GoldRush-Edit : A targeted, alignment-free polishing & finishing pipeline for long read assembly, using long read k-mers

Vladimir Nikolić [1] Lauren Coombe [1] Johnathan Wong [1] Janet Li [1, 2] René Warren [1] Inanç Birol [1]

[1]Canada's Michael Smith Genome Sciences Centre at BC Cancer, Vancouver, BC V5Z 4S6, Canada
[2]Bioinformatics Graduate Program, The University of British Columbia, Vancouver, BC V6T 1Z4, Canada

Background

- The ability to generate accurate genome sequences is cornerstone to many life sciences research projects and translational applications in clinical genomics. This is especially important today as genomics projects utilizing single molecule sequencing (SMS)/long reads exclusively, such as those from Oxford Nanopore Technologies PLC (ONT, Oxford, UK), are increasing despite the still appreciable error rates they afford.
- Most popular polishing methods for long reads, e.g. Racon [1], rely on sequence alignments. The current implementations are not scalable for large (>3Gbp) genomes, requiring large memory servers.
- We present GoldRush-Edit, a memory-efficient polishing pipeline to correct base errors in long read assemblies, using a scalable and targeted k-mer-based method.

Conclusions

- GoldRush-Edit polisher is capable of outputting draft assembly quality comparable to its competitor Racon, but at an order of magnitude less memory usage and similar run time.
- It is possible to use alignments for polishing, which slightly increases the run time, but improves the polishing quality.

Funding

Genome BritishColumbia
National Institutes of Health
GenomeCanada

References

[1] Robert Vaser et al. "Fast and accurate de novo genome assembly from long uncorrected reads". In: *Genome Research* 27.5 (Jan. 2017), pp. 737–746. DOI: 10.1101/gr.214270.116. URL: https://doi.org/10.1101/gr.214270.116.

[2] René L Warren et al. "ntEdit: scalable genome sequence polishing". In: *Bioinformatics* 35.21 (May 2019). Ed. by Bonnie Berger, pp. 4430–4432. DOI: 10.1093/bioinformatics/btz400. URL: https://doi.org/10.1093/bioinformatics/btz400.

[3] Daniel Paulino et al. "Sealer: a scalable gap-closing application for finishing draft genomes". In: *BMC Bioinformatics* 16.1 (July 2015). DOI: 10.1186/s12859-015-0663-4. URL: https://doi.org/10.1186/s12859-015-0663-4.

Pipeline

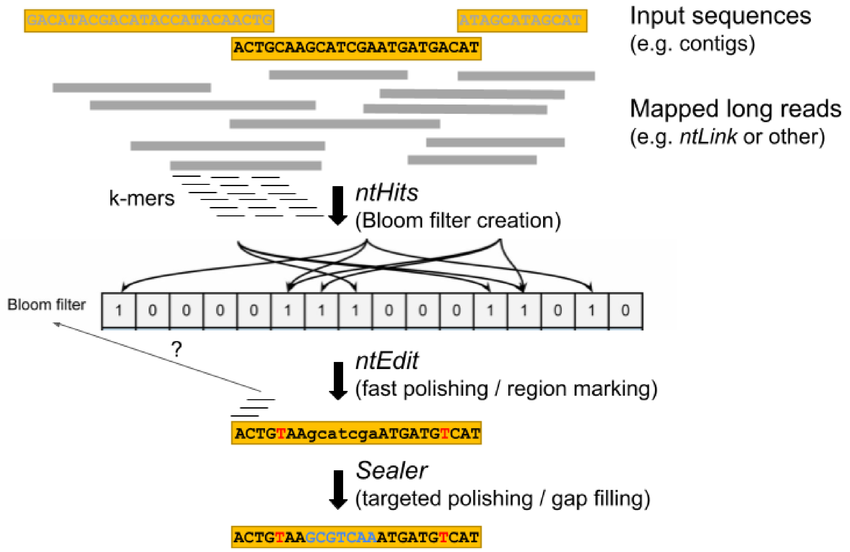

Figure 1. Diagram of the polishing paradigm: ntEdit+Sealer. The reads used for polishing are mapped to contigs and stored in Bloom filters, one for each contig. ntEdit [2] is used to correct mismatches and small indels, and mark the regions it cannot resolve. Sealer [3] is then used to polish marked regions.

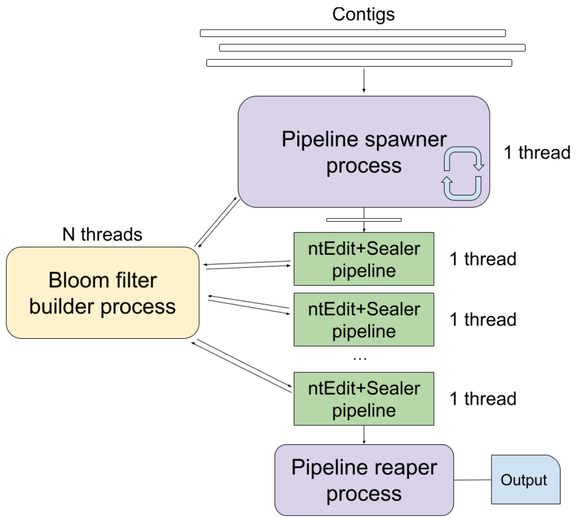

Figure 2. GoldRush-Edit spawns parallel ntEdit+Sealer pipelines and uses a dedicated Bloom filter builder process to maximize concurrency.

Performance

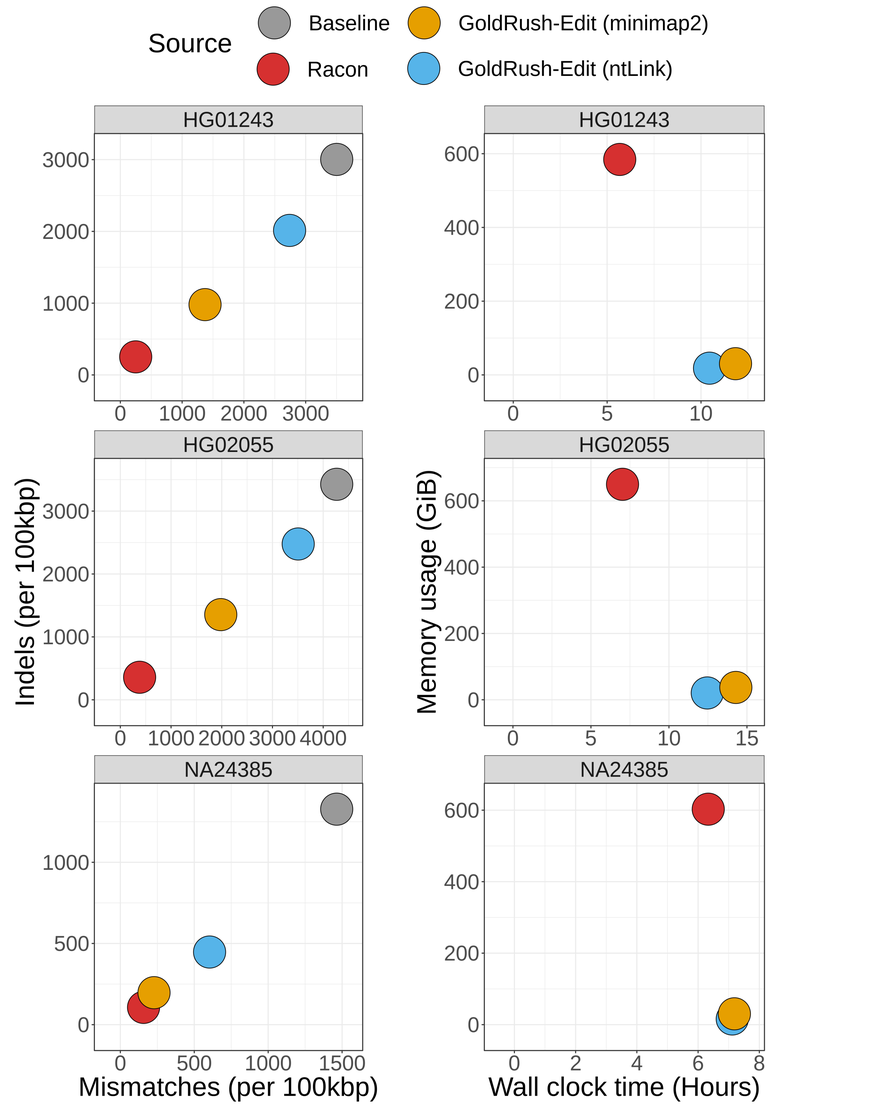

Figure 3. GoldRush-Edit by default runs ntLink to obtain alignment-free read-to-contig mappings. It can also run minimap2 to obtain mappings, which comes at a cost to run time and memory usage, but improves polishing quality. GoldRush-Edit uses over an order of magnitude less memory than Racon and has a comparable run time, while coming close in terms of polishing quality.