SPAdes: a New Genome Assembler for Single-Cell Sequencing

Algorithmic Biology Lab
St. Petersburg Academic University

A. Bankevich, S. Nurk, D. Antipov, A.A. Gurevich, M. Dvorkin, A.S. Kulikov, V.M. Lesin, S.I. Nikolenko, S. Pham, A.D. Prjibelski, A.V. Pyshkin, A.V. Sirotkin, N. Vyahhi, G. Tesler, M.A. Alekseyev, P.A. Pevzner

April 28, 2012

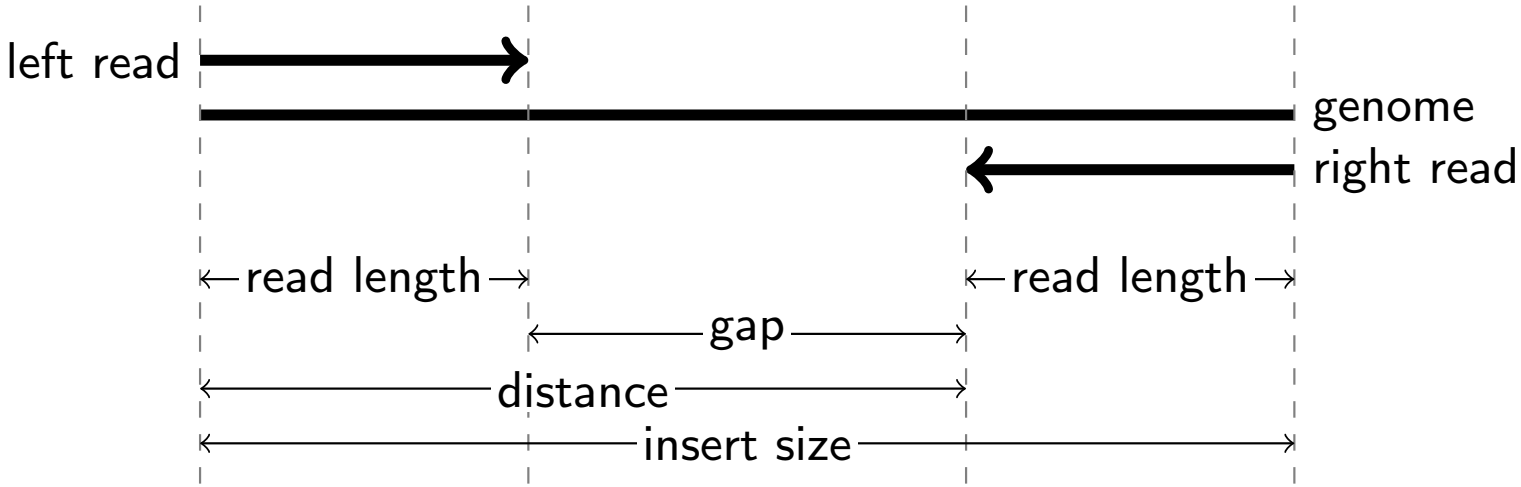
Outline

The assembly problem

- We have a long DNA sequence but cannot read it directly.
- Instead, we can perform *sequencing*, getting many random small pieces of the DNA (*reads*).
- For popular sequencing technologies, reads are usually $L \approx$ 35-400bp long.
- The assembly problem is to reconstruct the original sequence from the set of reads.

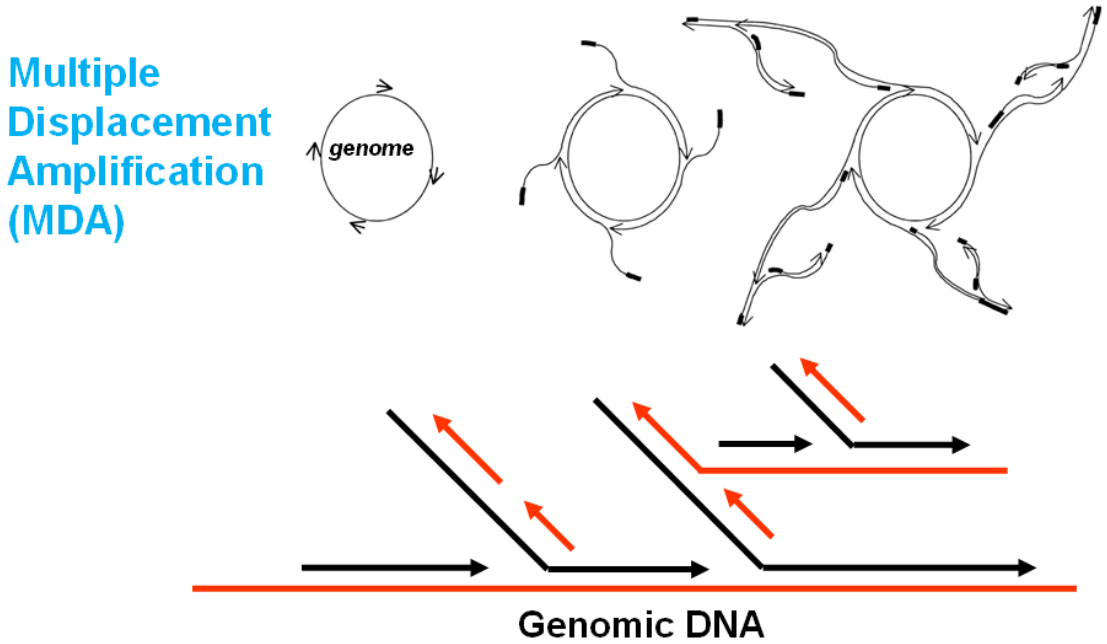
The assembly problem

- In many sequencing methods, we read $\approx L$ nucleotides from *both* ends of a longer DNA fragment.
- This leads to *paired-end reads* (read pairs): two reads for which we know the distance between them (Chaisson et al., 2009).



MDA technology

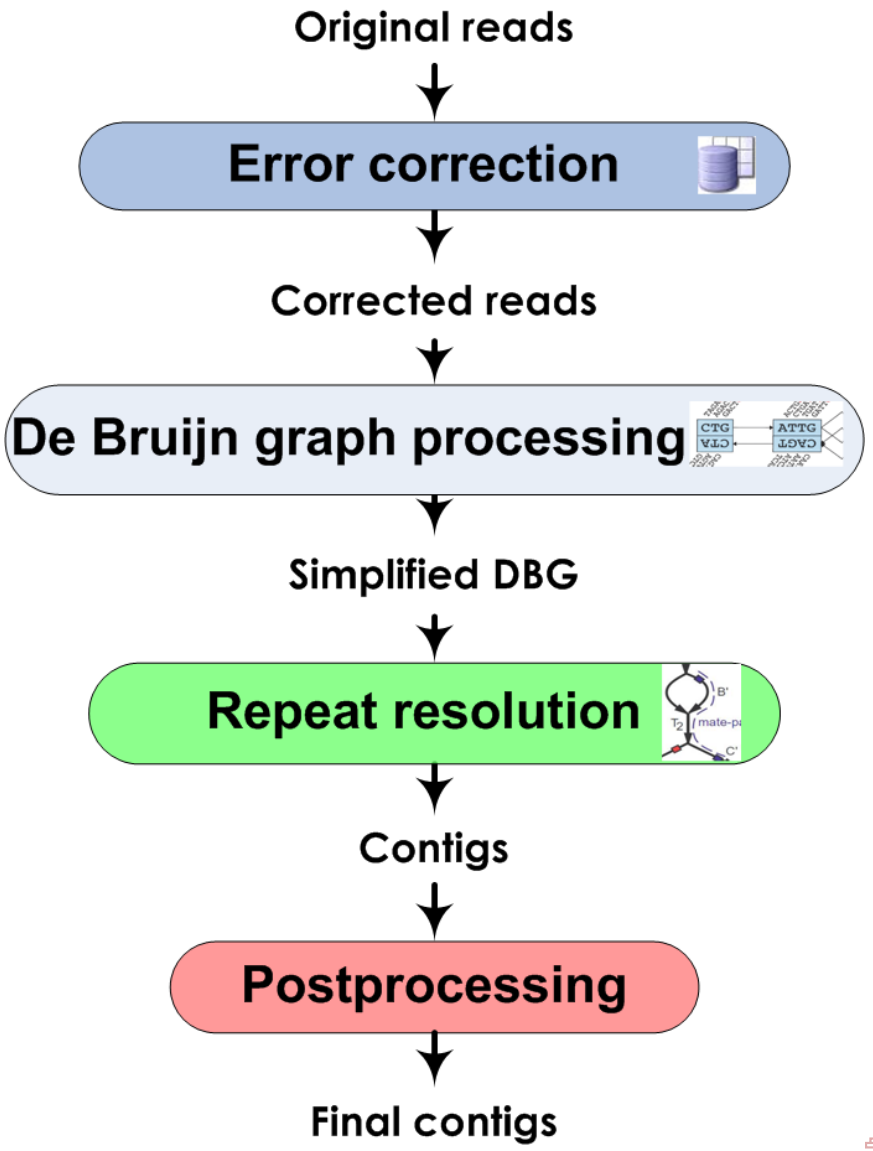
- Recent single-cell sequencing projects use the MDA technology (Chitsaz et al., 2011; Rodrigue et al., 2009)



MDA technology

- Computational problems of MDA:
  1. highly non-uniform coverage;
  2. more chimeric connections (sometimes with large coverage);
  3. more frequent errors.

Assembly pipeline



BayesHammer

- Basic idea: break reads into *k-mers* and study the set of *k*-mers.

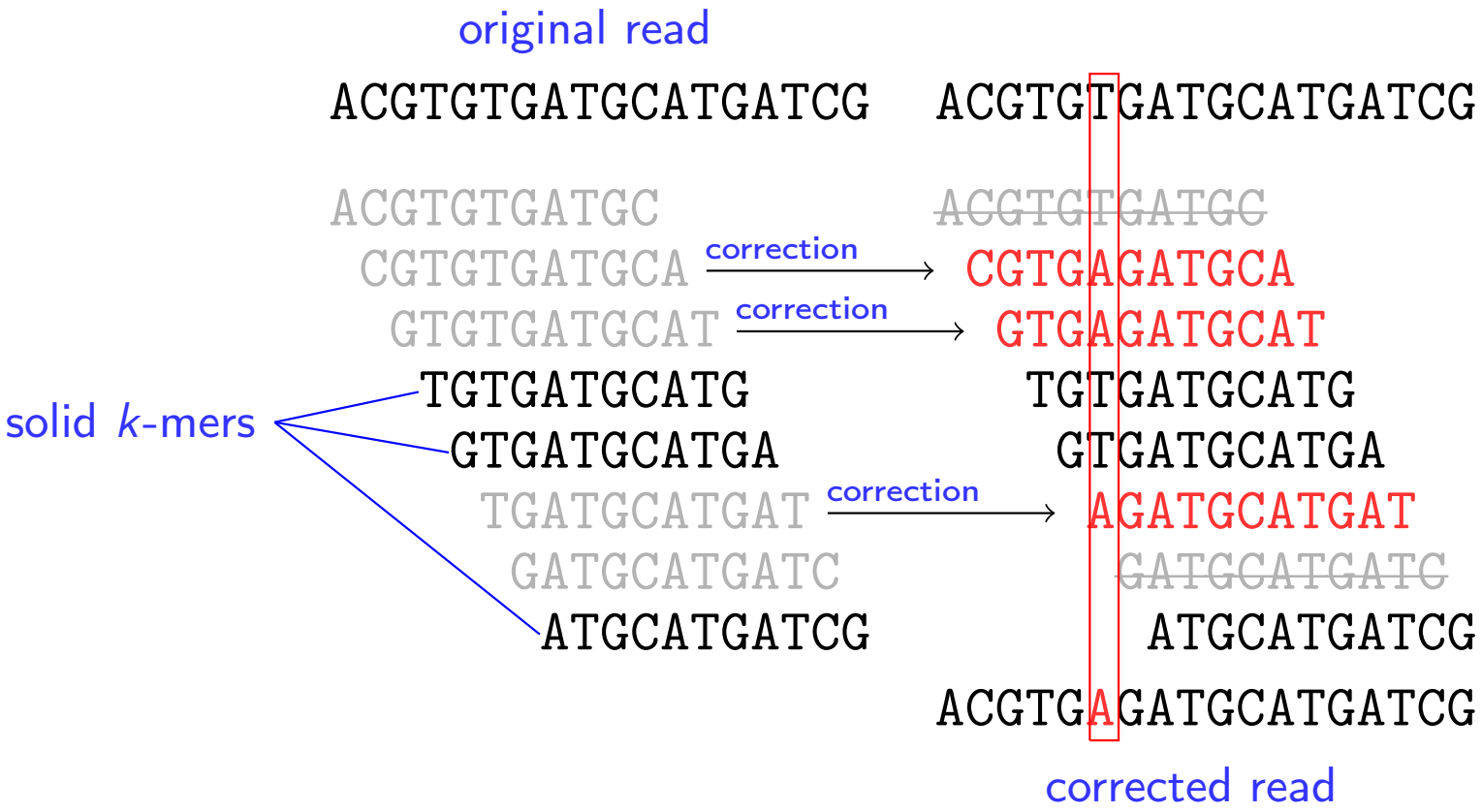
```
ACGTGTGATGCATGATCG

ACGTGTGATGC
 CGTGTGATGCA
  GTGTGATGCAT
   TGTGATGCATG
    GTGATGCATGA
     TGATGCATGAT
      GATGCATGATC
       ATGCATGATCG
```

- *k*-mers are the basic building block in modern assemblers (de Bruijn graph). However, de Bruijn graphs are not directly applicable because of errors in reads.

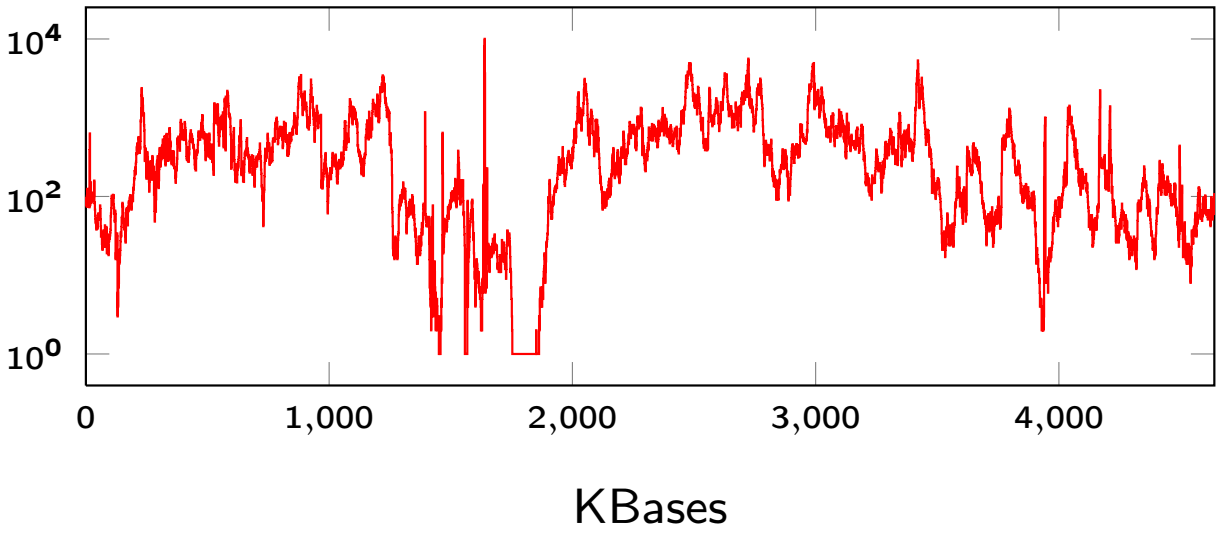
BayesHammer

- The first step in assembly is to fix as many errors as we can.
- If we know what k-mers are correct, it's easy to correct reads.



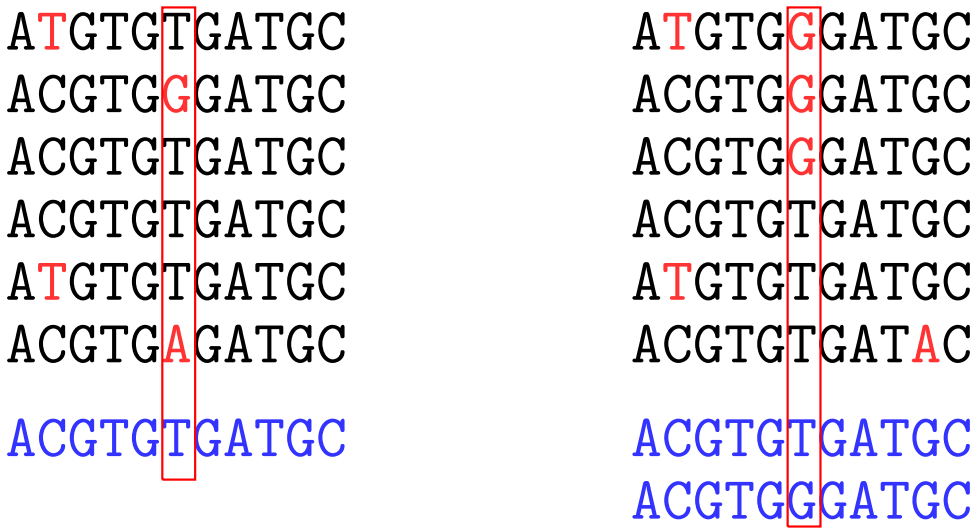
BayesHammer

- In regular (multi-cell) error correction, we can look at *how many copies* of a k-mer there are and assume that rare k-mers represent errors.
- In single-cell datasets, this idea fails due to non-uniform coverage.
- SPAdes (and BayesHammer) introduces many novel ideas to handle the non-uniform case.



BayesHammer

- Basic idea: a k-mer is covered many times (even if non-uniformly).
- Thus, if we look at the set of *similar* k-mers we can find out what to do because nucleotides of the central k-mer will be better covered than others.
- Problem: there may be several centers in such a cluster.

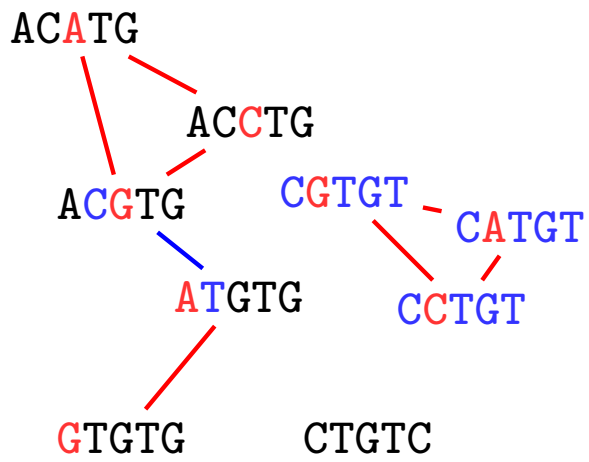


BayesHammer

- In Hammer, Medvedev et al. (2011) constructed and roughly clustered the Hamming graph of k-mers; BayesHammer uses probabilistic subclustering to get multiple centers in a cluster.

Reads	k**-mers**
ACGTGTG	ACGTG CGTGT GTGTG
ACATGTG	ACATG CATGT ATGTG
ACCTGTC	ACCTG CCTGT CTGTC

Hamming graph



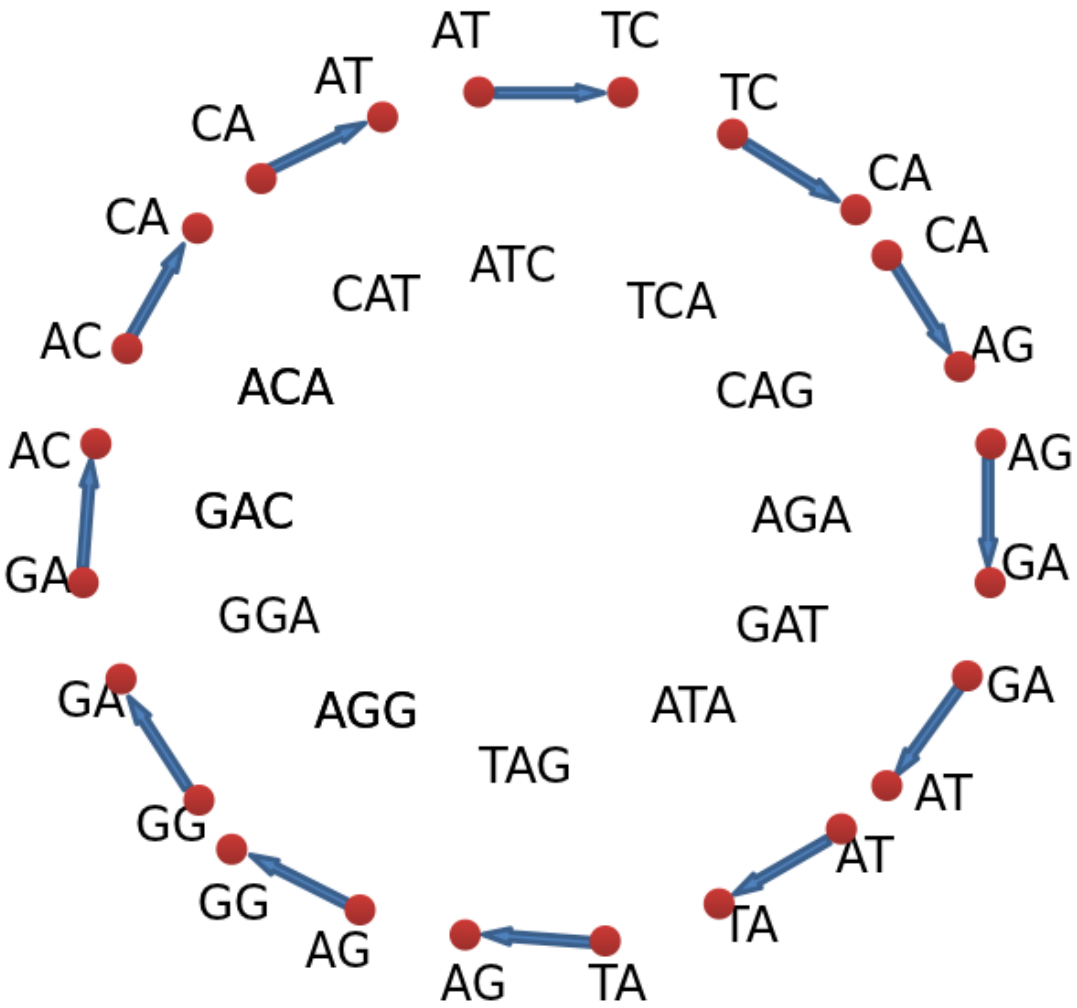
- As a result, BayesHammer corrects single-cell datasets much better than other tools.

Outline

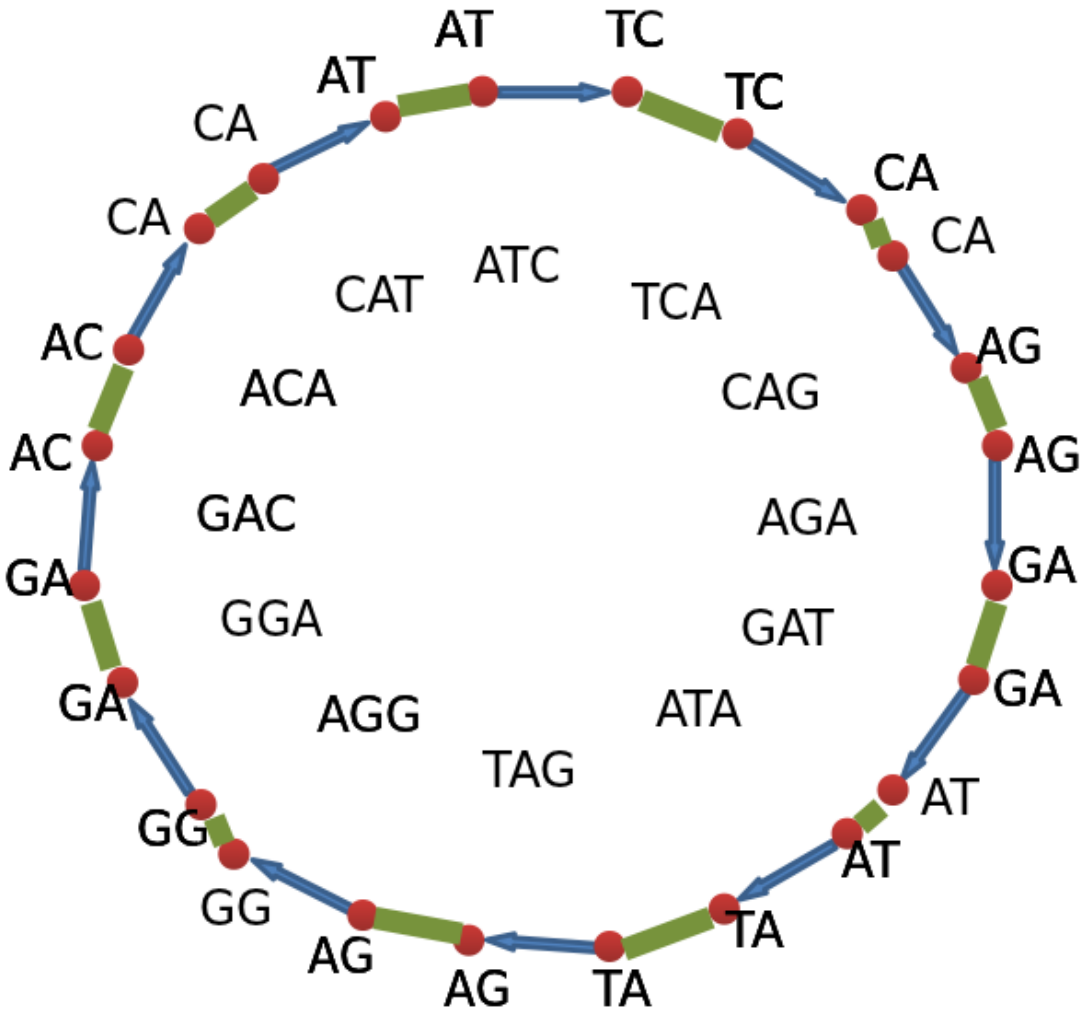
De Bruijn graphs

- When there are relatively few k-mers left, we can begin global processing (the actual assembly).
- Basic idea: de Bruijn graph (Idury & Waterman, 1995; Pevzner et al., 2001).

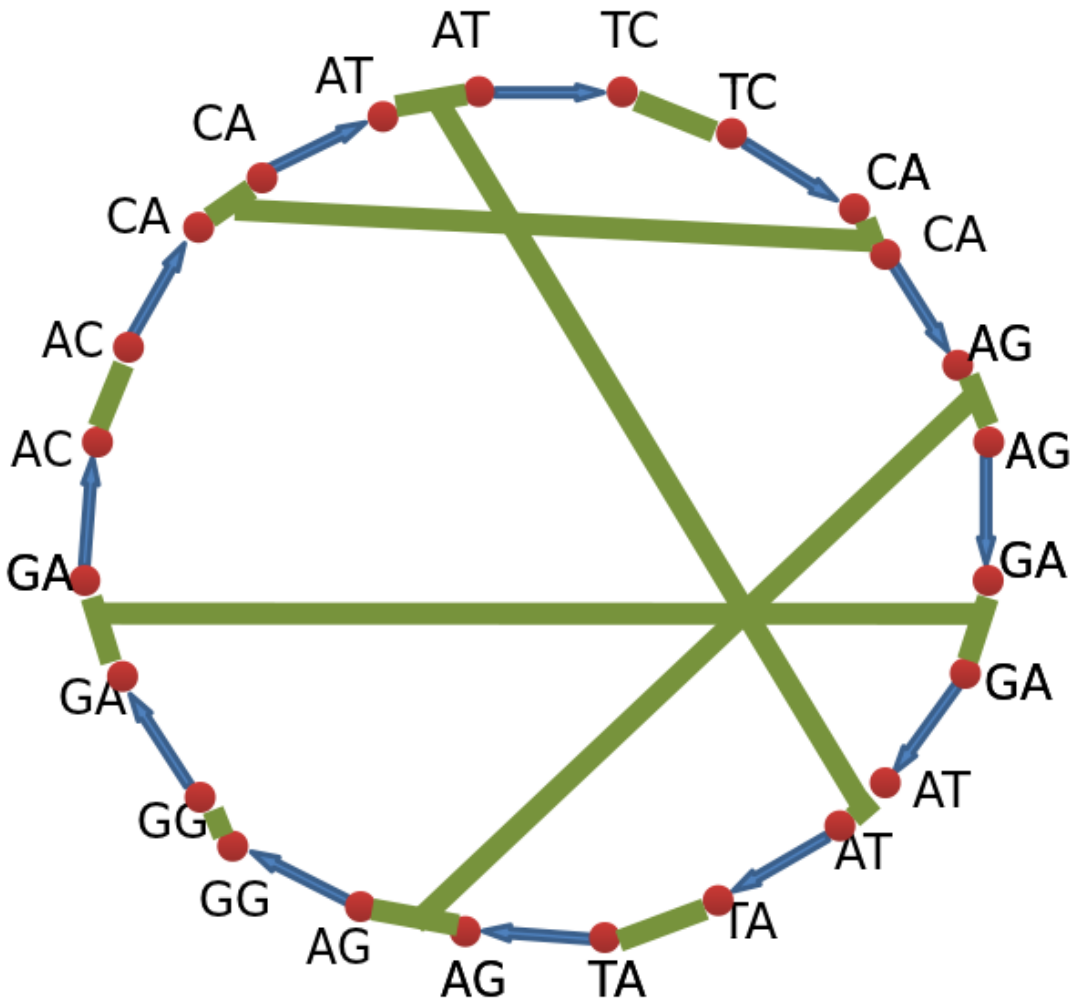
De Bruijn graphs



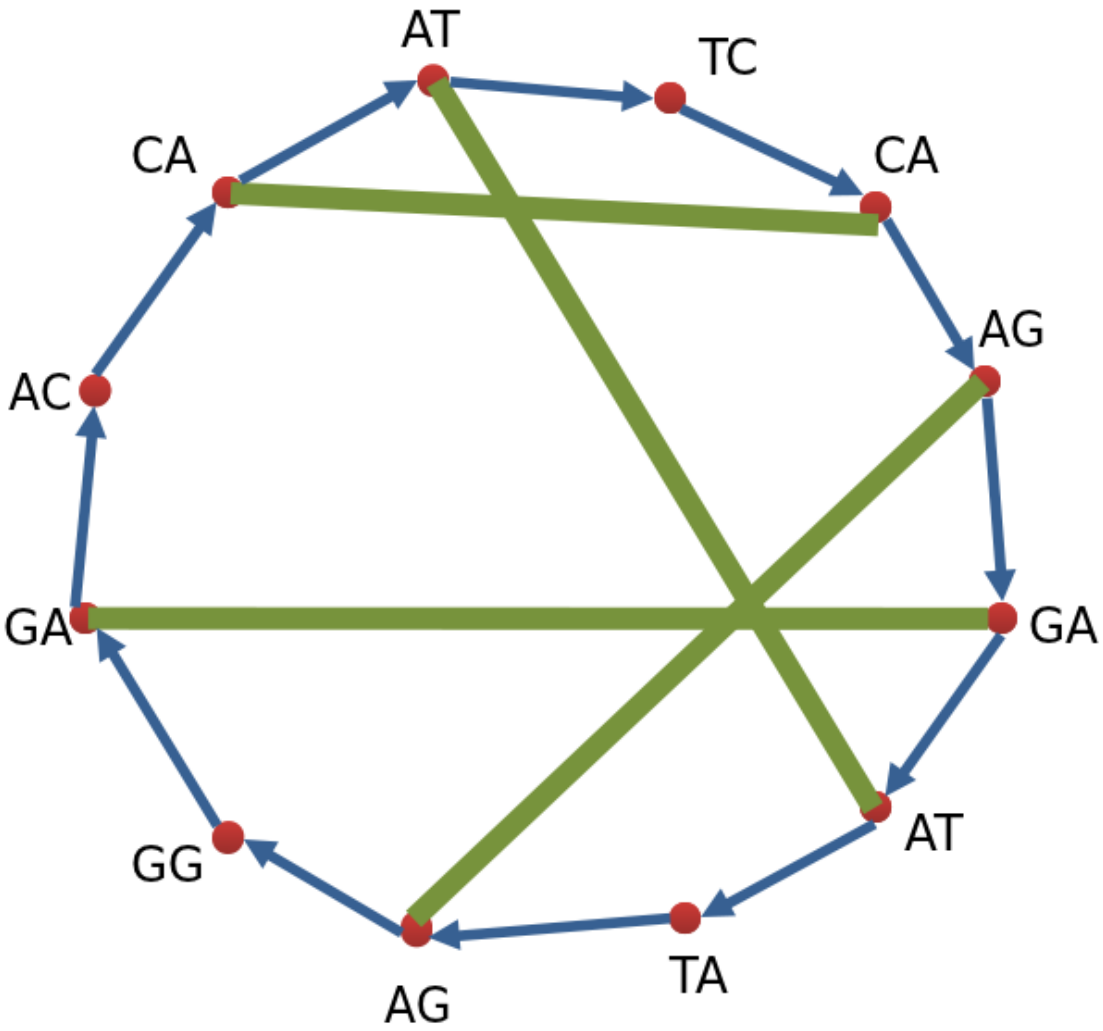
De Bruijn graphs



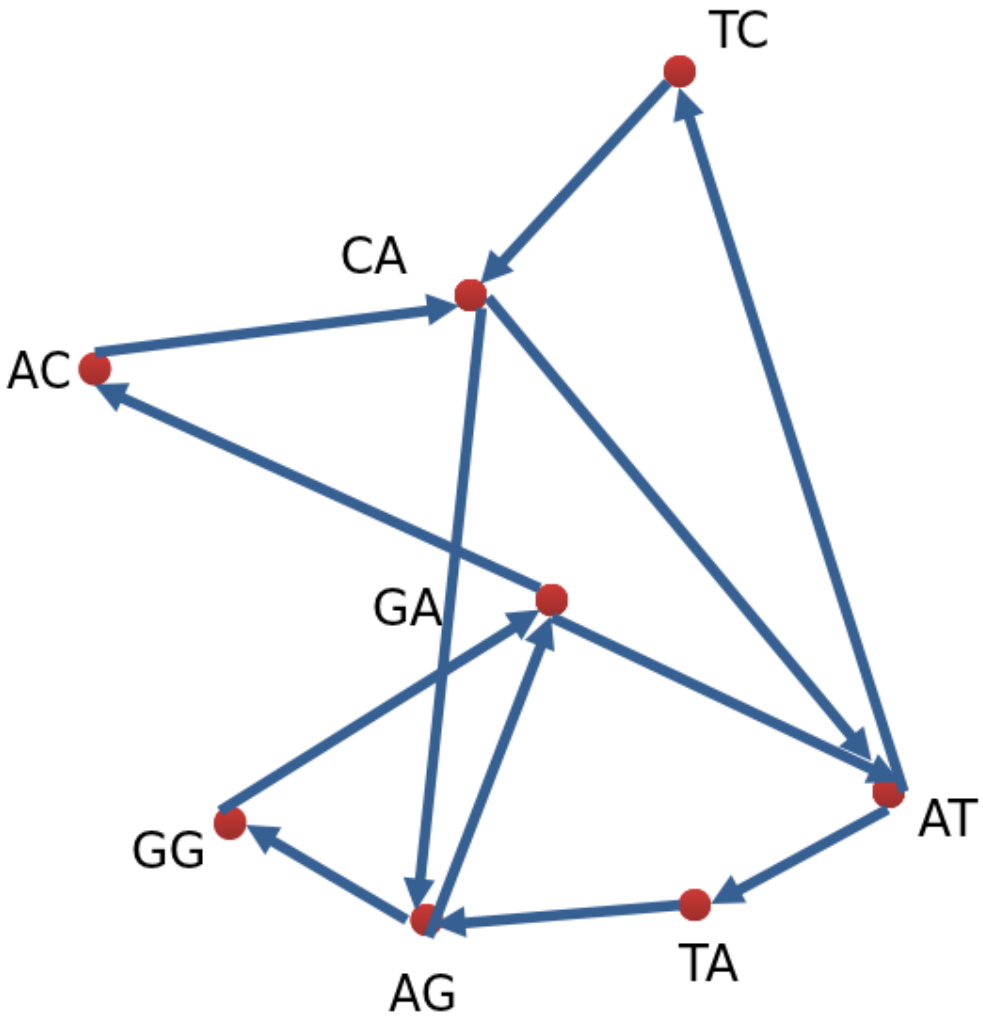
De Bruijn graphs



De Bruijn graphs



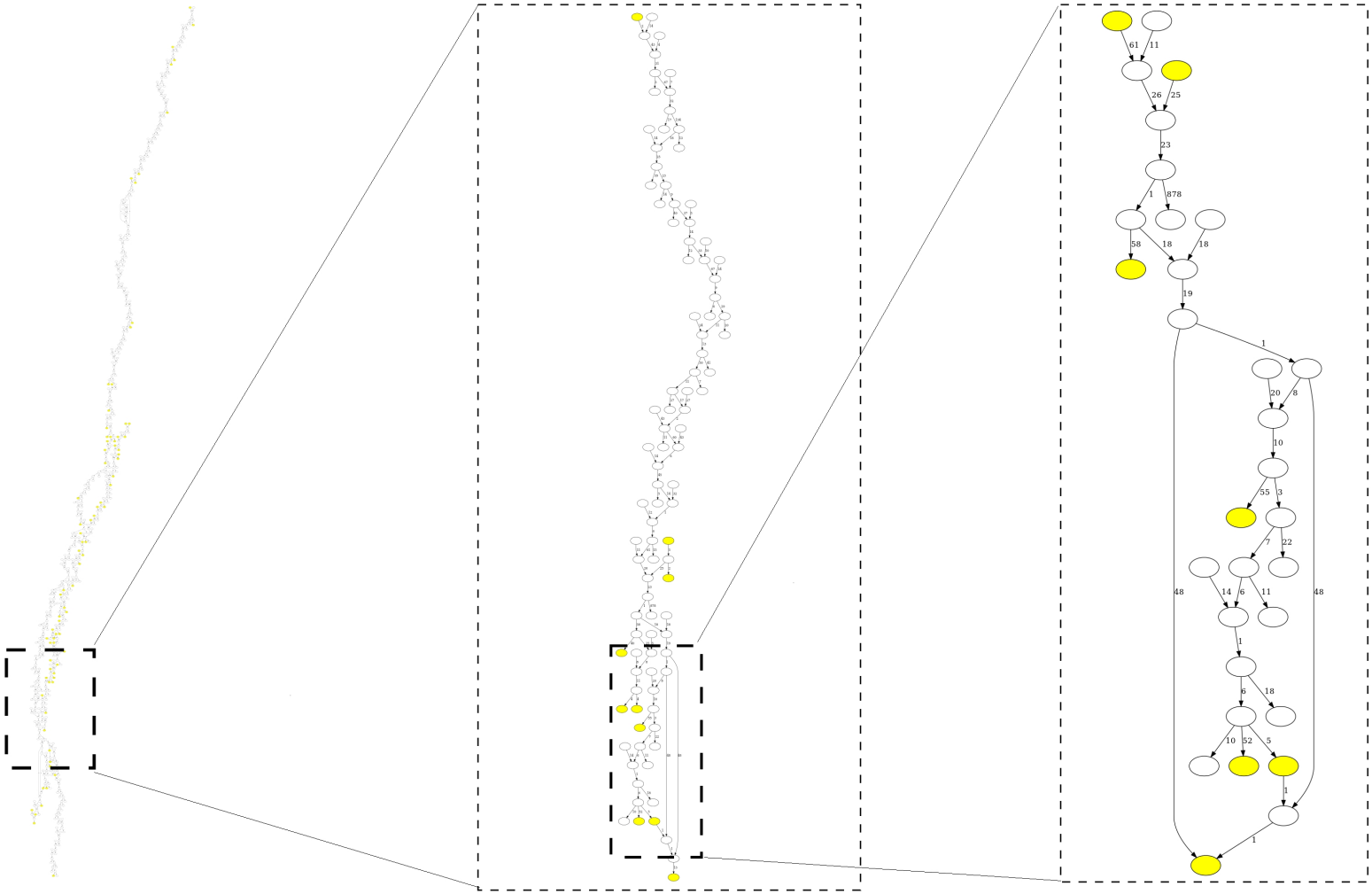
De Bruijn graphs



De Bruijn graph and errors

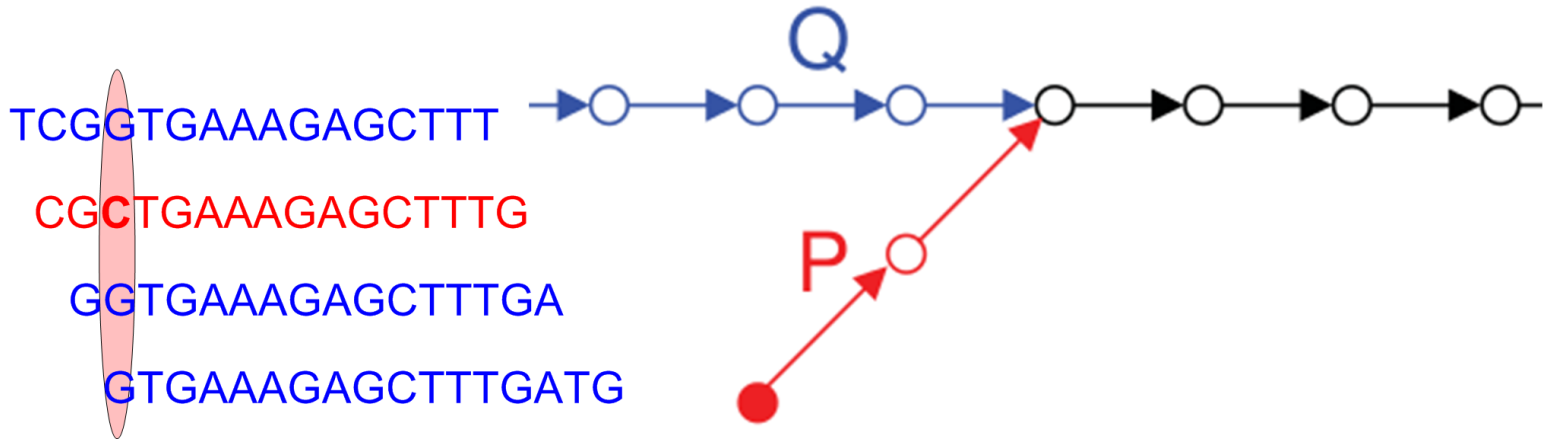
- Now, if there is a single string uniting all k-mers, it corresponds to a Eulerian cycle/path in this graph.
- However, due to sequencing errors and repeats we cannot just find a Eulerian cycle/path and think that we're done.
- In the presence of errors, this is a hard problem, and not very well defined.

De Bruijn graph before simplification



De Bruijn graph simplification: tips

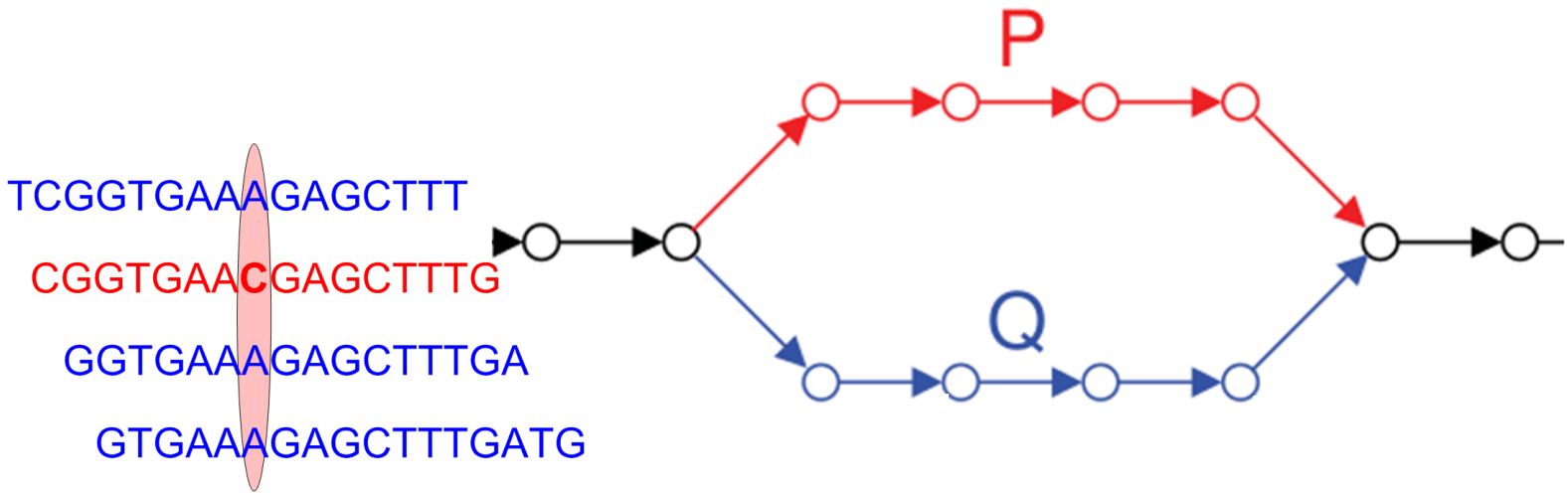
- There are three kinds of common errors in the de Bruijn graph. They all have additional complications in the single-cell case.
- A *tip* results from a single error close to the end of a read.



- SPAdes incorporates a tip clipping algorithm that is enhanced by gap closing (see below).

De Bruijn graph simplification: bulges

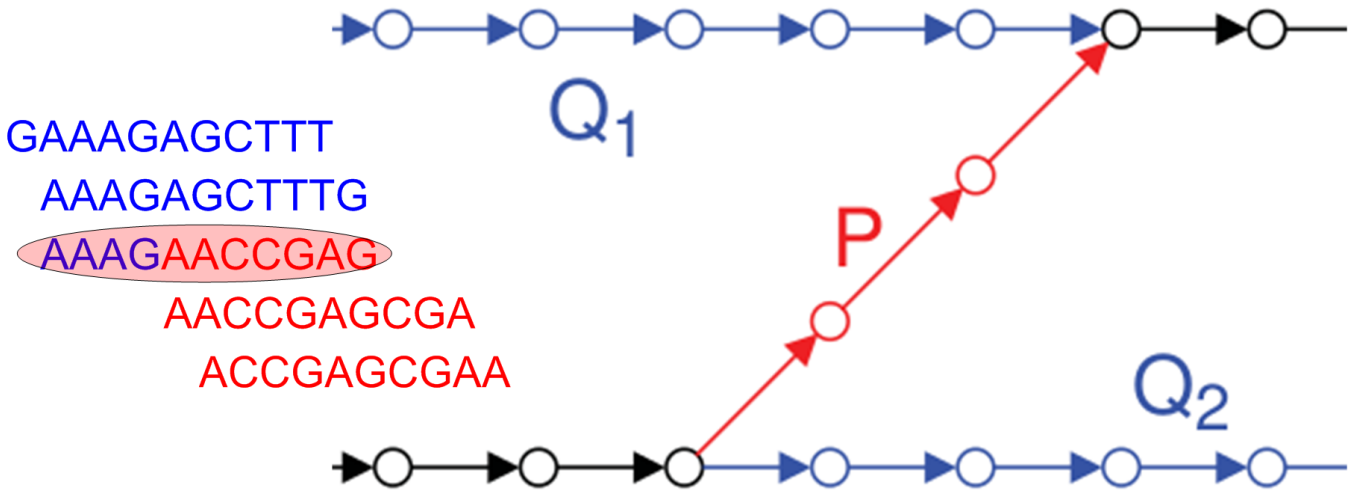
- There are three kinds of common errors in the de Bruijn graph. They all have additional complications in the single-cell case.
- A *bulge* occurs when the error is in the middle.



- SPAdes *projects* bulges with additional bookkeeping to preserve the original mappings – *bulge corremoval*.

De Bruijn graph simplification: chimeras

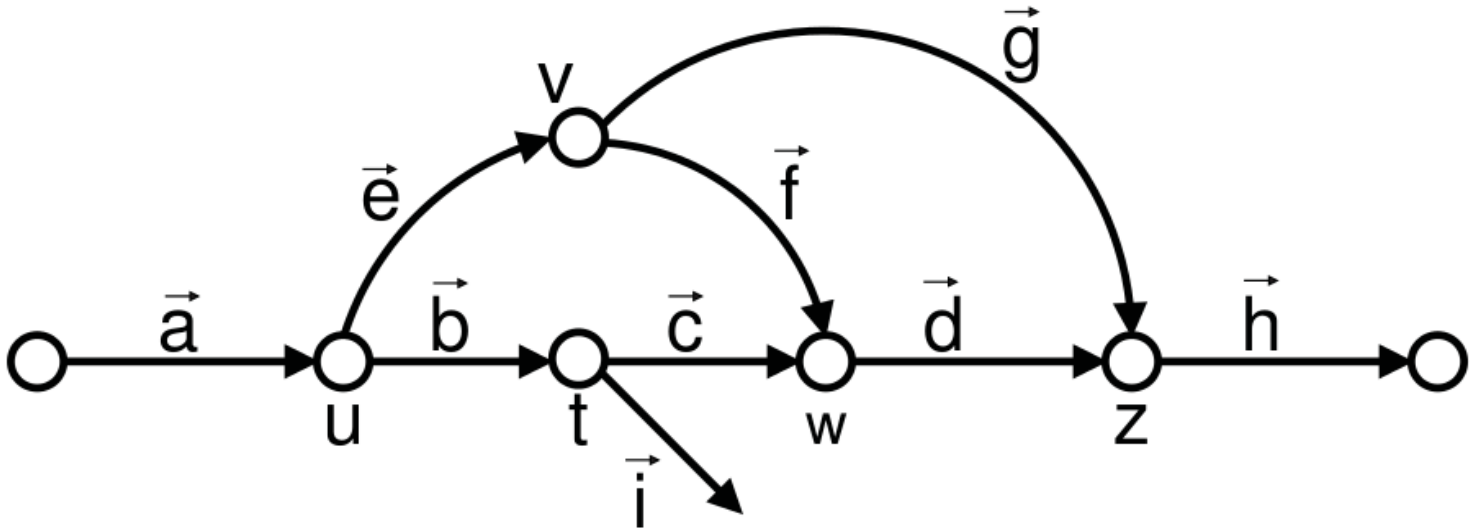
- There are three kinds of common errors in the de Bruijn graph. They all have additional complications in the single-cell case.
- A *chimeric connection* joins two unrelated parts of the graph.



- SPAdes uses a novel algorithm for removing chimeric connections based on max-flow graph algorithms.

De Bruijn graph simplification: scheduling

- There are three kinds of common errors in the de Bruijn graph. They all have additional complications in the single-cell case.
- And they are all usually superimposed in a single entangled mess.



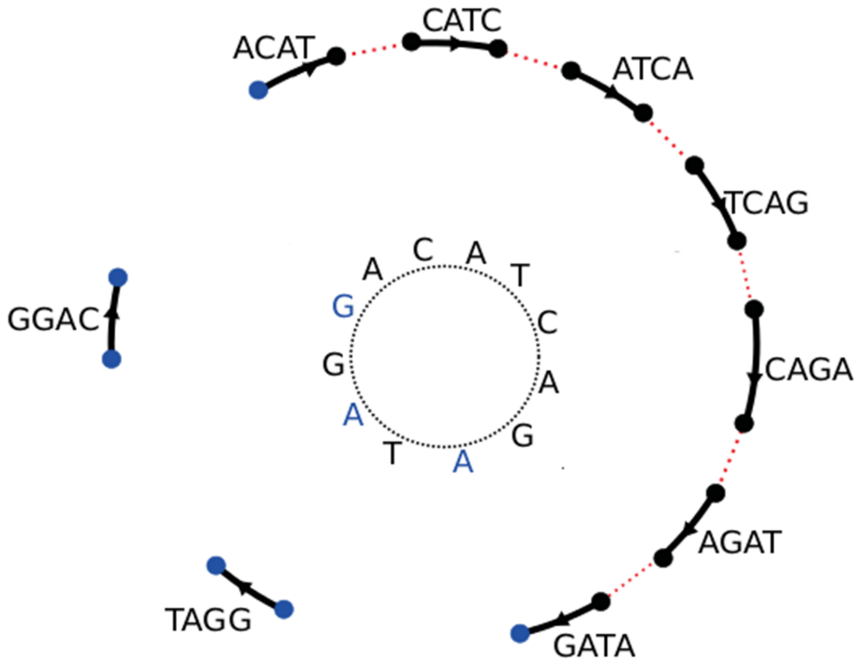
- SPAdes uses special scheduling algorithms to process bulges and chimerics in an optimal order.

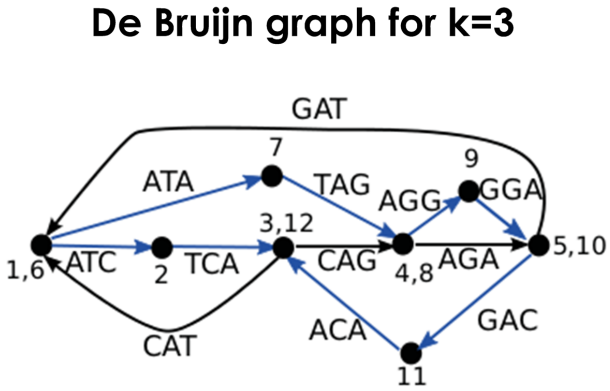
De Bruijn graph simplification

- To sum up: for single-cell datasets, errors are harder to correct because coverage is not a reliable way to recognize an error.
- We have developed several new algorithms that process errors based on graph structure rather than coverage alone.
- Besides, SPAdes corrects errors iteratively, in an order resulting from special scheduling policies.

Multisized de Bruijn graphs

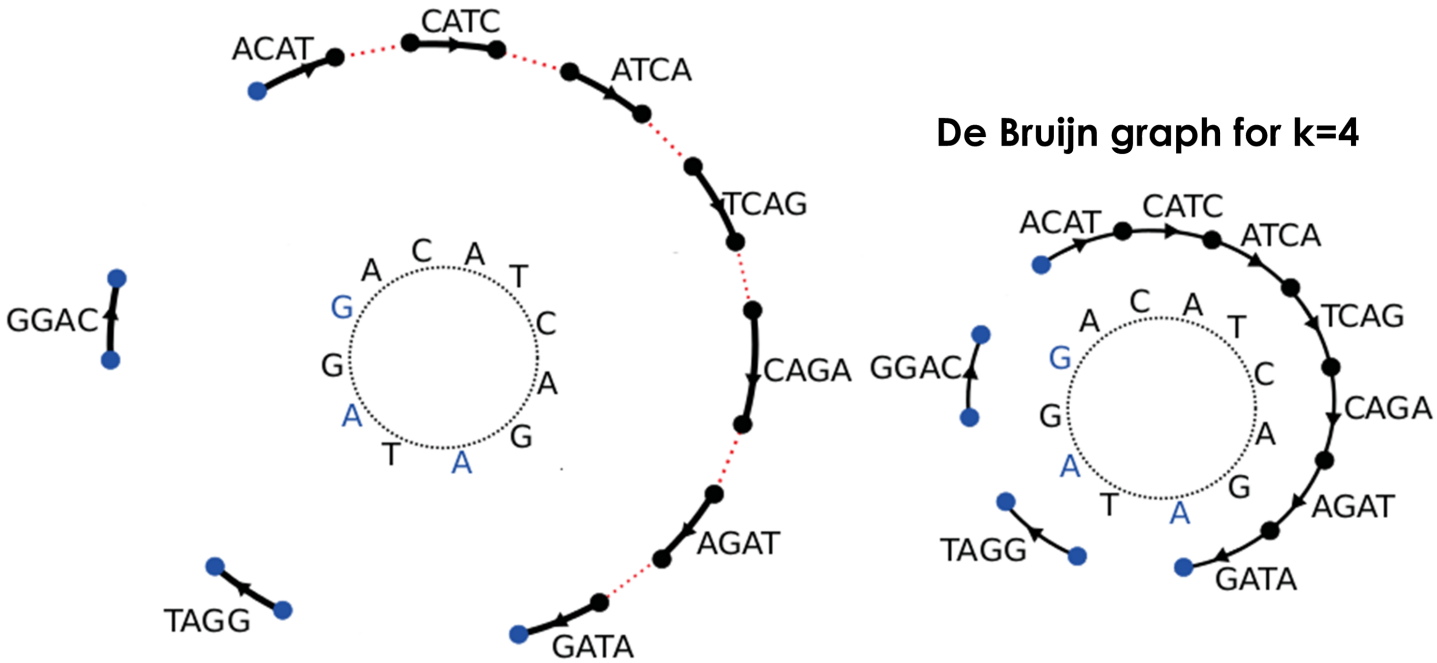
- Let us now think about the value of k. How does the graph change if k changes?
- For a small k, the graph is well-connected but very entangled.





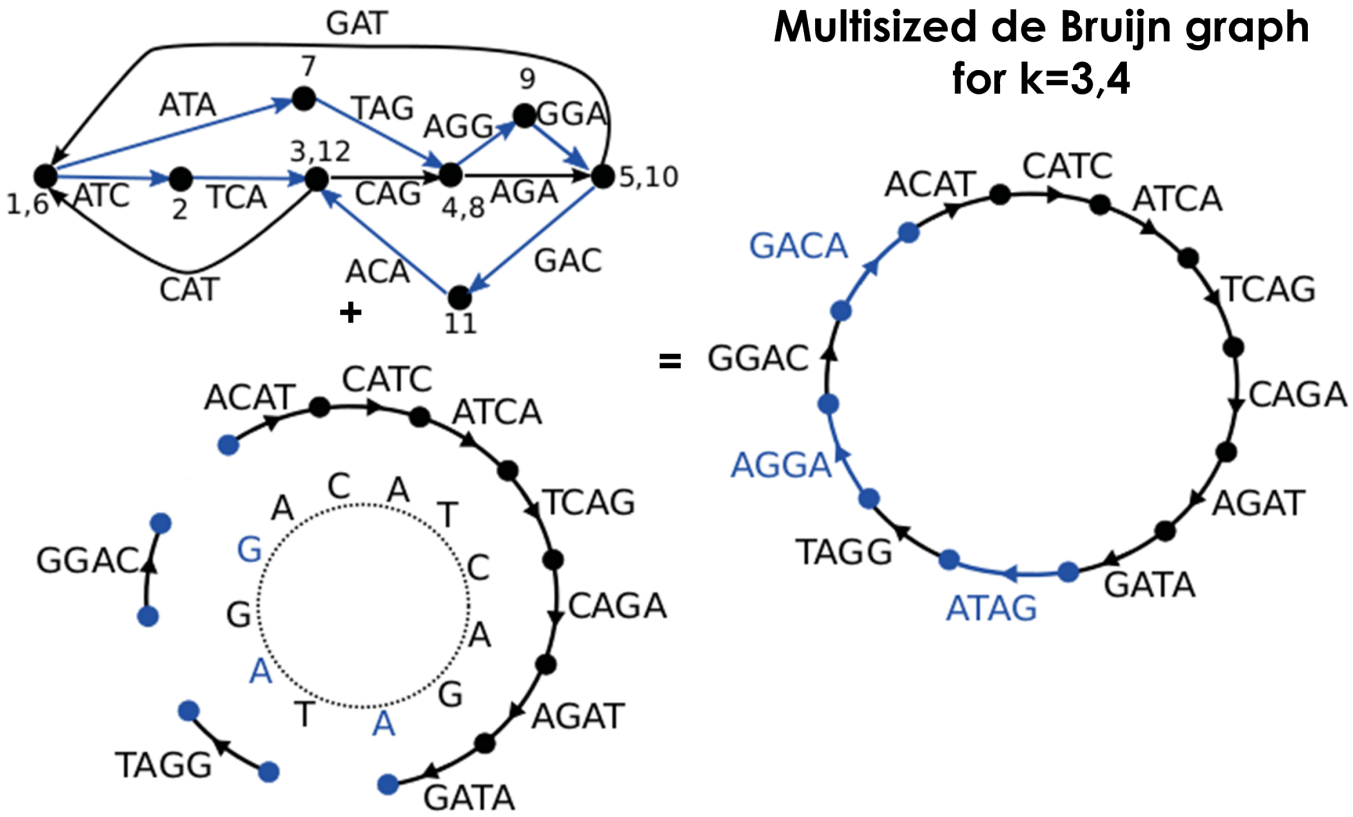
Multisized de Bruijn graphs

- Let us now think about the value of k. How does the graph change if k changes?
- For a large k, the graph is less entangled but may become disconnected.



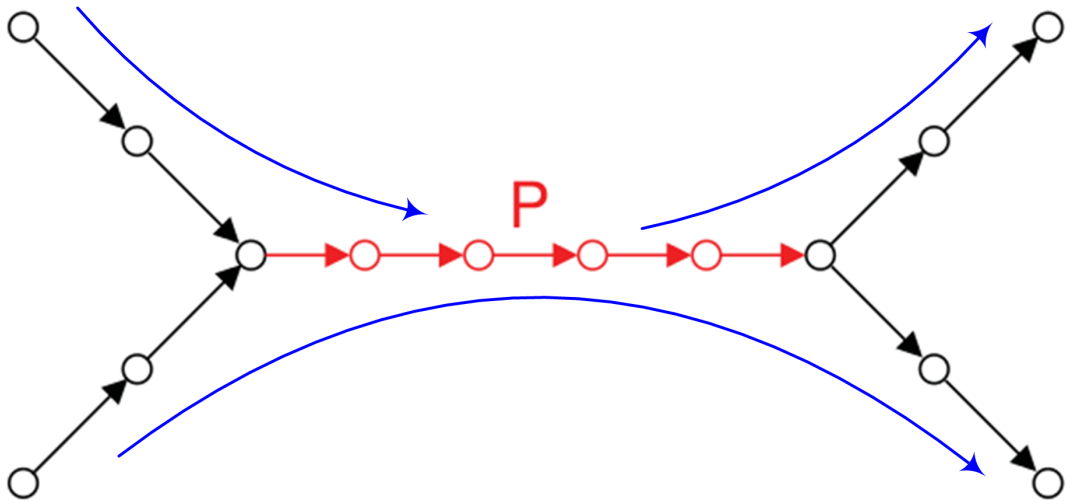
Multisized de Bruijn graphs

- Let us now think about the value of k. How does the graph change if k changes?
- SPAdes combines several different values of k to get the best of all worlds (see also IDBA (Peng et al., 2010)).



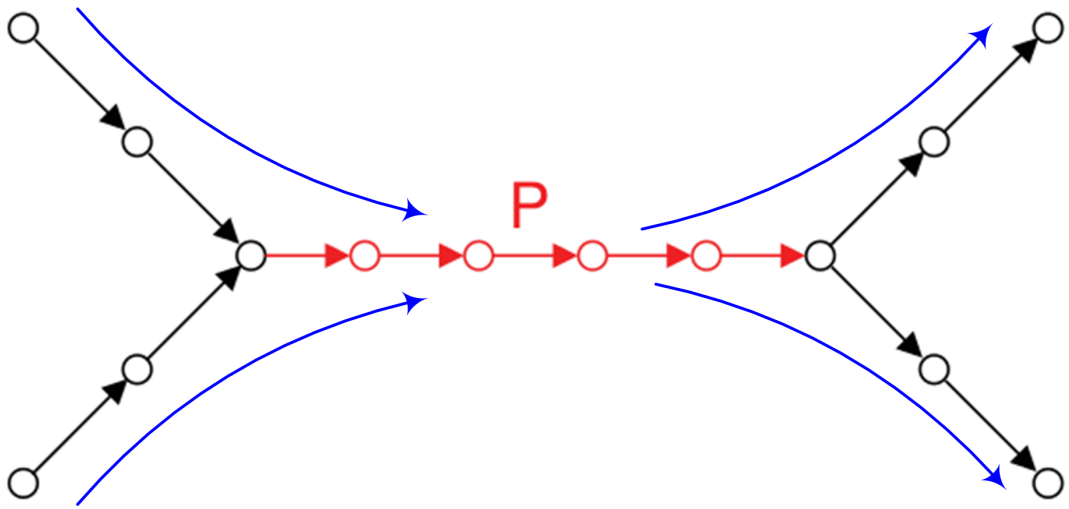
Repeats

- A *repeat* in the genome results in a path with multiple entrances and/or multiple exits. Sometimes they can be resolved with long reads; sometimes not, and they become *tangles*.
- This repeat is easy to handle.



Repeats

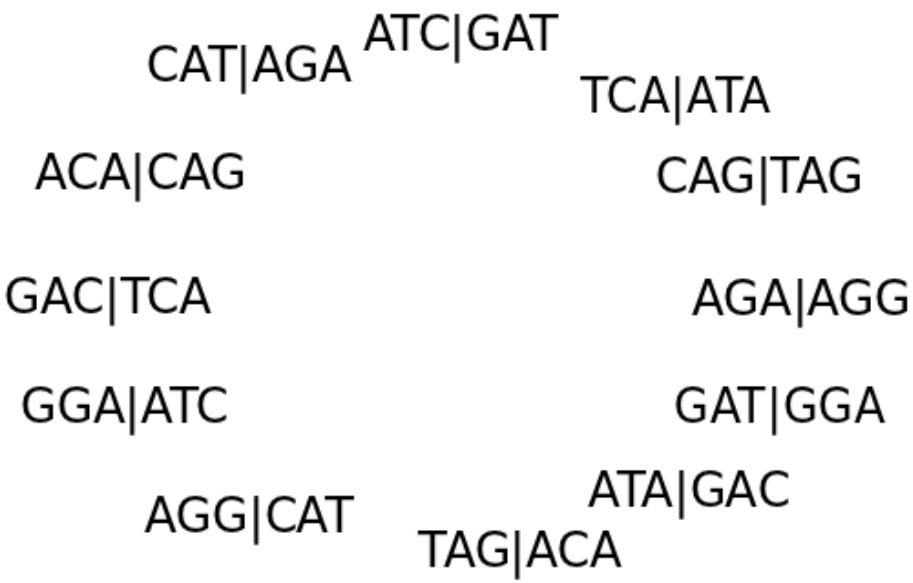
- A *repeat* in the genome results in a path with multiple entrances and/or multiple exits. Sometimes they can be resolved with long reads; sometimes not, and they become *tangles*.
- This one is not.



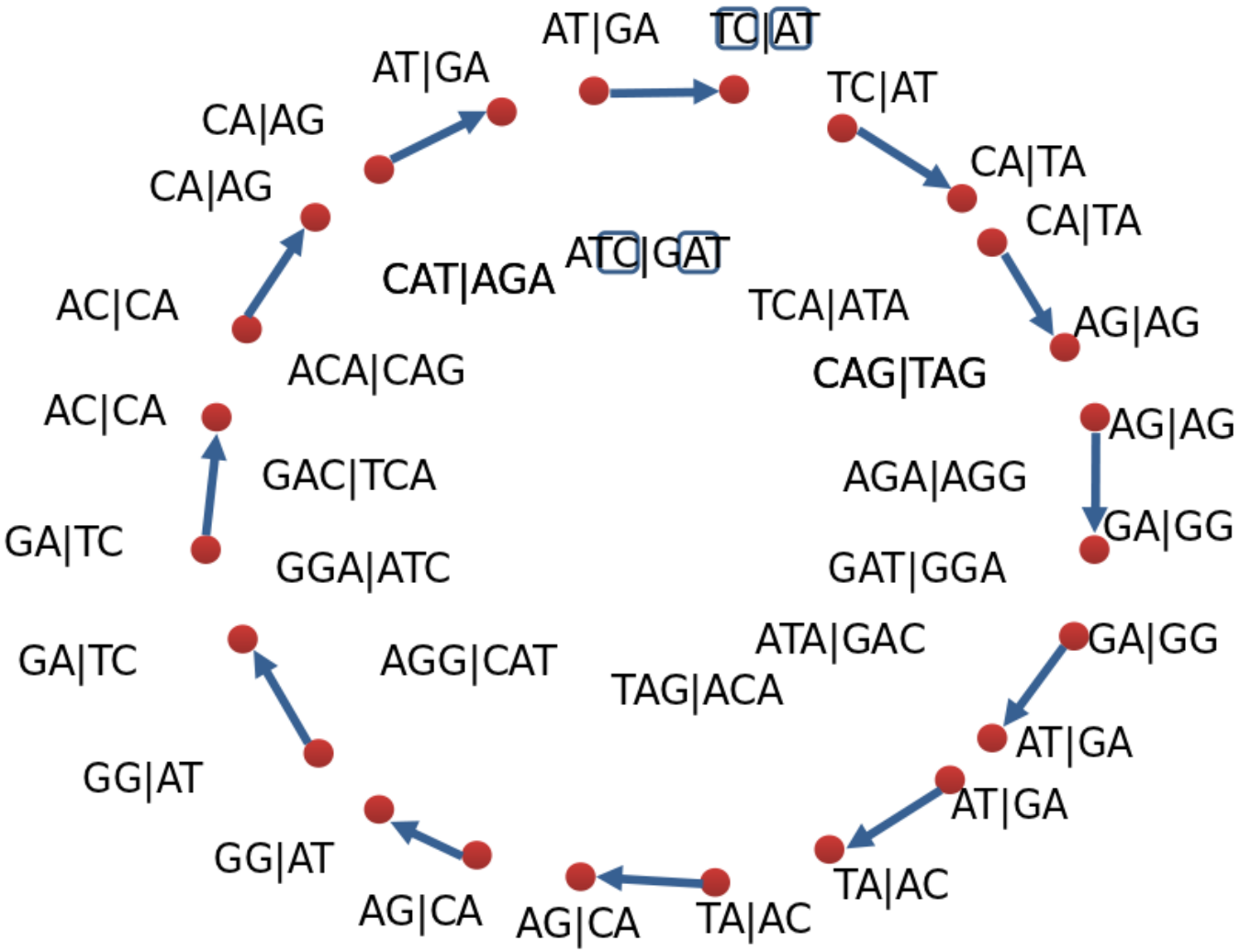
Paired reads and paired de Bruijn graphs

- Recall that reads come in pairs.
- We could construct a *paired de Bruijn graph* in which vertices correspond to pairs of k-mers (Medvedev et al., 2011). This graph is much sparser than a regular de Bruijn graph.

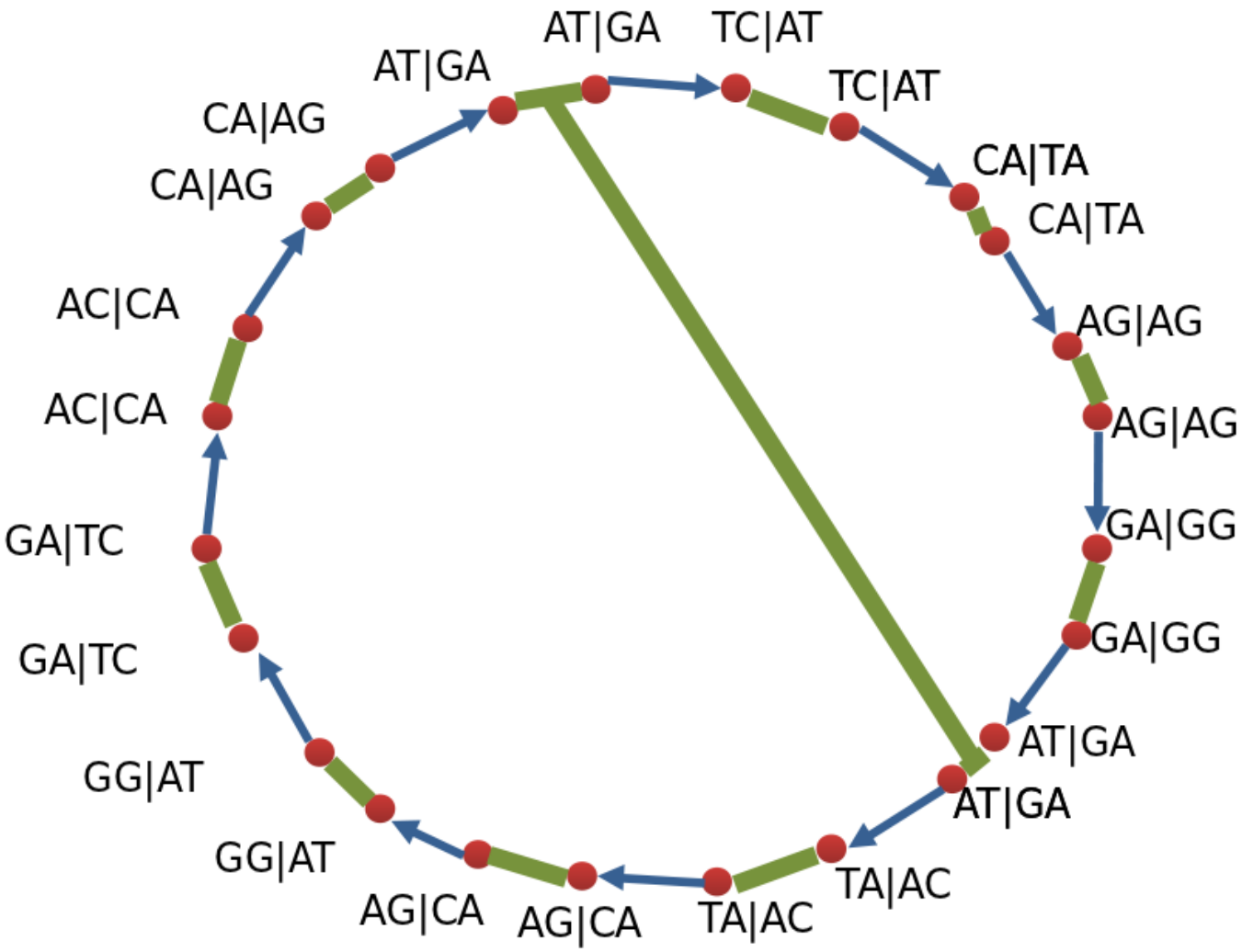
Paired reads and paired de Bruijn graphs



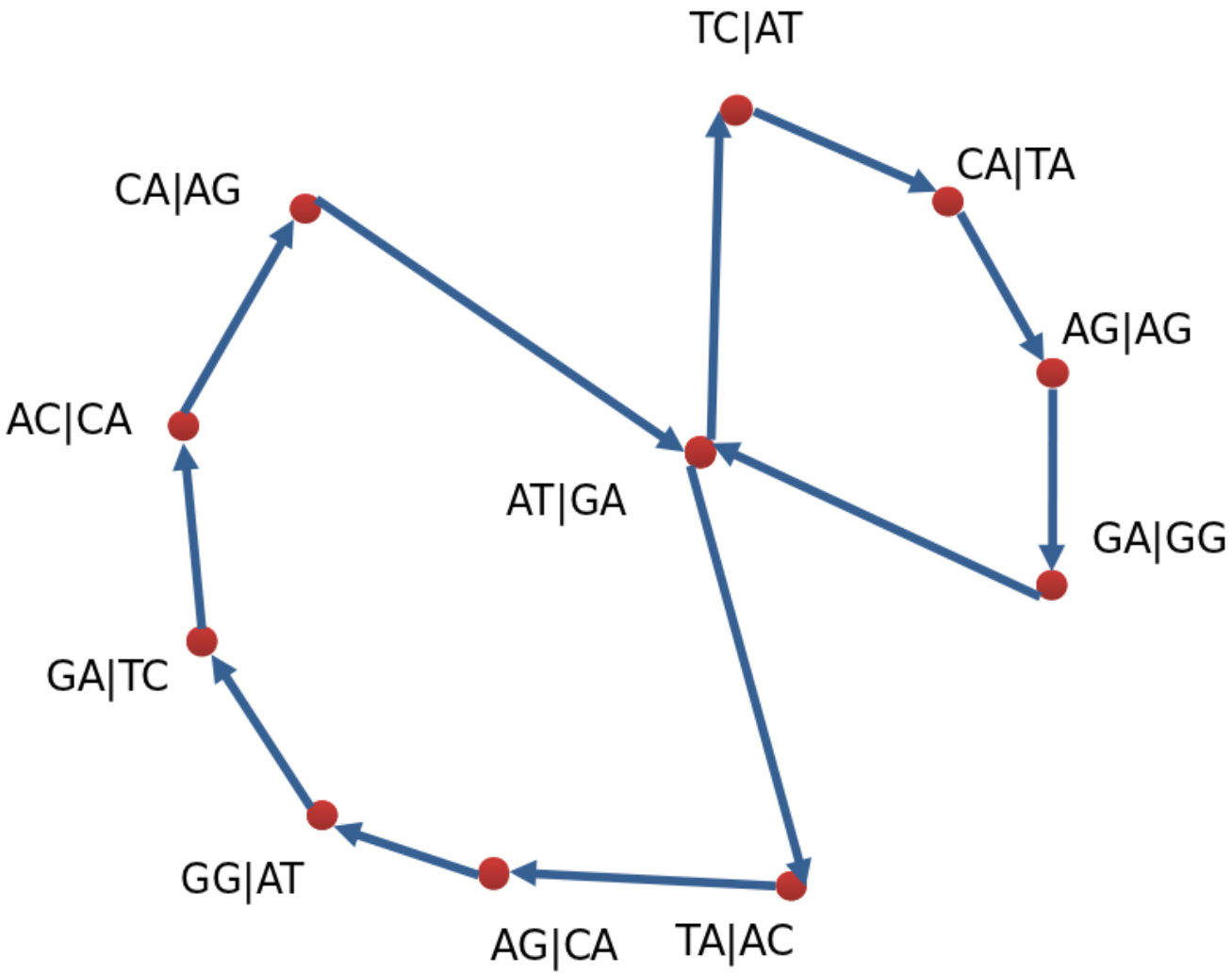
Paired reads and paired de Bruijn graphs



Paired reads and paired de Bruijn graphs



Paired reads and paired de Bruijn graphs

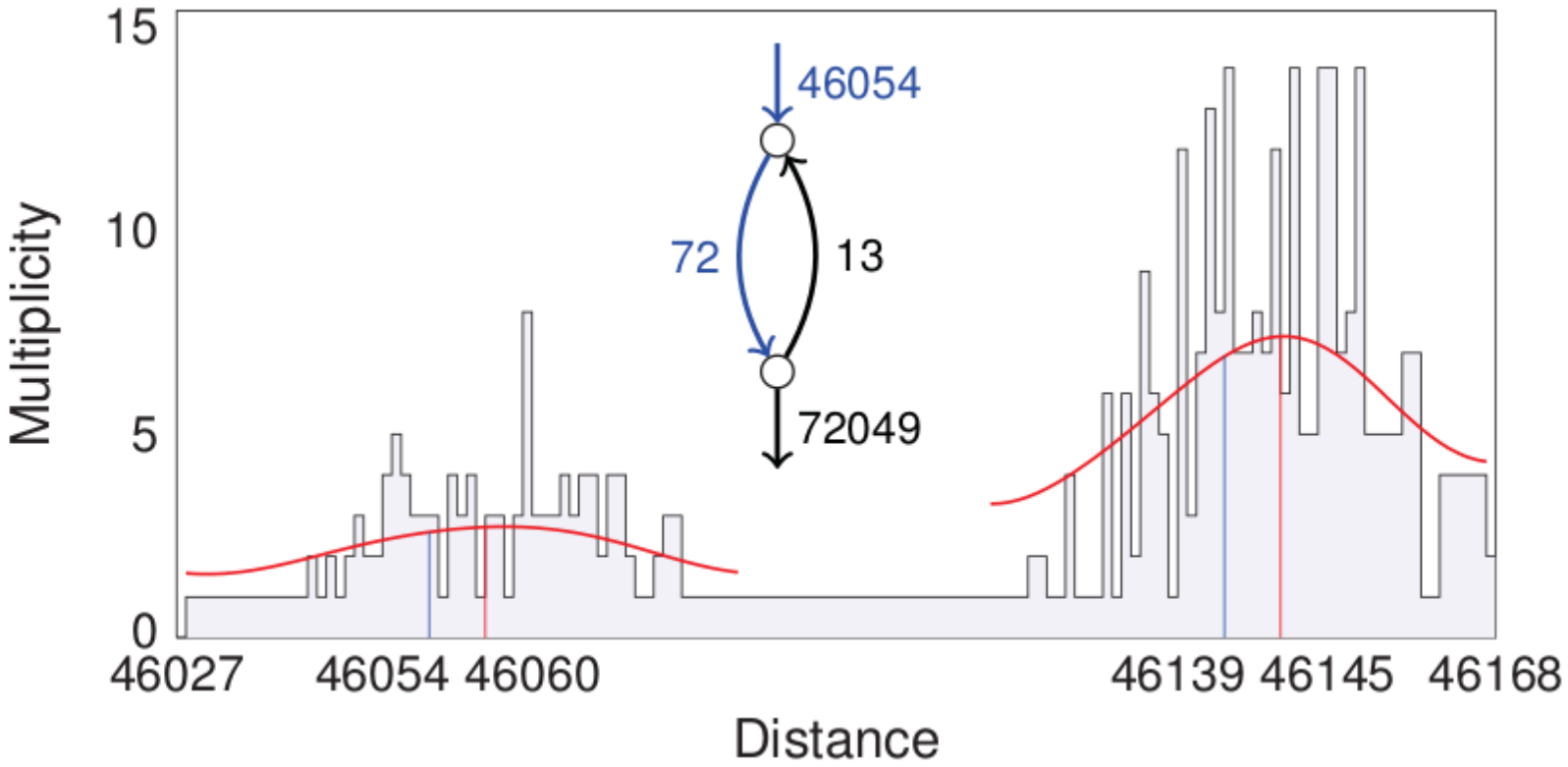


Paired DBG in SPAdes

- Repeat resolution in SPAdes stems from the basic idea of a paired de Bruijn graph (with a few extra strategies).
- In theory, there's no difference between theory and practice; in practice, there is: the distance between paired reads is not known exactly.
- This is a major problem for implementing paired de Bruijn graphs.

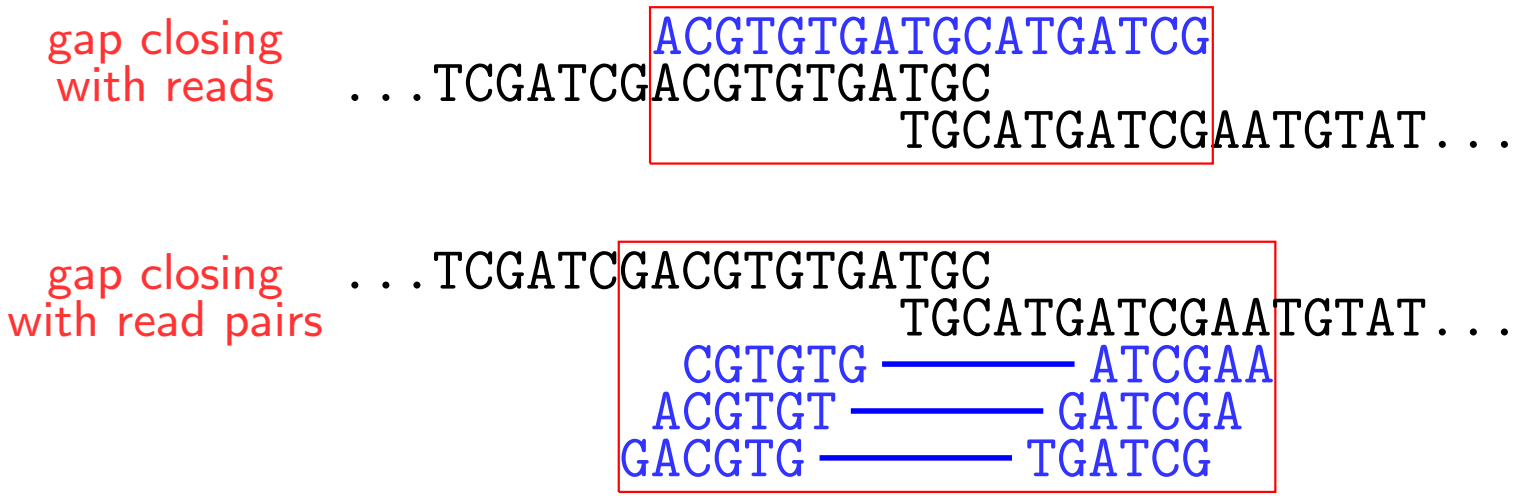
Paired DBG in SPAdes

- SPAdes approach to distance estimation: choose distances (from a finite set given by graph structure) by clustering existing paired info.



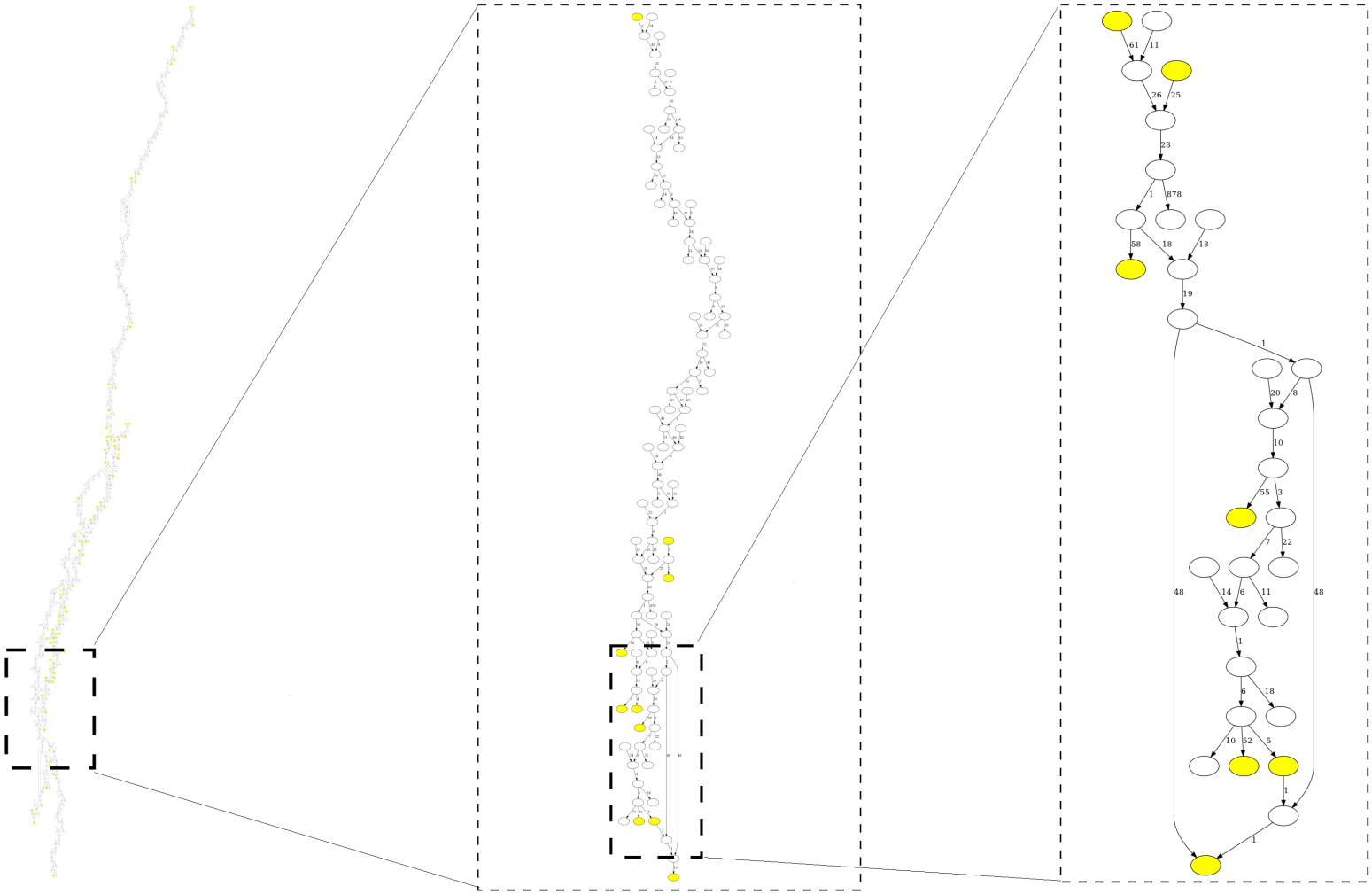
Closing gaps

- Gaps in coverage lead to discontinuities in the de Bruijn graph.
- Sometimes, gaps can be closed if matching paired info is available.

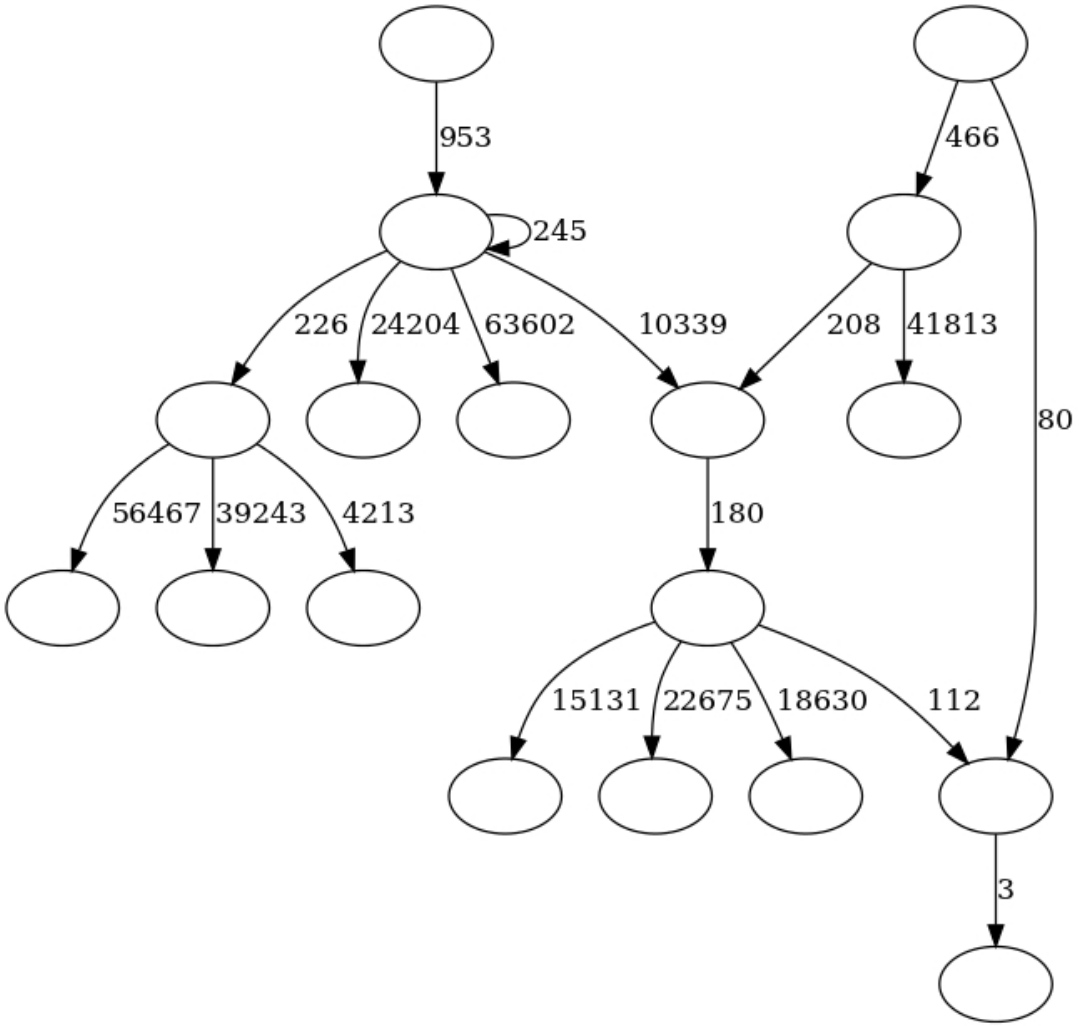


- SPAdes closes gaps not only during scaffolding, but also *before tip clipping*.

De Bruijn graph before simplification



De Bruijn graph after simplification



Omnigraph

- Note that many of these procedures (graph simplification, repeat resolution) actually do not depend on the underlying sequence, but only on a few edge parameters (coverage, length, paired info).
- In the code, these operations are abstracted out and made into general graph algorithms; we call this approach the *omnigraph*.
- Compare with *A-Bruijn graph* (Pevzner, Tang, Tesler, 2004).

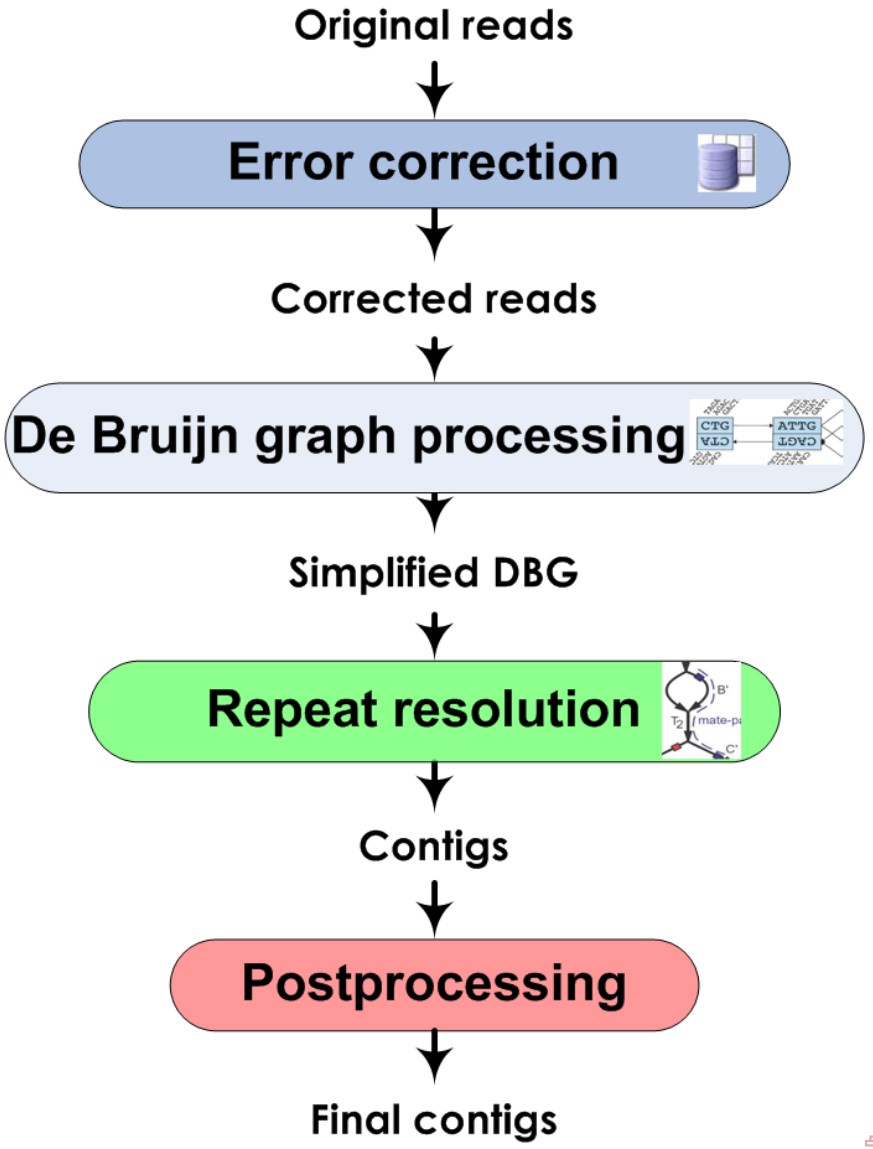
Omnigraph

- Thus, SPAdes can be used for other similar problems, e.g. *proteome assembly* (although no real-life projects have been completed yet).
- In general, assemblers often make unjustified assumptions; in the single-cell case, such assumptions are wrong much more often.
- We have designed SPADES to make *more accurate* assemblies, cutting less corners than other assemblers; for single-cell projects, it is more important than ever.

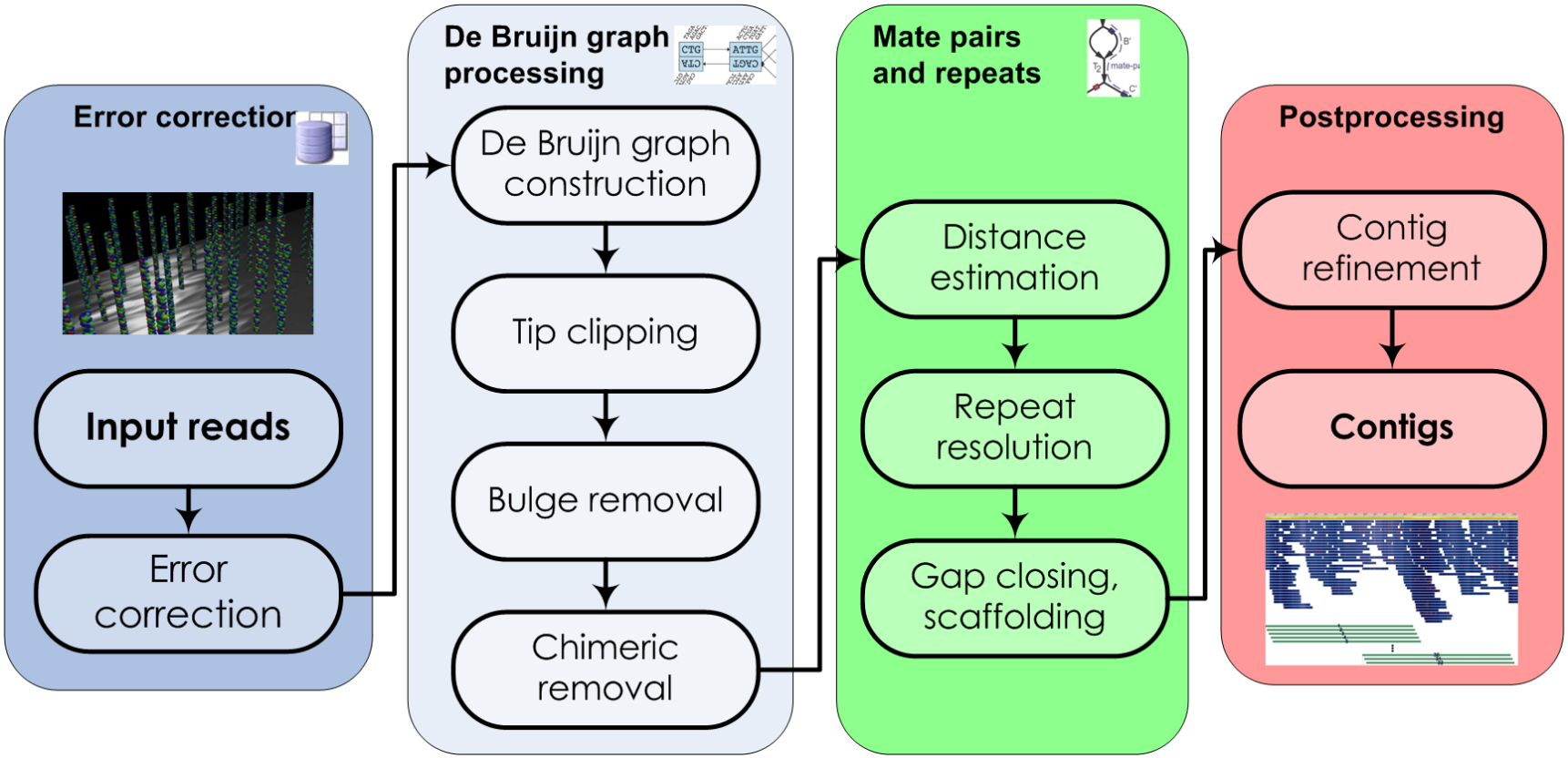
Results

Assembler	# contigs	N50 (bp)	Largest (bp)	Total (bp)	Covered (%)	Misassemblies	Mismatches (per 100 kbp)	Complete genes
Single-cell *E. coli* (ECOLI-SC)								
EULER-SR	1344	26662	126616	4369634	87.8	21	11.0	3457
SOAPdenovo	1240	18468	87533	4237595	82.5	13	99.5	3059
Velvet	428	22648	132865	3533351	75.8	2	1.9	3117
Velvet-SC	872	19791	121367	4589603	93.8	2	1.9	3654
E+V-SC	501	32051	132865	4570583	93.8	2	6.7	3809
SPAdes-single reads	1164	42492	166117	4781576	96.1	1	6.2	3888
SPAdes	1024	49623	177944	4790509	96.1	1	5.2	3911
Normal multicell sample of *E. coli* (ECOLI-MC)								
EULER-SR	295	110153	221409	4598020	99.5	10	5.2	4232
IDBA	191	50818	164392	4566786	99.5	4	1.0	4201
SOAPdenovo	192	62512	172567	4529677	97.7	1	26.1	4141
Velvet	198	78602	196677	4570131	99.9	4	1.2	4223
Velvet-SC	350	52522	166115	4571760	99.9	0	1.3	4165
E+V-SC	339	54856	166115	4571406	99.9	0	2.9	4172
SPAdes-single reads	445	59666	166117	4578486	99.9	0	0.7	4246
SPAdes	195	86590	222950	4608505	99.9	2	3.7	4268

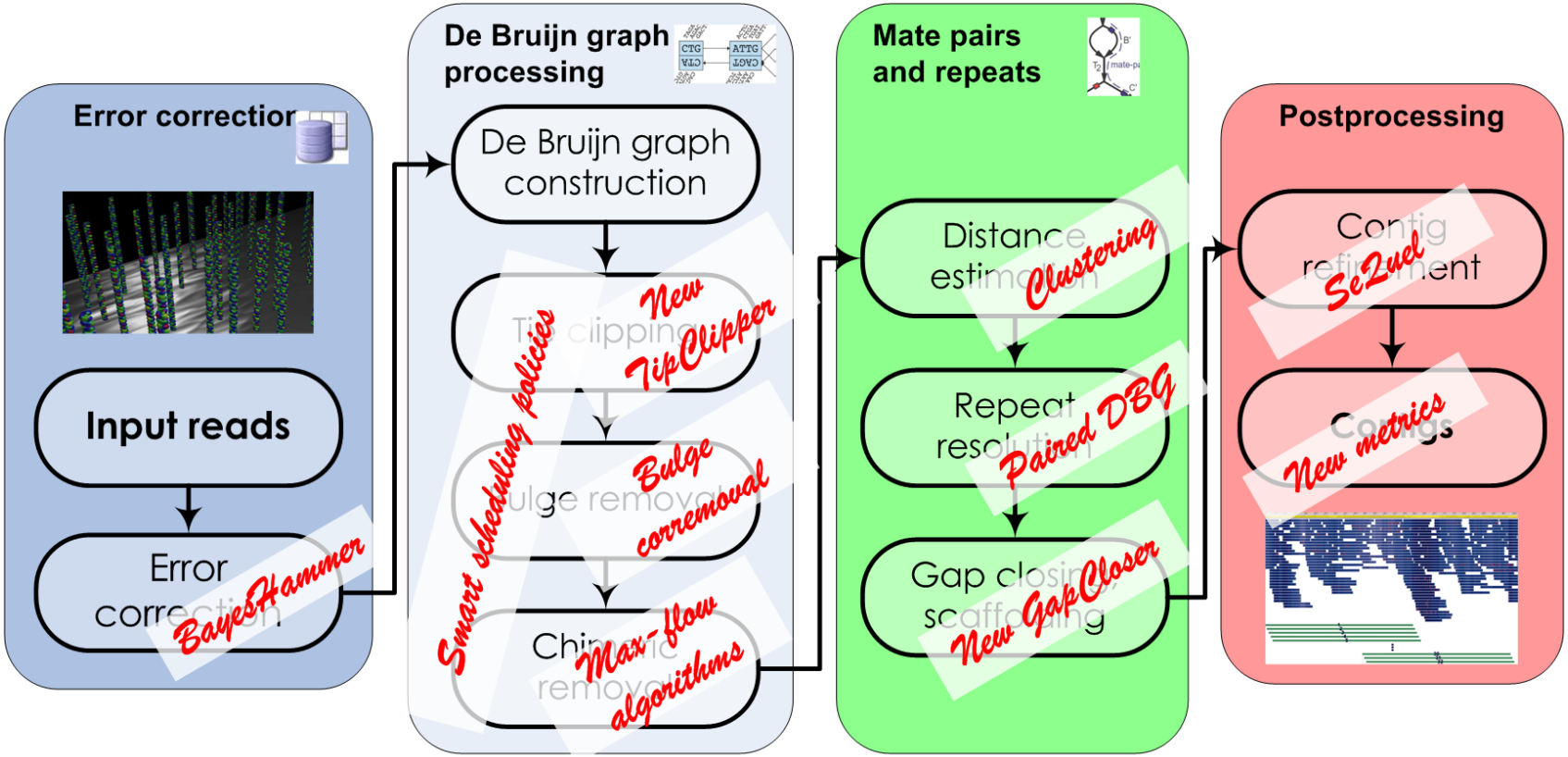
All together now



All together now



All together now



Collaborations

- Collaborations on single-cell projects:
  1. National Institute of Health (NIH), USA – Human Microbiome Project (HMP);
  2. J. Craig Venter Institite (JCVI);
  3. Joint Genome Institute (JGI) – Genomic Encyclopedia for Bacteria and Archaea (GEBA);
  4. Department of Ecology and Evolutionary Biology, Yale University.
- Other projects:
  1. Scripps Institution of Oceanography – colony of bacteria;
  2. Moscow State University.

Thank you!

Thank you for your attention!

Algorithmic Biology Lab
St. Petersburg Academic University

A. Bankevich, S. Nurk, D. Antipov, A.A. Gurevich, M. Dvorkin, A.S. Kulikov, V.M. Lesin, S.I. Nikolenko, S. Pham, A.D. Prjibelski, A.V. Pyshkin, A.V. Sirotkin, N. Vyahhi, G. Tesler, M.A. Alekseyev, P.A. Pevzner