Journal
doi.10.1093/xxxxxx/yyyyyy
Advance Access Publication Date: Day Month Year
Genome Analysis

Genome Analysis

Cluster efficient pangenome graph construction with nf-core/pangenome

Simon Heumos [1,2,3,4,*], Andrea Guarracino [5,6,†], Sven Nahnsen [1,2,3,4,*], Pjotr Prins [5], Erik Garrison [5]

[1] Quantitative Biology Center (QBiC), University of Tübingen, Tübingen 72076, Germany
[2] Biomedical Data Science, Department of Computer Science, University of Tübingen, Tübingen 72076, Germany
[3] M3 Research Center, University Hospital Tübingen, Tübingen 72076 , Germany
[4] Institute for Bioinformatics and Medical Informatics (IBMI), University of Tübingen, Tübingen 72076, Germany
[5] Department of Genetics, Genomics and Informatics, University of Tennessee Health Science Center, Memphis, TN 38163, USA
[6] Genomics Research Centre, Human Technopole, Milan 20157, Italy

[*] To whom correspondence should be addressed.
[†] The authors wish it to be known that, in their opinion, the first two authors should be regarded as Joint First Authors.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Pangenome graphs can encode the entire genomic variability between multiple genomes. Current pangenome graph construction methods exclude complex sequences or are reference-based, leading to reference, order, or orientation bias. This was addressed by the PanGenome Graph Builder (PGGB) pipeline. However, PGGB's bash implementation limits its ease of deployment, optimal use of compute resources, and cluster scalability, making it impractical to build very large pangenome graphs.
Results: We implemented *nf-core/pangenome*, a reference-unbiased approach to construct pangenome graphs. Mirroring PGGB, it iteratively refines an all-to-all whole-genome alignment graph that allows to explore sequence conservation and variation, infer phylogeny, and identify recombination events. nf-core/pangenome is implemented in Nextflow and follows the nf-core best practice development guidelines. Providing all software dependencies in biocontainers makes the pipeline portable and easy to install on high-performance computing environments. In contrast to PGGB, this allows nf-core/pangenome to distribute the quadratically complex all-to-all base-level alignments across nodes of a cluster. Evaluating 1024 E. coli haplotypes, the time spent on base-pair level alignments is reduced linearly with an increase in alignment problem chunks. To demonstrate the scalability of nf-core/pangenome, we built pangenome graphs of 1000 chromosome 19 human haplotypes, and of 2146 E. coli sequences. nf-core/pangenome was two to three times faster compared to PGGB while not increasing the greenhouse gas emissions.
Availability: nf-core/pangenome is published as free software under the MIT open-source license. Source code can be downloaded from `https://github.com/nf-core/pangenome` and the documentation is accessible at `https://nf-co.re/pangenome/1.1.2/docs/usage`. Each release is archived on Zenodo `10.5281/zenodo.8202636`.
Contact: simon.heumos@qbic.uni-tuebingen.de, sven.nahnsen@qbic.uni-tuebingen.de

1 Introduction

The availability of high-quality collections of population-wide whole-genome assemblies (Liao *et al.*, 2023; Kang *et al.*, 2023; Weller *et al.*, 2023; Zhou *et al.*, 2022; Liu *et al.*, 2020; Leonard *et al.*, 2022) offers new opportunities to study sequence evolution and variation within and between genomic populations. A challenge is simultaneously representing and analyzing hundreds to thousands of genomes at a gigabase scale. One solution here is a pangenome. A pangenome models a population's entire set of genomic sequences (Ballouz *et al.*, 2019). In contrast to reference-based genomic approaches, which relate sequences to a linear genome, pangenomics relates each new sequence to all the others represented in the pangenome (The Computational Pan-Genomics Consortium, 2016;

Eizenga *et al.*, 2020; Sherman and Salzberg, 2020) minimizing reference-bias. Pangenomes can be described as sequence graphs which store DNA sequences in nodes with edges connecting the nodes as they occur in the individual sequences (Hein, 1989). Genomes are encoded as paths traversing the nodes (Garrison *et al.*, 2018).

Current pangenome graph construction methods exclude complex sequences, are tree-guided, or reference-based (Li *et al.*, 2020; Hickey *et al.*, 2023) leading to reference, order, or orientation bias. Although whole genome scaling approaches for unbiased pangenome graph construction (Chin *et al.*, 2023; Minkin *et al.*, 2016) exist, their reliance on k-mer-based data structures often leads to unwanted complexity for downstream analysis. One recent approach that overcomes such limitations is the PanGenome Graph Builder (PGGB) pipeline (Garrison *et al.*, 2023). PGGB iteratively refines an all-to-all whole-genome alignment graph that lets us explore sequence conservation and variation, infer phylogeny, and identify recombination events. PGGB was already extensively evaluated (Garrison *et al.*, 2023; Andreace *et al.*, 2023) and applied to build the first draft human pangenome reference (Liao *et al.*, 2023). However, PGGB is implemented in bash: This (a) makes it difficult to deploy on HPC systems, (b) does not allow for a fine granular tuning of computing resources for different steps of the pipeline (Sztuka *et al.*, 2024), and (c) limits its cluster scalability because PGGB can only use the resources of one node.

To compensate for that, we wrote *nf-core/pangenome*, a reference-unbiased approach to construct pangenome graphs. Mirroring PGGB, nf-core/pangenome is implemented in Nextflow (Di Tommaso *et al.*, 2017) and follows the community-curated nf-core (Ewels *et al.*, 2020) best practice development guidelines. Providing all software dependencies in biocontainers (da Veiga Leprevost *et al.*, 2017) makes the pipeline portable and easy to install on HPC environments. In contrast to PGGB, this facilitates nf-core/pangenome to distribute the quadratic all-to-all base-level alignments across nodes of a cluster by splitting the approximate alignments into problems of equal size. We benchmarked the time spent on base-pair level alignments and show that it is reduced linearly with an increase in alignment problem chunks. We showcase the workflow's scalability by applying it to 1000 chromosome 19 human haplotypes, and to 2146 E. coli sequences, which were built in less than half the time PGGB required while not increasing the CO2 equivalent (CO2e) emissions.

2 Material and Methods

3 Results

4 Discussion

Acknowledgments

We thank Matthias Seybold from the Quantitative Biology Center for maintaining the Core Facility Cluster.

Funding

S.H. acknowledges funding from the Central Innovation Programme (ZIM) for SMEs of the Federal Ministry for Economic Affairs and Energy of Germany. S.N. acknowledges Germany's Excellence Strategy (CMFI), EXC-2124 and (iFIT)—EXC 2180–390900677. This work was supported by the BMBF-funded de.NBI Cloud within the German Network for Bioinformatics Infrastructure (de.NBI) (031A532B, 031A533A, 031A533B, 031A534A, 031A535A, 031A537A, 031A537B, 031A537C, 031A537D and 031A538A). A.G. acknowledges efforts by Nicole Soranzo to establish a pangenome research unit at the Human Technopole in Milan, Italy. JNM.S., J.L., Z.Z., P.P., and E.G. acknowledge funding from the NSF PPoSS Award #2118709. The authors gratefully acknowledge support from National Institutes of Health/NIDA U01DA047638 (E.G.), National Institutes of Health/NIGMS R01GM123489 (E.G. and P.P.)

Competing interests

Author J.H. is employed by Computomics GmbH.

Software and data availability

Software versions, code, and links to data used to prepare this manuscript can be found at `https://github.com/pangenome/sorting-paper`. Animations of the algorithm are deposited at `https://doi.org/10.5281/zenodo.8288999`.

References

Andreace, F. *et al.* (2023). Comparing methods for constructing and representing human pangenome graphs. *Genome Biology*, **24**(1).

Ballouz, S. *et al.* (2019). Is it time to change the reference genome? *Genome Biology*, **20**(1), 159.

Chin, C.-S. *et al.* (2023). Multiscale analysis of pangenomes enables improved representation of genomic diversity for repetitive and clinically relevant genes. *Nature Methods*, **20**(8), 1213–1221.

da Veiga Leprevost, F. *et al.* (2017). Biocontainers: an open-source and community-driven framework for software standardization. *Bioinformatics*, **33**(16), 2580–2582.

Di Tommaso, P. *et al.* (2017). Nextflow enables reproducible computational workflows. *Nature Biotechnology*, **35**(4), 316–319.

Eizenga, J. M. *et al.* (2020). Pangenome graphs. *Annual Review of Genomics and Human Genetics*, **21**(1), 139–162.

Ewels, P. A. *et al.* (2020). The nf-core framework for community-curated bioinformatics pipelines. *Nature Biotechnology*, **38**(3), 276–278.

Garrison, E. *et al.* (2018). Variation Graph Toolkit Improves Read Mapping by Representing Genetic Variation in the Reference. *Nature Biotechnology*, **36**(9), 875–879.

Garrison, E. *et al.* (2023). Building pangenome graphs. *bioRxiv*.

Hein, J. (1989). A new method that simultaneously aligns and reconstructs ancestral sequences for any number of homologous sequences, when the phylogeny is given. *Molecular Biology and Evolution*.

Hickey, G. *et al.* (2023). Pangenome graph construction from genome alignments with minigraph-cactus. *Nature Biotechnology*, **42**(4), 663–673.

Kang, M. *et al.* (2023). The pan-genome and local adaptation of arabidopsis thaliana. *Nature Communications*, **14**(1).

Leonard, A. S. *et al.* (2022). Structural variant-based pangenome construction has low sensitivity to variability of haplotype-resolved bovine assemblies. *Nature Communications*, **13**(1).

Li, H. *et al.* (2020). The Design and Construction of Reference Pangenome Graphs with Minigraph. *Genome Biology*, **21**(1), 265.

Liao, W.-W. *et al.* (2023). A draft human pangenome reference. *Nature*, **617**(7960), 312–324.

Liu, Y. *et al.* (2020). Pan-genome of wild and cultivated soybeans. *Cell*, **182**(1), 162–176.e13.

Minkin, I. *et al.* (2016). Twopaco: an efficient algorithm to build the compacted de bruijn graph from many complete genomes. *Bioinformatics*, **33**(24), 4024–4032.

Sherman, R. M. and Salzberg, S. L. (2020). Pan-genomics in the human genome era. *Nature Reviews Genetics*, **21**(4), 243–254.

Sztuka, M. *et al.* (2024). Nextflow vs. plain bash: different approaches to the parallelization of SNP calling from the whole genome sequence data. *NAR Genomics and Bioinformatics*, **6**(2), lqae040.

The Computational Pan-Genomics Consortium (2016). Computational pan-genomics: status, promises and challenges. *Briefings in Bioinformatics*, page bbw089.

Weller, C. A. *et al.* (2023). Highly complete long-read genomes reveal pangenomic variation underlying yeast phenotypic diversity. *Genome Research*, **33**(5), 729–740.

Zhou, Y. *et al.* (2022). Assembly of a pangenome for global cattle reveals missing sequences and novel structural variations, providing new insights into their diversity and evolutionary history. *Genome Research*, **32**(8), 1585–1601.

5 Supplement