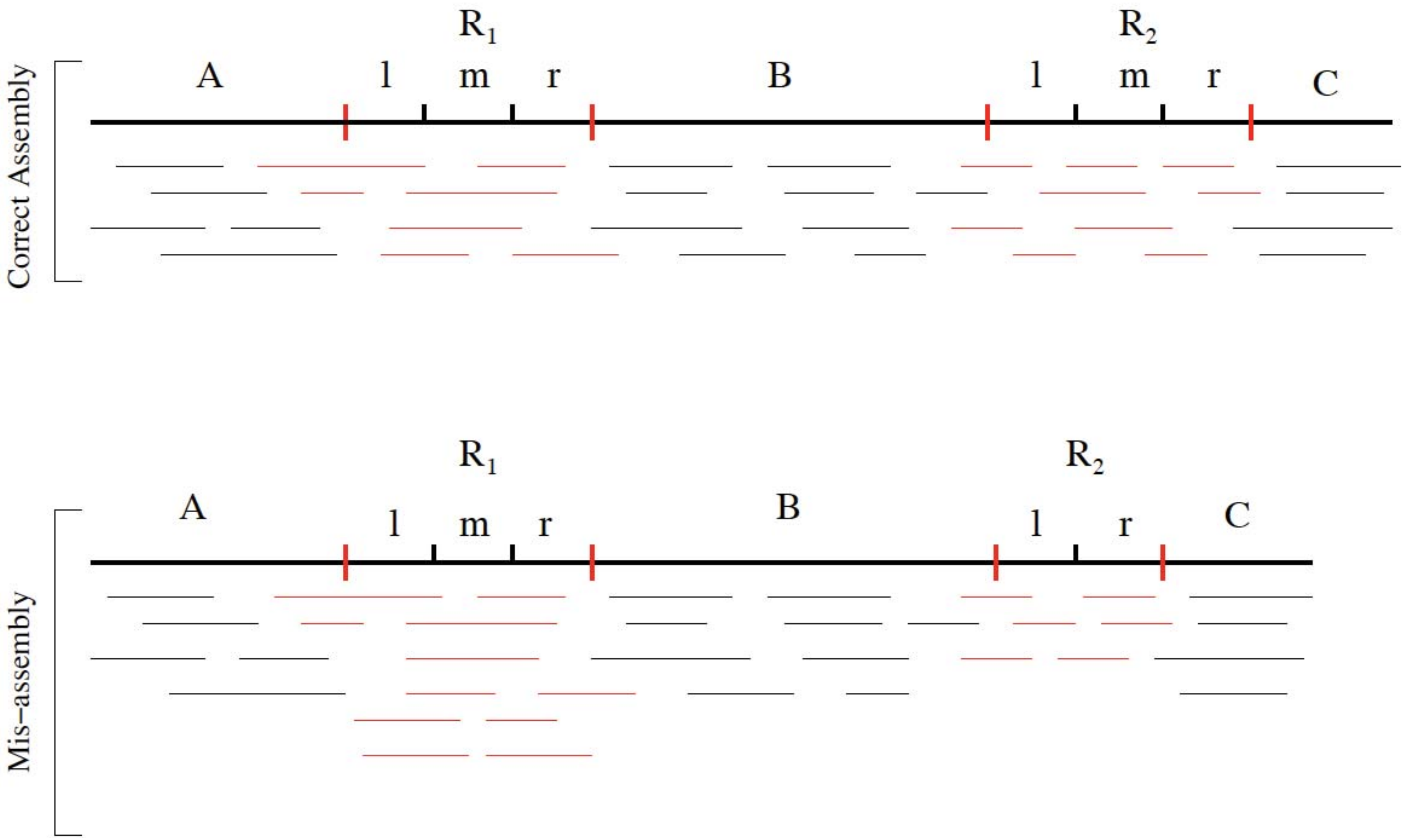
  "An elegant theoretical abstraction, but fundamentally flawed"

  [Richard Karp. *Computational Systems Bioinformatics Conference.* 2003]

- Sequencing errors? Fragment orientation? Repeats?

Repeats

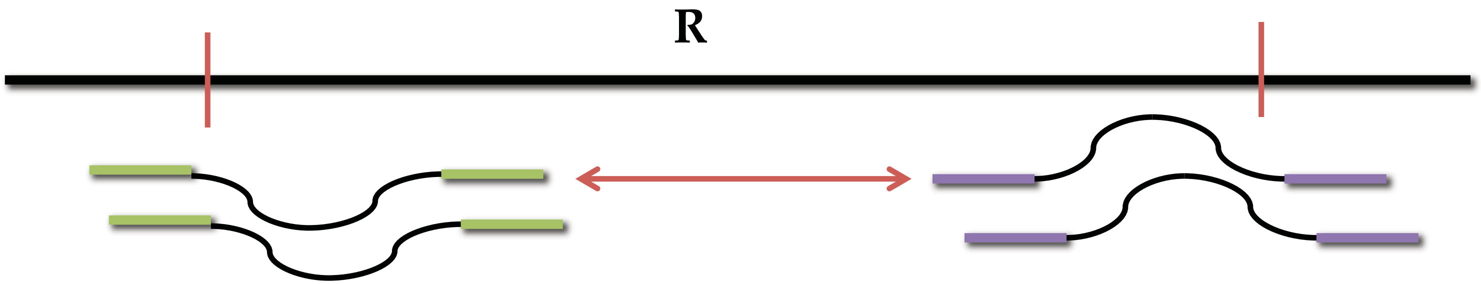
- If we look for a reconstruction of minimum length, the reconstructed string can have many errors due to repeats.



Repeat types

- **Tandem Repeats:**
  - Microsatellite: $(a_1 \ldots a_k)^k$ where k ~ 3-6
    (e.g. CAGCAGTAGCAGCACCAG)
- **Interspersed repeats:**
  - SINEs (Short Interspersed Nuclear Elements)
    (e.g., Alu: ~300 bp long, 10^6 copies)
  - LINEs (Long Interspersed Nuclear Elements)
    (e.g., ~ 500 - 5,000 bp long, 200,000 copies)

Use mate-pare to resolve repeats, however:
The maximum length of repeal R that can be spanned is twice the maximum length of the clones (the repeat region can be walked into from both sides).



A better formulation

Narzisi and Mishra, **Bioinformatics**, 2011

Constrained optimization problem

- Given a collection of fragments *F* and a tolerance level *ε* find a reconstruction *R* use layout *L* is ***ε*-valid**, **consistent** and such that the following properties are satisfied:

① **Overlap Constraint** : the cumulative overlap score *O* of the layout *L* is optimized.

② **Mate-Pair-Constraint:** The cumulative mate-pair score S_{MP} of the distance between reads in the layout *L* is consistent with the mate-pair constraints.

③ **Optical-Map-Constraint:** The observed distribution of restriction enzyme sites in the layout *L*, is consistent with the distribution of experimental optical map (obtained by a restriction enzyme digestion process).

④ …

Goal: perform assembly and validation in a *unified step.*

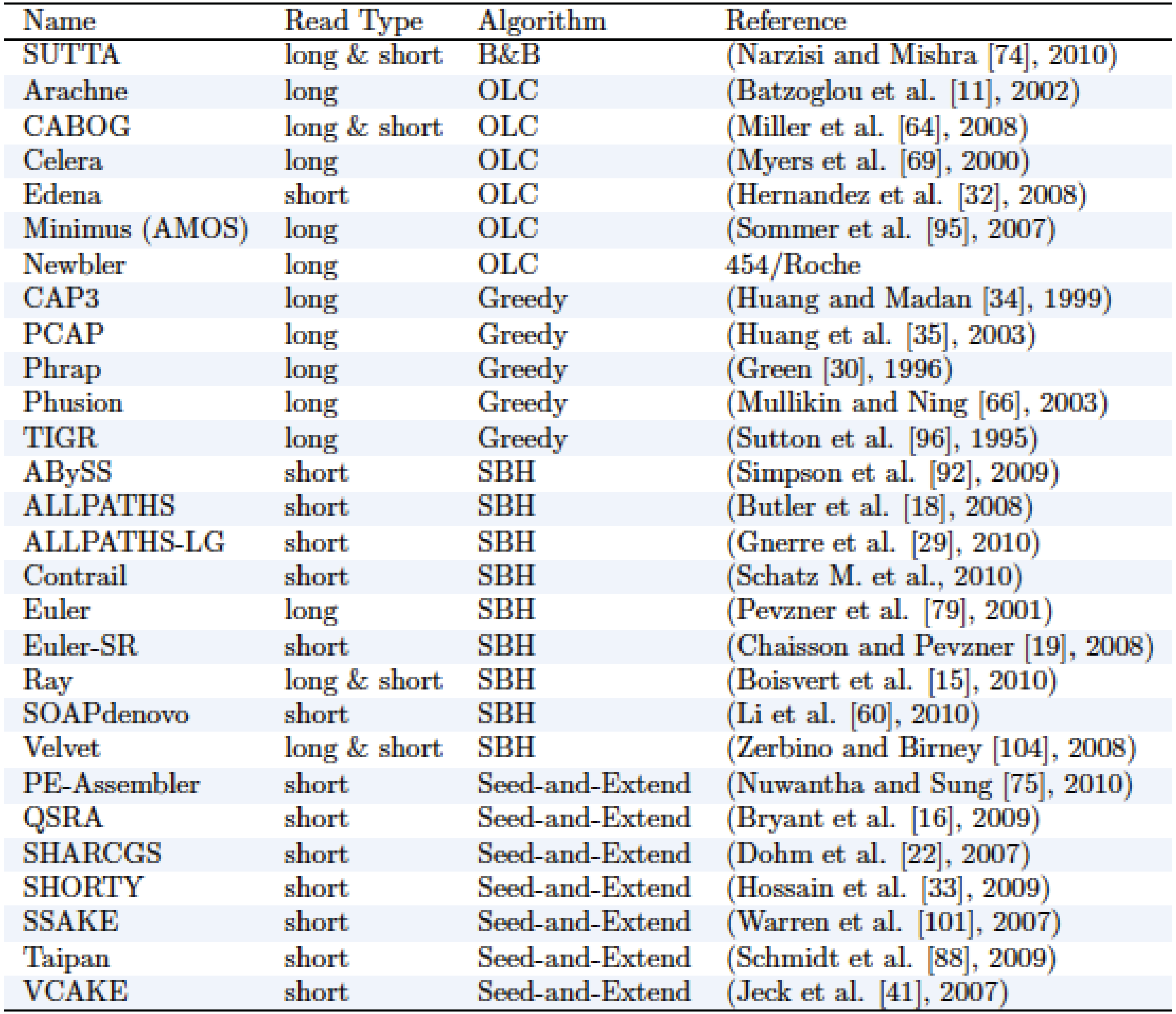
Myers proposed to design "algorithms that are capable of solving a 'pure' shotgun problem….", however, he explains that such a *shotgun-with-constraints* problem should be explored "if there is to be any hope of solving these more difficult constraint problems"

[Myers. **Journal of Computational Biology**, 2:275–290, 1995]

ASSEMBLY PARADIGMS

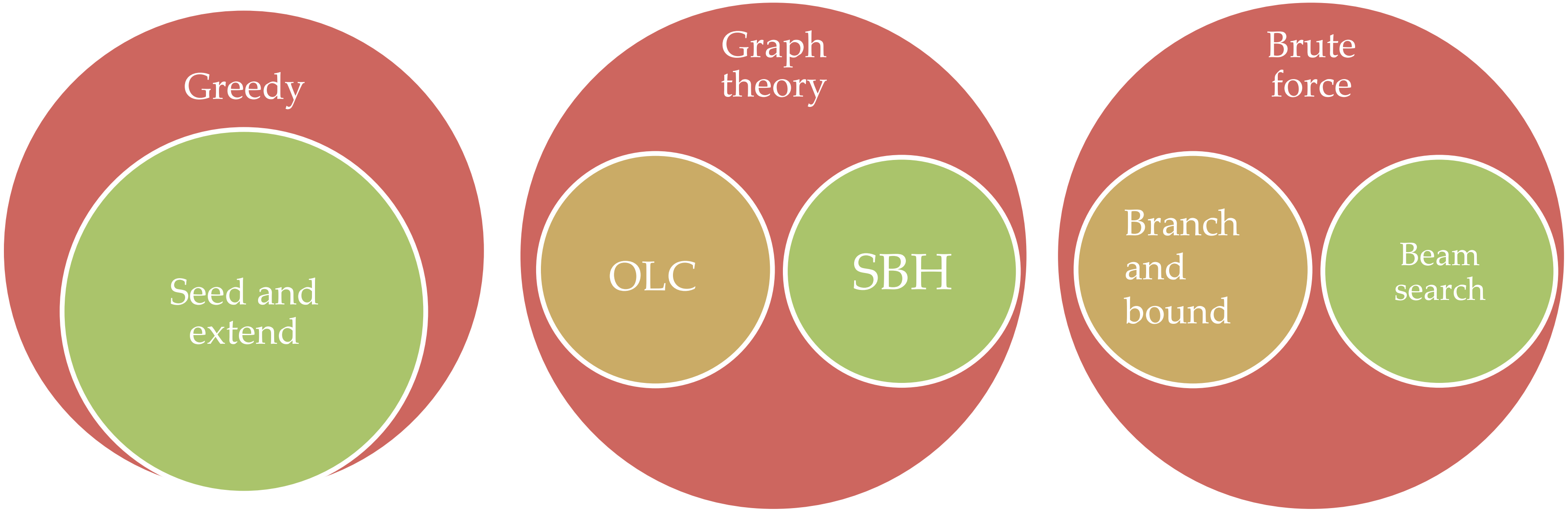
The art of solving a difficult puzzle

Sequence Assemblers



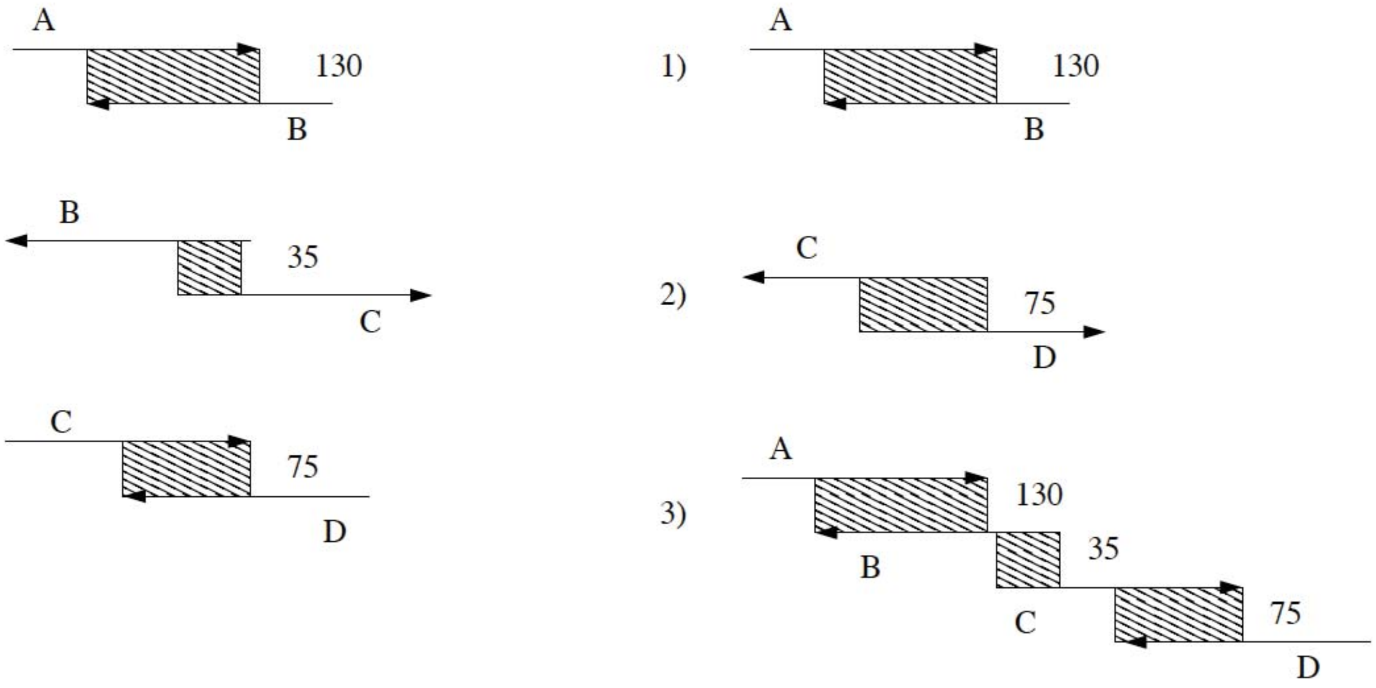
Name	Read Type	Algorithm	Reference
SUTTA	long & short	B&B	(Narzisi and Mishra [74], 2010)
Arachne	long	OLC	(Batzoglou et al. [11], 2002)
CABOG	long & short	OLC	(Miller et al. [64], 2008)
Celera	long	OLC	(Myers et al. [69], 2000)
Edena	short	OLC	(Hernandez et al. [32], 2008)
Minimus (AMOS)	long	OLC	(Sommer et al. [95], 2007)
Newbler	long	OLC	454/Roche
CAP3	long	Greedy	(Huang and Madan [34], 1999)
PCAP	long	Greedy	(Huang et al. [35], 2003)
Phrap	long	Greedy	(Green [30], 1996)
Phusion	long	Greedy	(Mullikin and Ning [66], 2003)
TIGR	long	Greedy	(Sutton et al. [96], 1995)
ABySS	short	SBH	(Simpson et al. [92], 2009)
ALLPATHS	short	SBH	(Butler et al. [18], 2008)
ALLPATHS-LG	short	SBH	(Gnerre et al. [29], 2010)
Contrail	short	SBH	(Schatz M. et al., 2010)
Euler	long	SBH	(Pevzner et al. [79], 2001)
Euler-SR	short	SBH	(Chaisson and Pevzner [19], 2008)
Ray	long & short	SBH	(Boisvert et al. [15], 2010)
SOAPdenovo	short	SBH	(Li et al. [60], 2010)
Velvet	long & short	SBH	(Zerbino and Birney [104], 2008)
PE-Assembler	short	Seed-and-Extend	(Nuwantha and Sung [75], 2010)
QSRA	short	Seed-and-Extend	(Bryant et al. [16], 2009)
SHARCGS	short	Seed-and-Extend	(Dohm et al. [22], 2007)
SHORTY	short	Seed-and-Extend	(Hossain et al. [33], 2009)
SSAKE	short	Seed-and-Extend	(Warren et al. [101], 2007)
Taipan	short	Seed-and-Extend	(Schmidt et al. [88], 2009)
VCAKE	short	Seed-and-Extend	(Jeck et al. [41], 2007)

Assembly paradigms



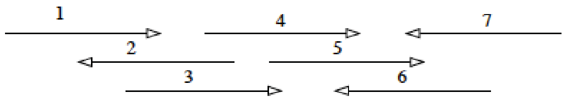
Greedy strategy
(TIGR 1995, Phrap 1996, CAP3 1999)

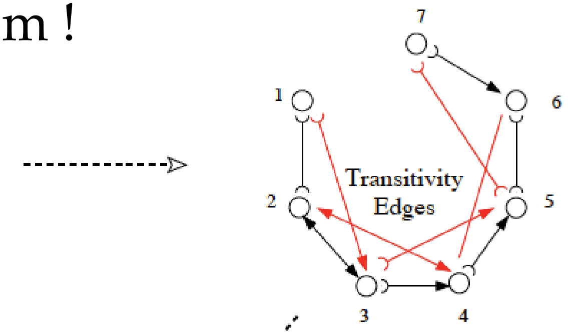
① Pick the highest scoring overlap.
② Merge the two fragments (add this new sequence to the pool of sequences).
③ Heuristically correct regions of the overlay in some plausible manner (whenever possible).
④ Regions that do not yield to these error-correction heuristics are abandoned as irrecoverable and shown as gaps.
⑤ Repeat until no more merges can be done.



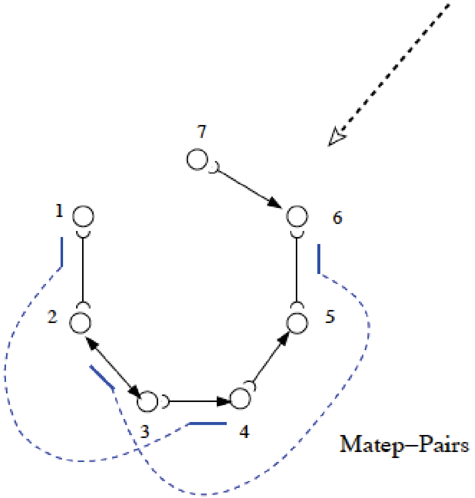
Overlap-Layout-Consensus
(ARACHNE 2002, CELERA 2000, Minimus 2007)

- **Idea:** Construct a graph where nodes represent reads and edges indicate overlaps.
- **Goal:** Need to solve an Hamiltonian path problem !





- **Heuristic strategy**:
  ① Remove *contained* and *transitivity* edges.
  ② Collapse "unique connector" overlaps (chordal subgraph with no conflicting edges).
  ③ Use mate-pairs to validate, connect and order the contigs.



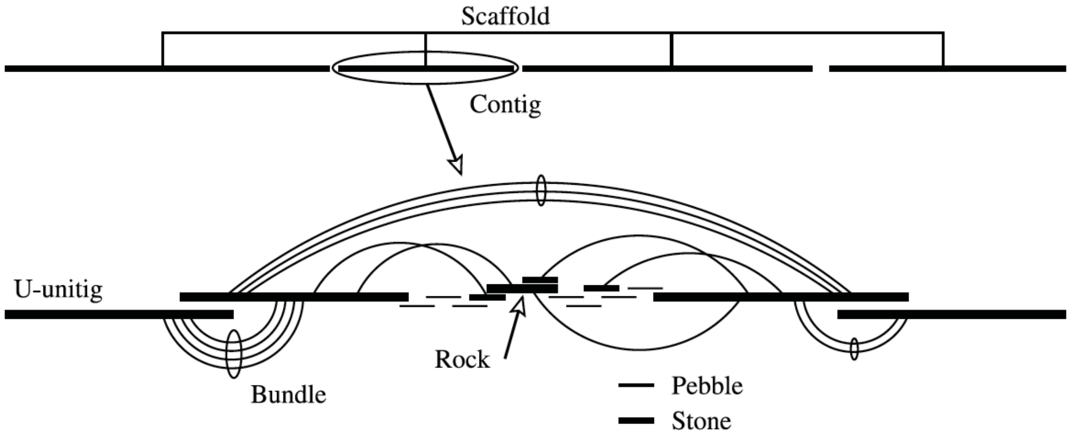
Contigs = nonintersecting simple paths in the reduced graph.

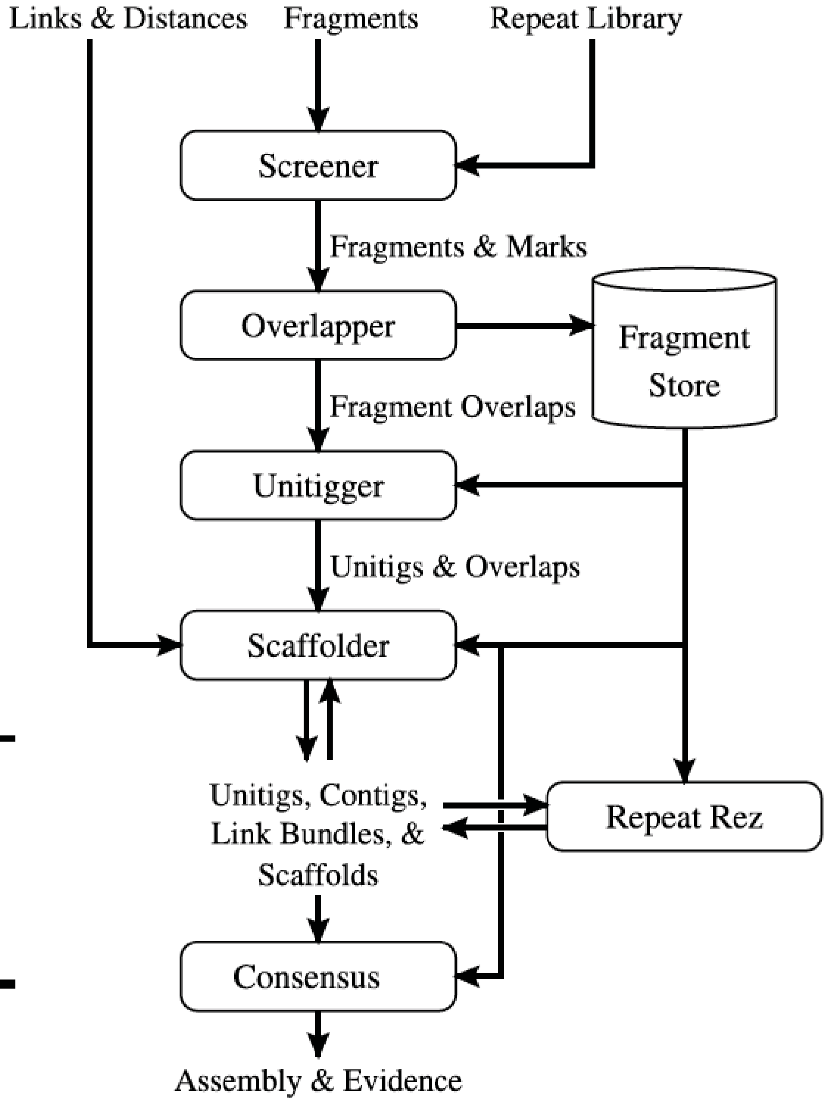
Celera/CABOG

Myers *et al.* **Science** 2000

- First large-scale assembly in 2000: *Drosophila* - 120 Mbp

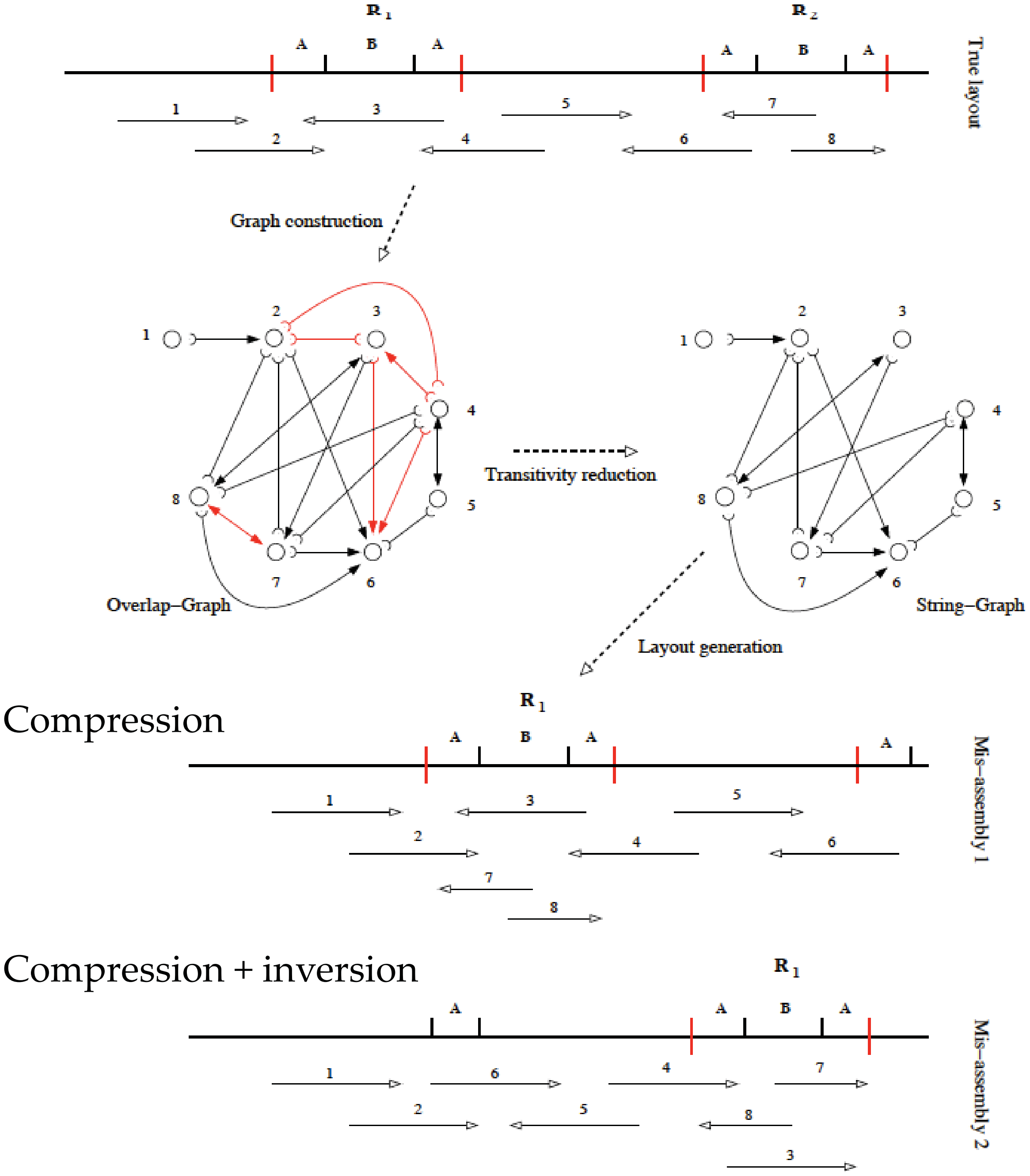
- Time: ~week





Example of miss-assembly

After removing the transitivity edges every (Hamiltonian) path is misassembled.

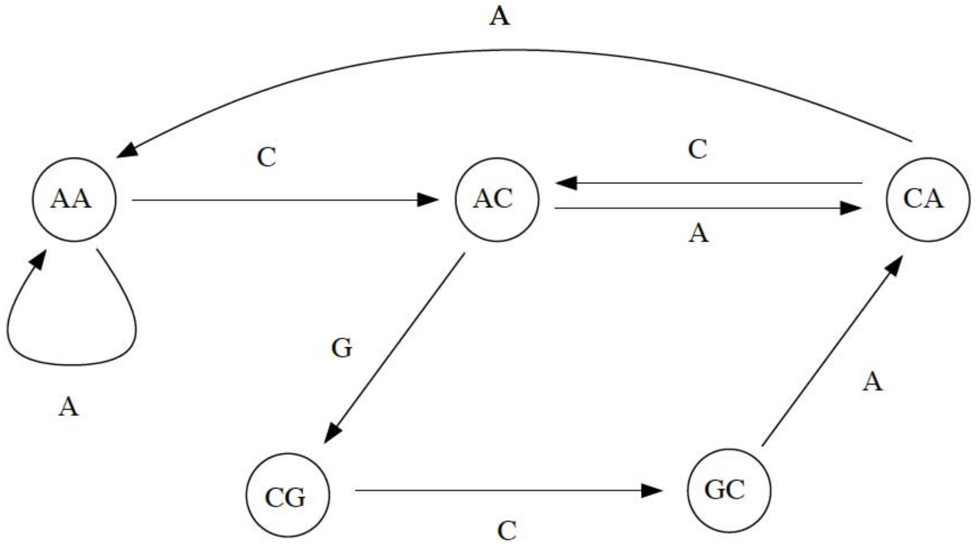


Compression

Compression + inversion

Sequencing by Hybridization
(EULER 2001, Velvet 2008, SOAPdenovo, ALLPATH-LG 2011)

- **Idea**: Build a *DeBruijn* graph G(V, E):
  - V = all possible *n*-mers
  - E = overlaps of size *n*-1. The source and destination nodes are respectively the *n* − 1 prefix and *n* − 1 suffix of the corresponding *n*-mer.



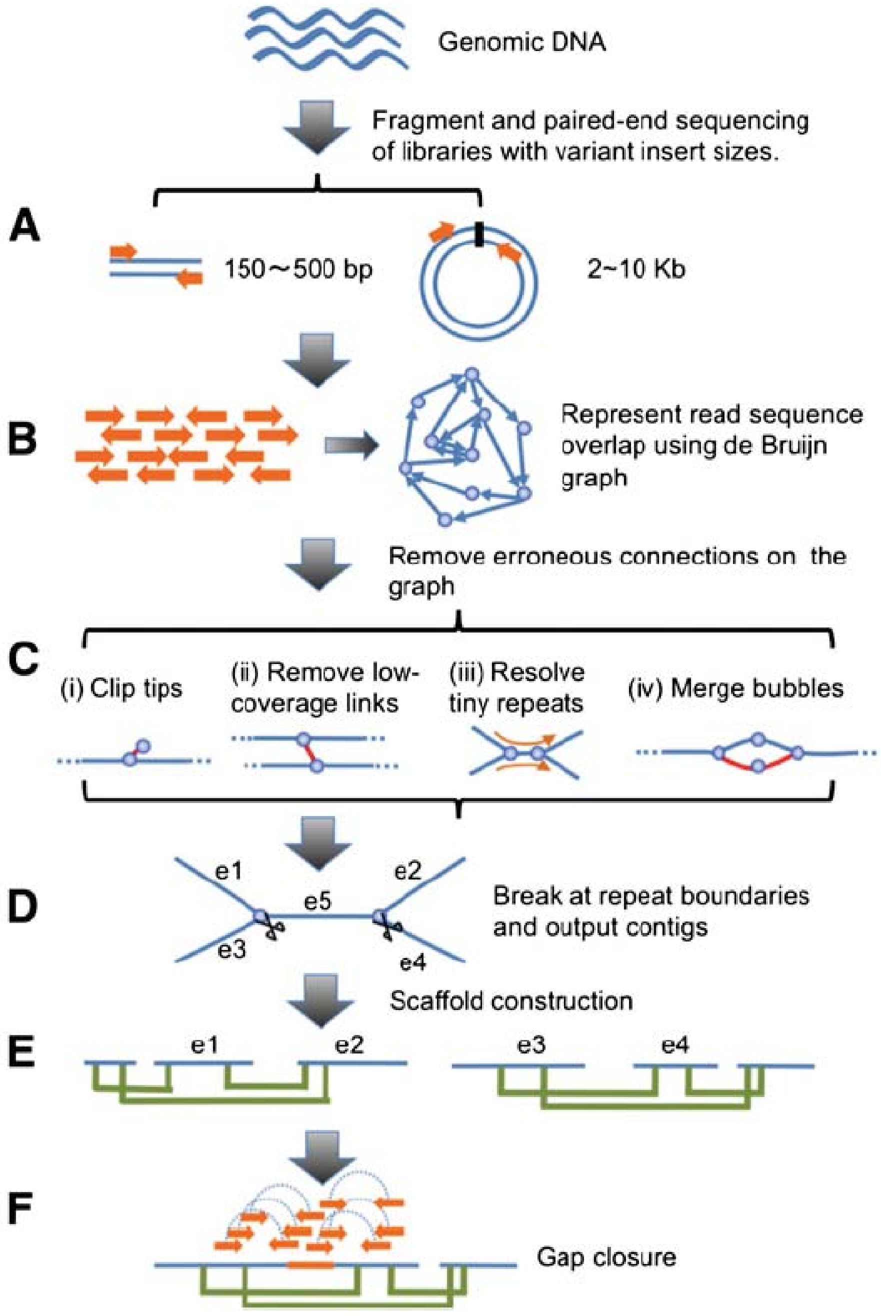
DeBruijn graph for the list *L* = {*AAA*, *AAC*, *ACA*, *CAC*,*CAA*, *CGC*,*GCG*}.
The Euler path is: *AC* ➔ *CA* ➔ *AC* ➔ *CG* ➔ *GC* ➔ *CA* ➔ *AA* ➔ *AA* ➔ *AC*

- **Ideal Goal**: find an Eulerian path (linear time algorithm).
- **Real Goal**: Eulerian-superpath. Given an Eulerian graph and a sequence of paths, find an Eulerian path in the Eulerian graph that contains all these paths as sub-paths (*NP*-hard).

SOAPdenovo

Li *et al.* **Genome Research** 2009

- In practice no one computes Eulerian paths
- Use heuristics instead!
  - Similar to the OLC approach


Genomic DNA
Fragment and paired-end sequencing of libraries with variant insert sizes.
A
150～500 bp
2~10 Kb
B
Represent read sequence overlap using de Bruijn graph
Remove erroneous connections on the graph
C
(i) Clip tips
(ii) Remove low-coverage links
(iii) Resolve tiny repeats
(iv) Merge bubbles
D
e1
e2
e5
e3
e4
Break at repeat boundaries and output contigs
Scaffold construction
E
e1
e2
e3
e4
F
Gap closure

De Novo Genome Assembly

- *“An assembler must either "guess" (often incorrectly) the correct genome from among a large number of alternatives (a number that grows exponentially with the number of repeats in the genome) or restrict itself to assembling only the non-repetitive segments of the genome, thereby producing a fragmented assembly.”*

  [Pop and Salzberg, **Trends in Genetics**, 2008]

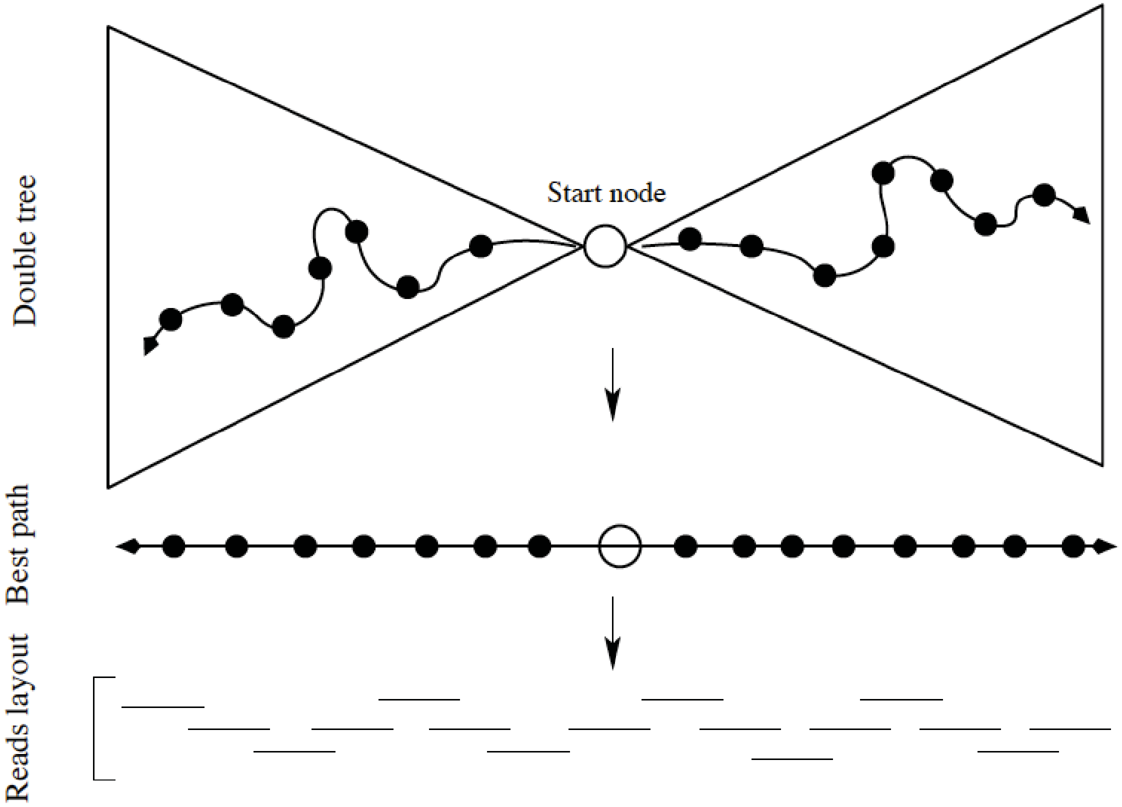
Is there anything in between??

Branch and Bound

- **Applications**: TSP (traveling-salesman prob.), MAX-SAT (maximal satisfiability), QAP (quadratic assignment prob.), …
- **Idea**: search complete space of solutions (exhaustive search, not greedy).
- **Caveat**: explicit enumeration is impossible (exponential).
- **Solution**: explore only the subspace that contains the optimum by pruning implausible overlays quickly.
- **How?**: use *smart* functions to be optimized.

SUTTA

Narzisi and Mishra, **Bioinformatics** 2011

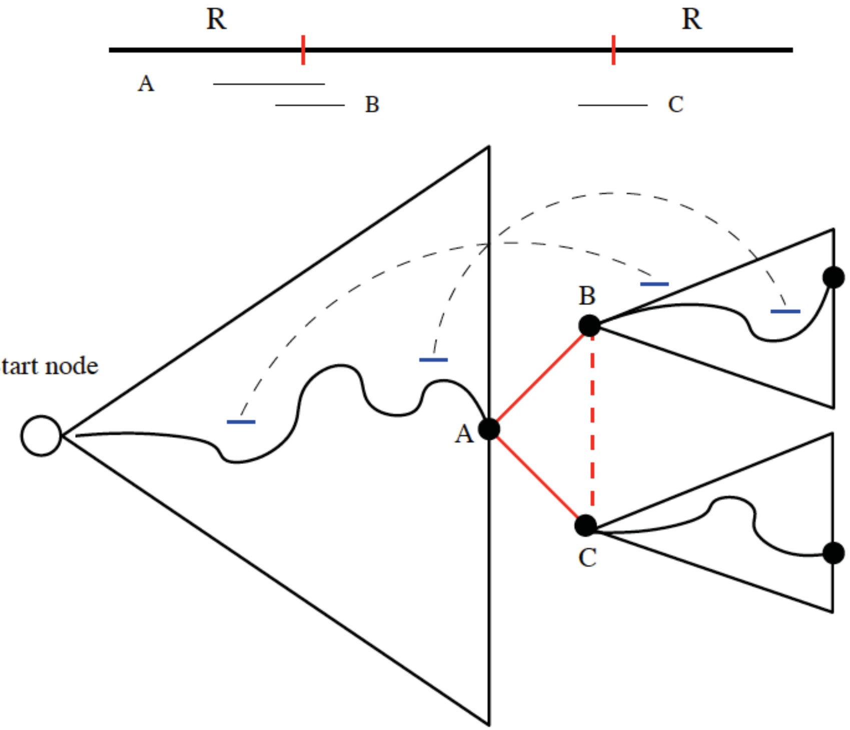


- Generate LEFT and RIGHT trees for the start read.
- Best LEFT path is concatenated with the root and the best RIGHT path to create a globally optimal contig.

Lookahead

How to resolve repeats

- **Scenario**: A potential repeat boundary between reads *A*, *B* and *C*. Read *A* overlaps both reads *B* and *C*, but *B* and *C* do not overlap each other.
- **Observation**: No decision can be made at this point on which read to keep/prune.



- **Idea**: Chose between reads *A* and *B* based on how well the mate-pairs (or other long-range data) in their subtree satisfy the length constraints.
- Easily extendable to resolve **dead-ends** and **bubbles.**

How to choose an assembler

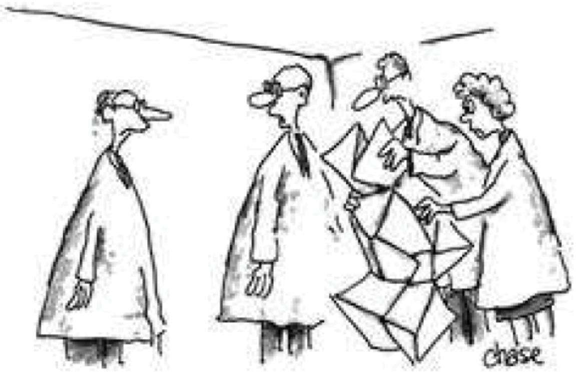
- **Do we need so many sequence Assemblers?**
  - I would like to say no, but…
- **What is the best sequence Assembler?**
  - Depends on application, type of data (sequencing technology), genome type (bacteria, human, etc) and size.
- **Specialized or general Assembler?**
  - Specialization is good, but a more general (flexible) framework should be devised.
- **Universal Assembler?**
  - Probably an utopic idea.

ASSEMBLY QUALITY

How to evaluate an assembly

The need for Quality Assessment

- **How well did we do?**
  - Beware of mis-assembled genomes.
    [Salzberg and Yorke. *Bioinformatics* (2005)]
  - Revolution Postponed: Why the Human Genome Project Has Been Disappointing
    [Stephen S. Hall, *Scientific American*, 2010]
  - Limitations of next-generation genome sequence assembly.
    [Alkan et al. *Nat Methods* (2011)]
  - Assemblies: the good, the bad, the ugly.
    [Birney. *Nat Methods* (2011)]

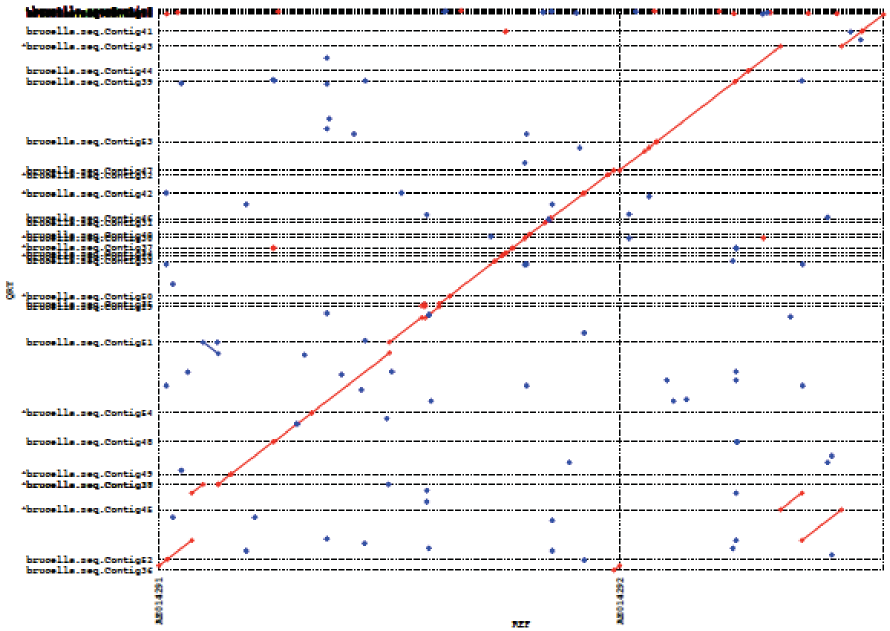


- Need for Quality Assessment!
- Assemblathon
  (but only very recently, 2011)

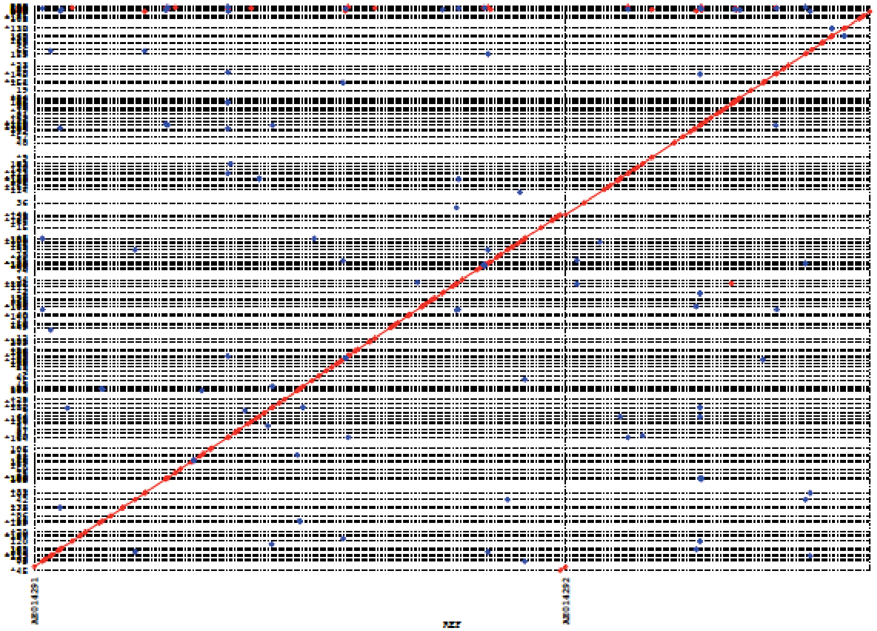
N50 contig size

- *Given M contigs of size $c_1, c_2, \ldots, c_M$,* ***N50*** *is defined as the largest number L such that the combined length of all contigs of length ≥ L is at least 50% of the total length of all contigs.*

- **Problem**: emphasizes only size, without capturing quality!



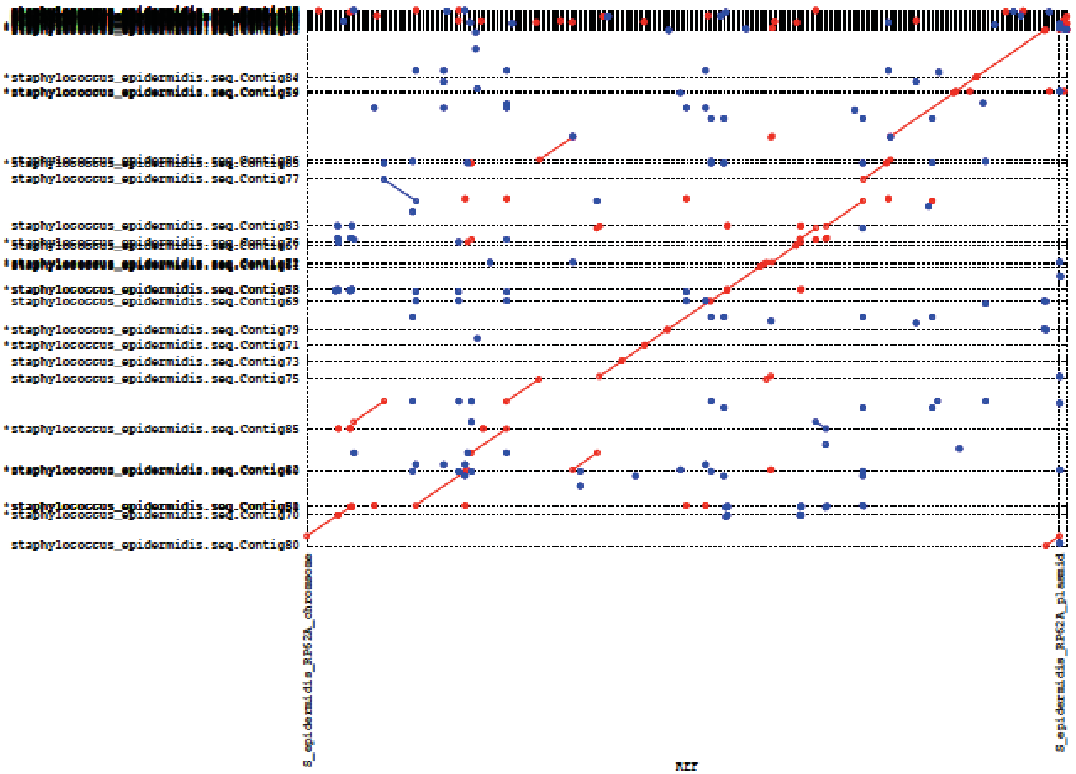
Few very long contigs: useless if mis-assembled.



Many short contigs: too short for annotation efforts.

Counting errors…

Typically used for NGS data

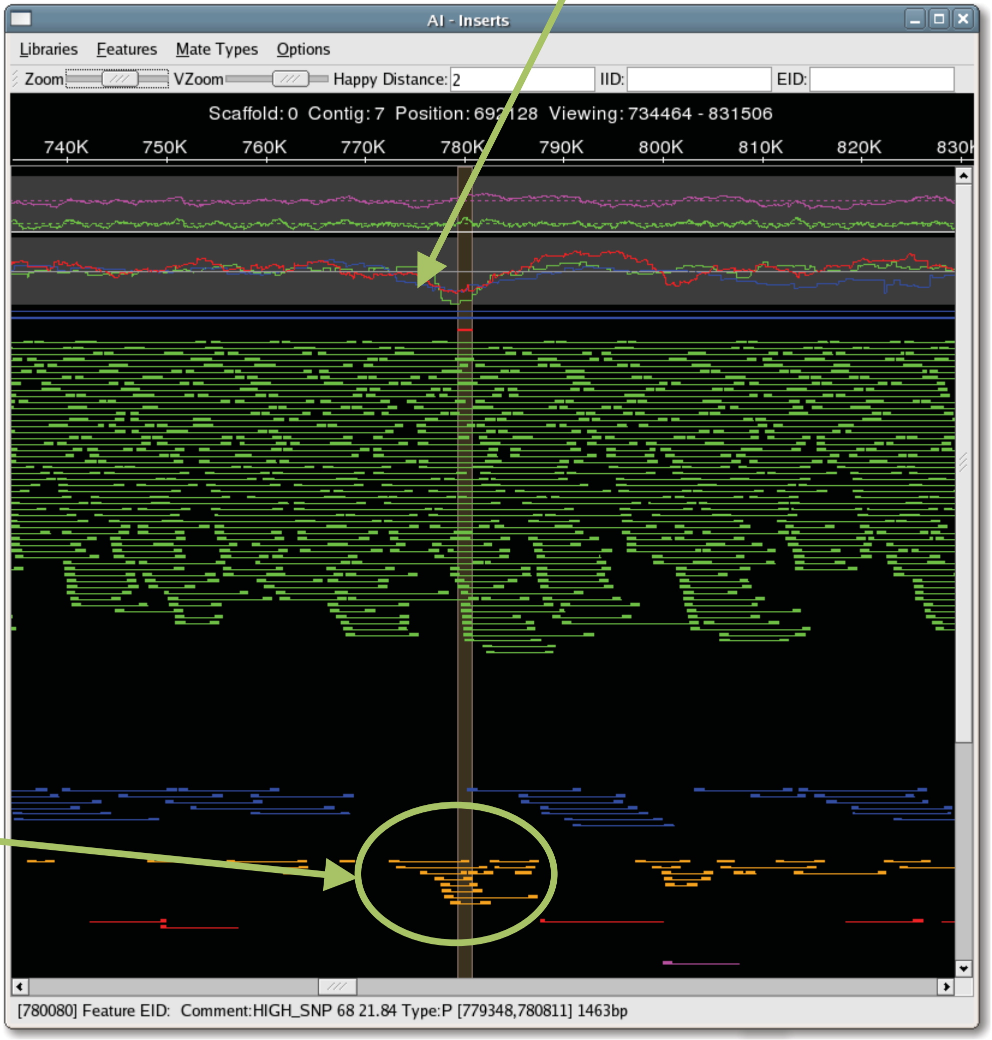


- **Count** the number of mis-assembled contigs by alignments to the reference genome (if available).
- **Problem**: error types are not weighted accordingly.

Visualization tools

Hawkeye: Schatz et al., Genome Biology 2007

- Good for inspection.
- Automation is needed!



Happy Distance: 2 IID: EID:

Scaffold: 0 Contig: 7 Position: 692128 Viewing: 734464 - 831506

[780080] Feature EID: Comment:HIGH_SNP 68 21.84 Type:P [779348,780811] 1463bp

-5.5 CE Dip

Compressed Mates Cluster

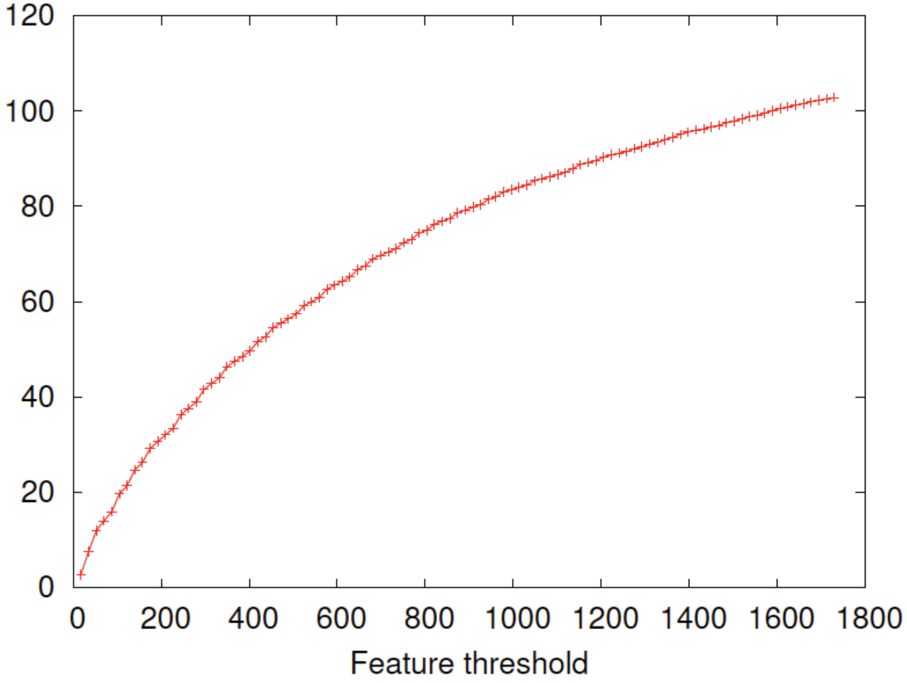
Ideal metric

- One single number or function
- Capture trade-off between assembly quality and contiguity
- Use long-range data for validation
- No need for a reference
- Easy to understand !

Feature-Response Curve

Narzisi and Mishra, **PLoS ONE** 2011

Goal: evaluate the structural properties of the contigs and of the reads arranged in the layout.

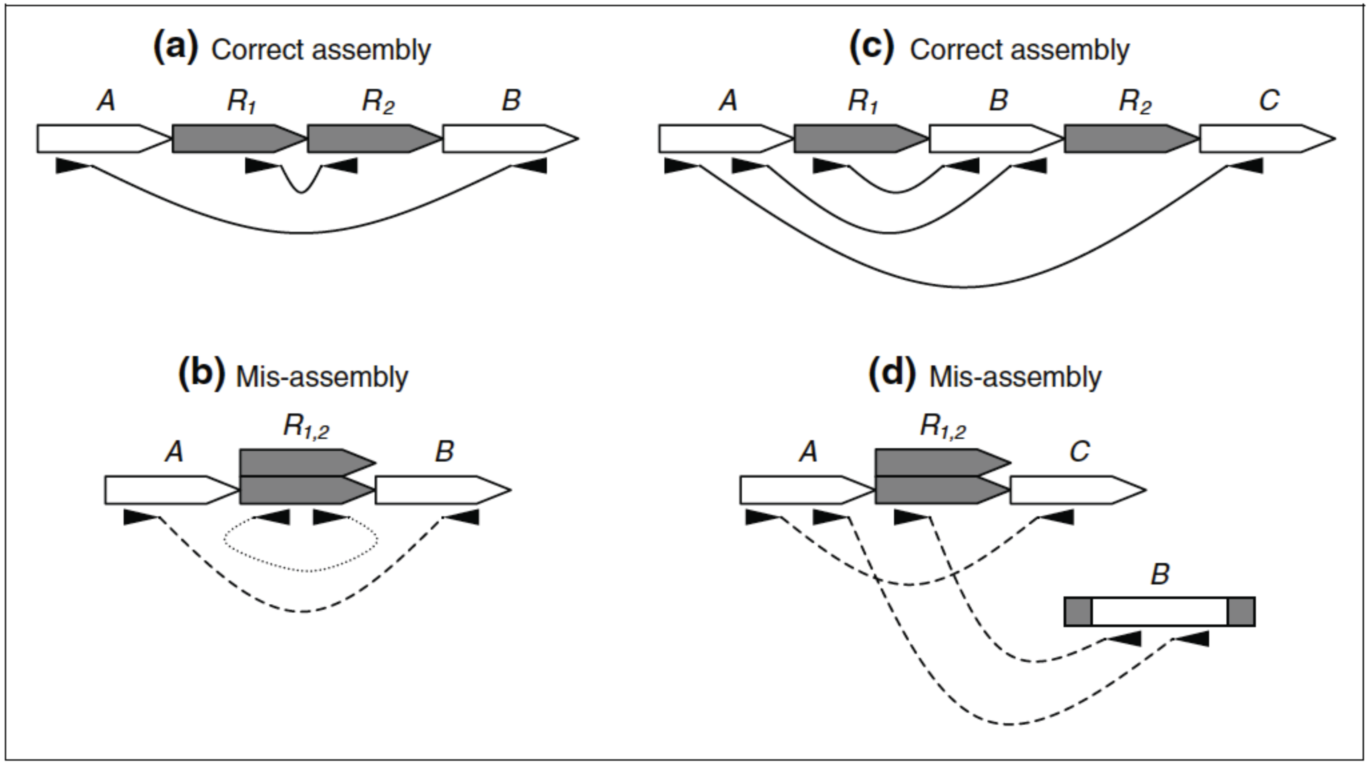


Characterizes the sensitivity (*coverage*) of the sequence assembler as a function of its discrimination threshold (*number of features/errors*).

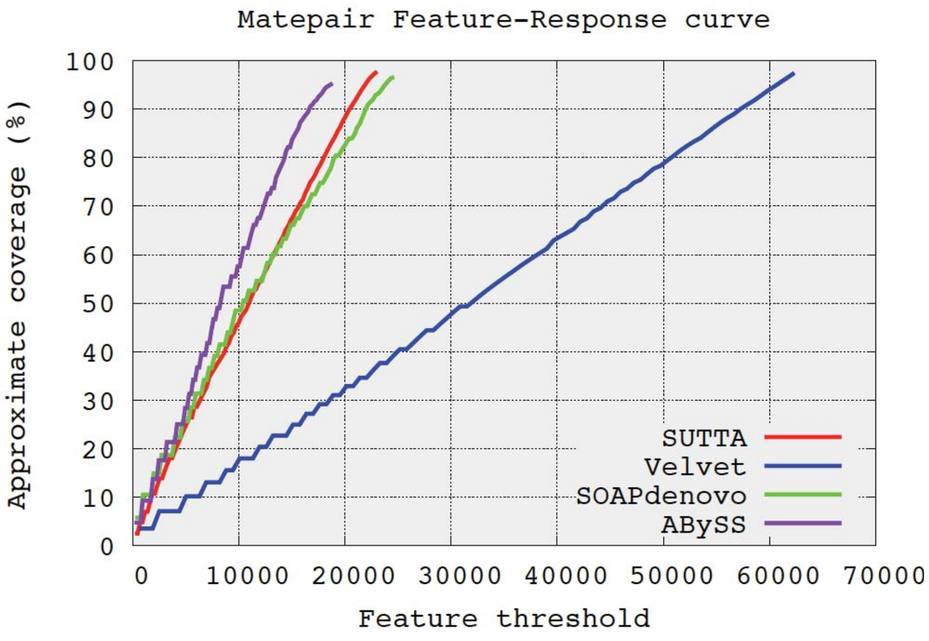
Features in amosvalidate

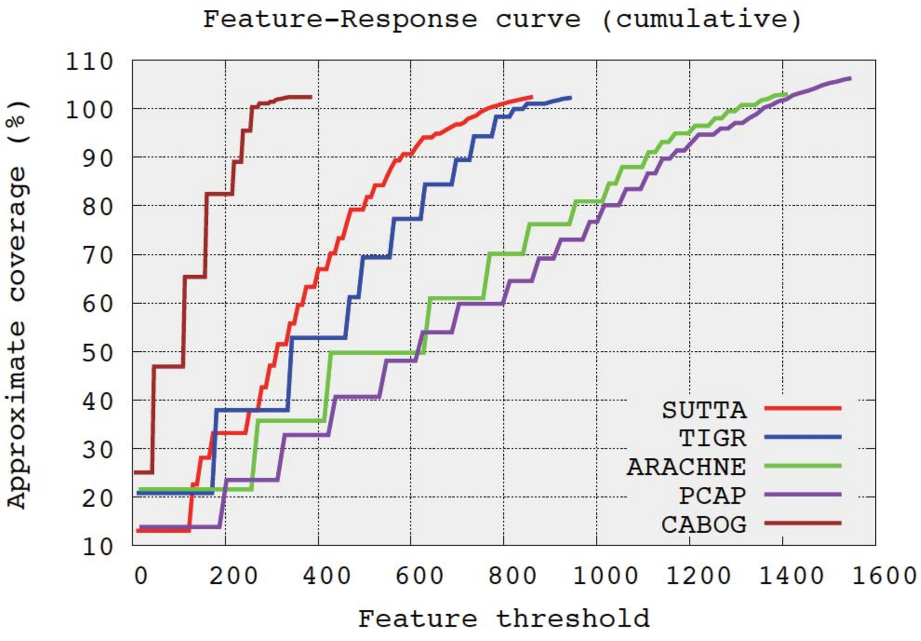
Phillippy, Schatz, Pop, **Genome Research** 2008

- (*M*) mate-pair orientations and separations,
- (*K*) repeat content by *k*-mer analysis,
- (*C*) depth-of-coverage,
- (*P*) correlated polymorphism in the read alignments, and
- (*B*) read alignment breakpoints to identify structurally suspicious regions.



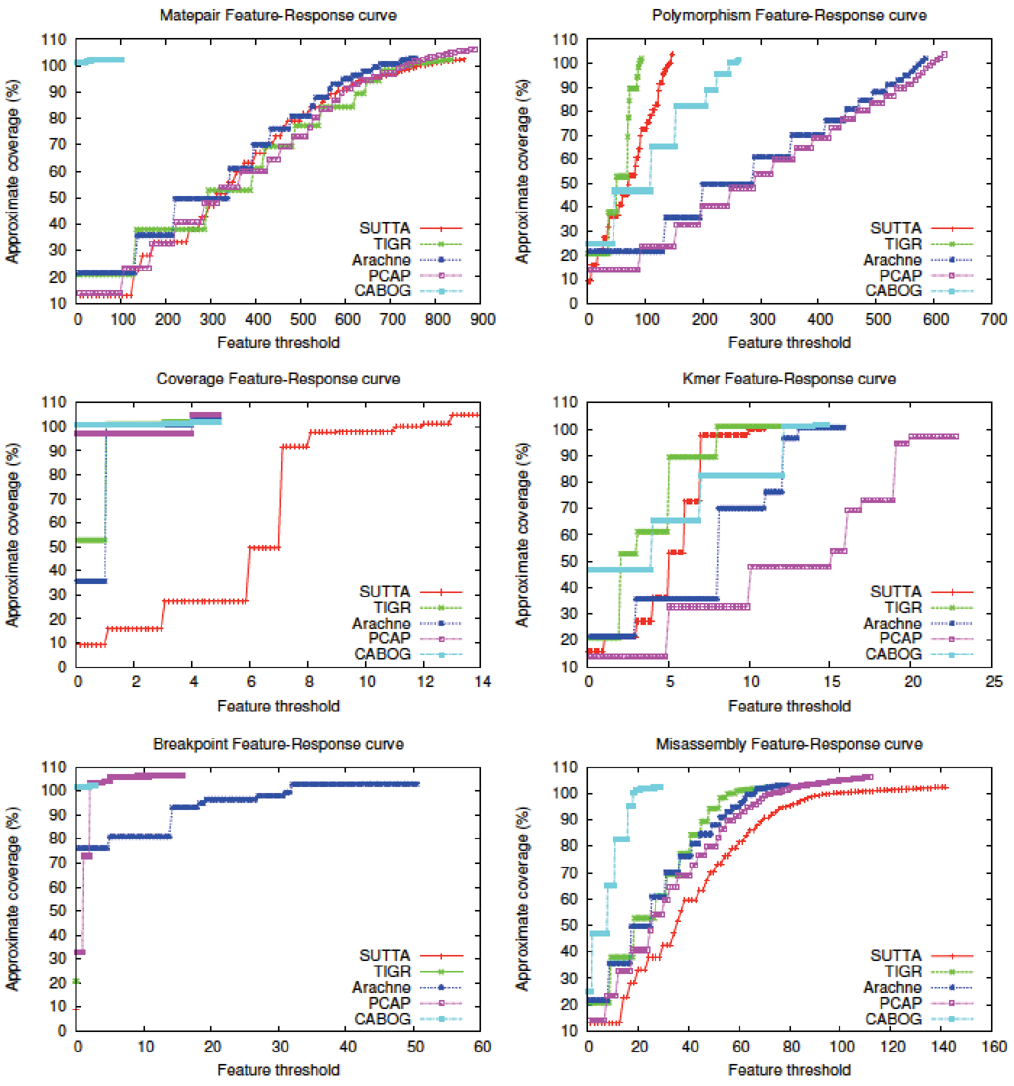
Feature-Response Curve: examples





A large experimental analysis

Narzisi and Mishra, **PLoS ONE** 2011



Matepair Feature-Response curve

Polymorphism Feature-Response curve

Coverage Feature-Response curve

Kmer Feature-Response curve

Breakpoint Feature-Response curve

Misassembly Feature-Response curve

- **7 different genomes** (Bacterial and Human).
- **Simulated** and **real** data.
- **16 different sequence assemblers** (both for old Sanger and next-generation Illumina sequencing technology).
- **All the generally accepted assembly paradigms** (Greedy, OLC, SBH, Seed-and-Extend, and B&B).

⇓

Quality and performance of the existing assemblers varies dramatically!

Conclusion

- Sequencing data are growing faster than computing power (*data tsunami*)
  - Parallelization is required!
- Genome assembly is not a solved problem!
  - We are dealing with a *wicked* problem.
- Be especially cautious about the absence of a particular sequence or gene.
  - Assembly artifact rather than a genuine lineage-specific deletion.
- *Every scientist should be skeptic of analyses performed at genome-wide scale using assembly techniques, and must critically examine any conclusions*

OPEN PROBLEMS

More challenges…

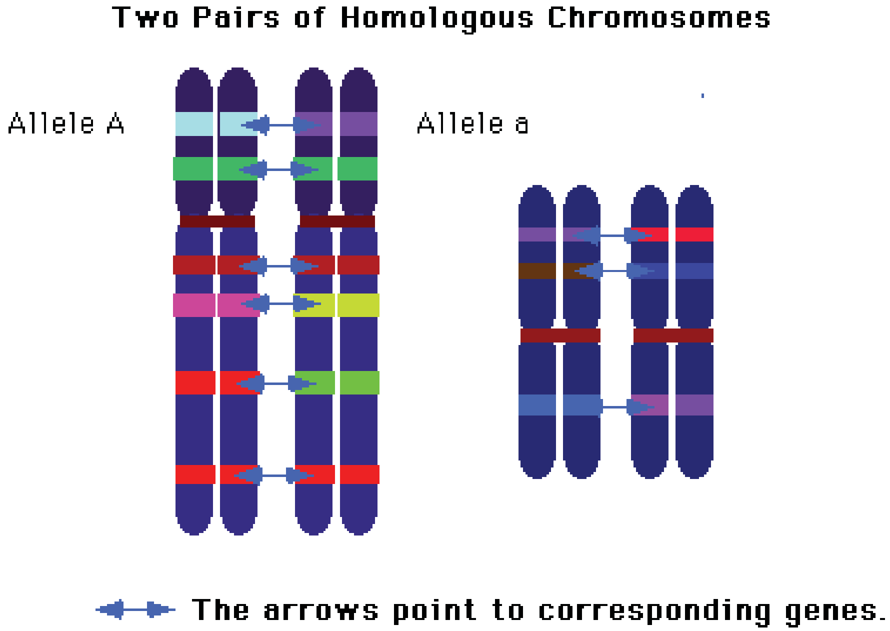
Variant discovery

De Novo mutations in exome sequencing

- Are there *de novo* mutation hot spot in the genome?
- What is the best way to distinguish between pathogenic and non-pathogenic mutations?
- How to deal with error in the reads?
- Exome sequencing of patient trios (father-mother-child)
  - Can assembly techniques be used in this case?
  - Concurrent assembly of three individuals?

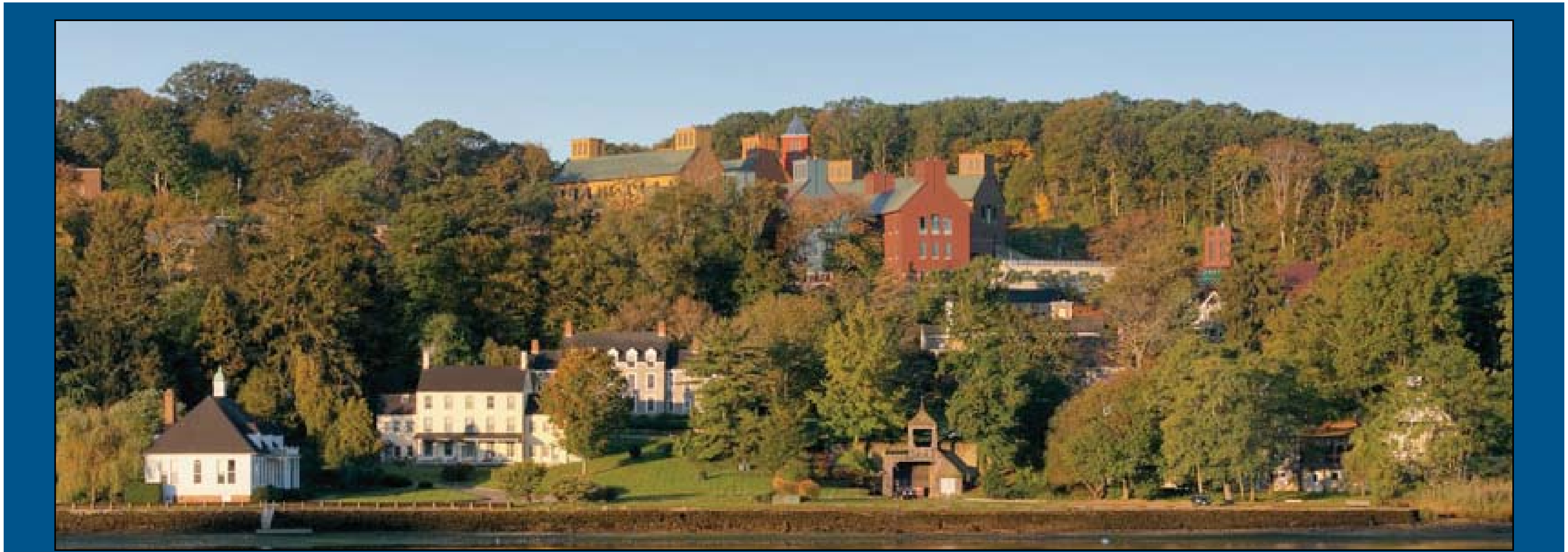
Haplotypic assembly

- **Haplotypic structure**: human cells have two homologous copies of each chromosome (except for the sex chromosomes X and Y), one from the mother and one from the father.
- **Problem**: current sequencing technologies do not distinguish between the two strands and the two homologous copies.
- We need better methods to disambiguate **bubble types**:
  - Sequencing error?
  - Haplotypic variation?
  - Homologous repeats?


Two Pairs of Homologous Chromosomes
Allele A
Allele a
The arrows point to corresponding genes.

Acknowledgment

- Prof. **Bud Mishra** (NYU)
- Prof. **Michael C. Schatz** (CSHL)
- Engg. **Fabian Menges** (NYU)
  - TotalReCaller/SUTTA
- Dr. **Andreas Witzel** (NYU)
  - SUTTA
- **Francesco Vezzi** (Applied Genomics Institute)
  - FRCurve feature analysis



THE END

Thank you

Email: gnarzisi@cshl.edu