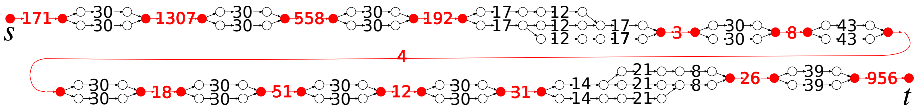
Safely filling gaps
with partial solutions common to all solutions

WABI 2016

Leena Salmela   Alexandru I. Tomescu

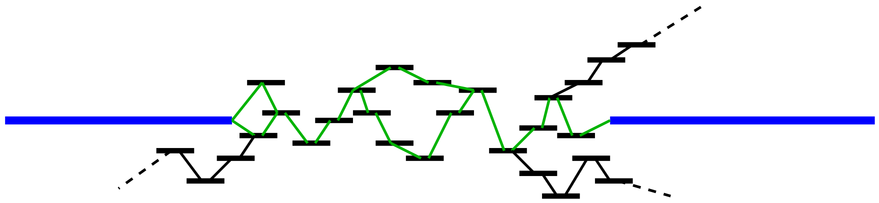
University of Helsinki

August 23rd, 2016



Gap filling

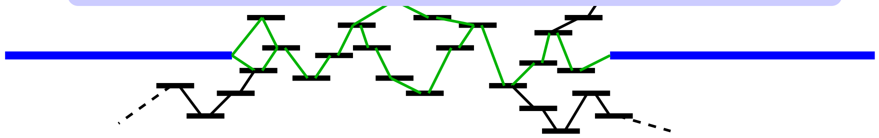
- Gap filling is the last phase in genome assembly
- Input: Scaffolds (=linearly ordered contigs) and reads
- Output: Scaffolds where gaps between contigs have been filled



Gap filling

- Gap filling is the last phase in genome assembly
- Input: Scaffolds (=linearly ordered contigs) and reads
- Output: Scaffolds where gaps between contigs have been filled

What if there are multiple paths?

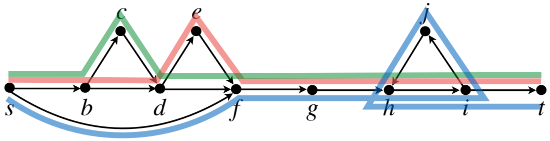


Safe and complete solutions

An algorithm is

- safe: it returns only partial solutions that are common to all solutions
- complete: it returns all safe solutions

We give a gap filling algorithm that is safe.



Previous work on safe solutions for gap filling

- Wetzel et al. 2011
  - Unique paths
  - Unique shortest paths
- Konnector (Vandervalk et al. 2014)
  - Exhaustive enumeration of all *s*-*t* paths (up to a threshold)
  - Fill a gap only when there are at most a given number of paths
  - Form multiple alignment
  - Use consensus sequence with IUPAC codes

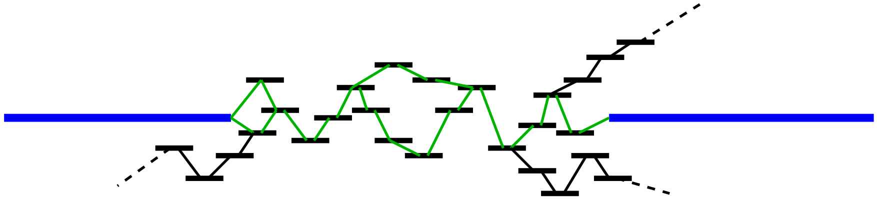
Gap filling problem

Given

- an (overlap or de Bruijn) graph $G = (V, E)$ of the whole read set
- a cost function $c : E \mapsto \mathbb{Z}_+$
- two vertices s and t representing the flanks of the contigs
- estimate of the gap length d

find an s-t path $P = v_1, v_2, \ldots, v_k$ such that

$$cost(P) = \sum_{i=1}^{k-1} c(v_i, v_{i+1}) = d.$$

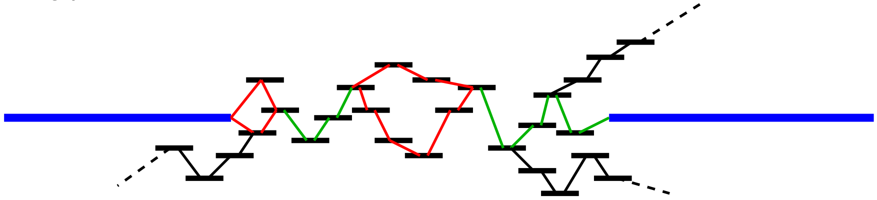


Safe and complete gap filling problem

Given

- an (overlap or de Bruijn) graph $G = (V, E)$ of the whole read set
- a cost function $c : E \mapsto \mathbb{Z}_+$
- two vertices s and t representing the flanks of the contigs
- estimate of the gap length d

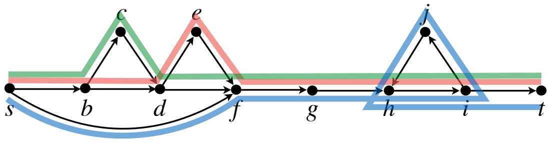
find all maximal paths in G that are sub-paths of all solutions to the gap filling problem.

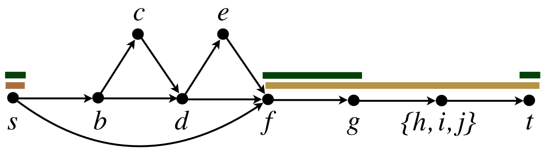


...GTTTACGtggGATCgacggggGAGCTACTAGACGGTA...

Safe (but not complete) algorithm for gap filling

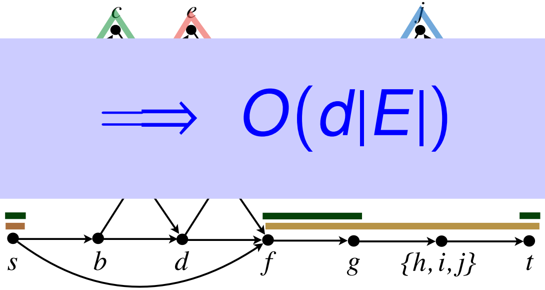
- Determine the graph G_d made up of all edges and vertices on some s-t path in G of length d
- Compute the graph G_d^{SCC} of strongly connected components $\implies$ DAG!
- Find all maximal paths that are sub-paths of all s-t paths in G_d^{SCC} $\iff$ Find bridges and cut vertices
- The corresponding paths in G_d are also sub-paths of all s-t paths in G_d





Safe (but not complete) algorithm for gap filling

- Determine the graph G_d made up of all edges and vertices on some s-t path in G of length d: $O(d|E|)$
- Compute the graph G_d^{SCC} of strongly connected components $\implies$ DAG!: $O(|V| + |E|)$
- Find all maximal paths that are sub-paths of all s-t paths in G_d^{SCC} $\iff$ Find bridges and cut vertices: $O(|E|)$
- The corresponding paths in G_d are also sub-paths of all s-t paths in G_d



$$\implies O(d|E|)$$

Experimental setup

Algorithms considering the safety notion:

- Gap2Seq Safe (this work)
- Gap2Seq Unique (only unique paths are safe)
- Sealer (Paulino et al. 2015)

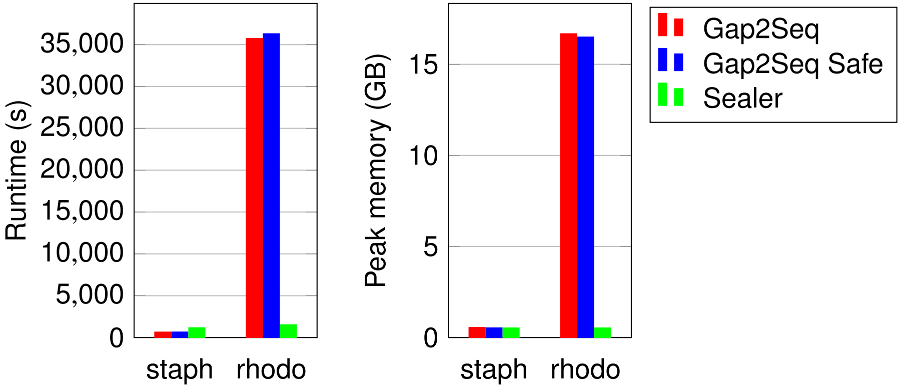
Other algorithms:

- Gap2Seq (Salmela et al. 2015)
- GapFiller with Bowtie (Boetzer and Pirovano 2012)
- GapFiller with BWA (Boetzer and Pirovano 2012)
- SOAPdenovo GapCloser (Luo et al. 2012)

Data sets:

- All *S. aureus* and *R. sphaeroides* assemblies from GAGE

Runtime and memory usage



- Experiments run on all 8 GAGE assemblies.
- We show aggregates over all assemblies.

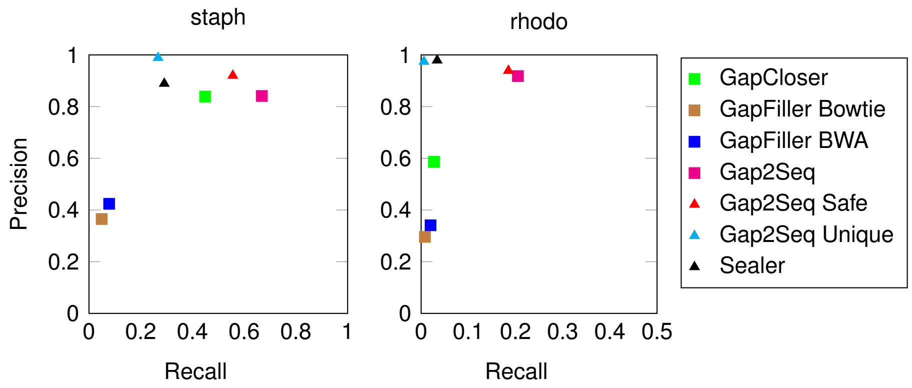
Precision and recall

Align the filled scaffolds against the reference and compute for aligned filled sequences:

- Precision:
$$\frac{\#\text{correct safe bases}}{\#\text{safe bases}}$$
- Recall:
$$\frac{\#\text{correct safe bases}}{\#\text{total gap bases}}$$

For algorithms that do not differentiate between safe and nonsafe bases all bases were considered safe.

Precision and recall



- Experiments run on all 8 GAGE assemblies.
- We show aggregates over all assemblies.

Conclusions

- We introduced the safe and complete gap filling problem
- We gave an algorithm for gap filling which is safe
- Open problem:
  Existence of an efficient safe and complete algorithm for gap filling

Thanks!

Questions?

http://www.cs.helsinki.fi/u/lmsalmel/Gap2Seq/