Streamlining genome assembly and QC with the reads2genome workflow

We want to swiftly generate genome assemblies and produce quality control statistics to gauge the need for more curation. We built a Nextflow pipeline that assembles Illumina, Nanopore, or PacBio sequencing reads for a single organism and runs QC checks on the resulting assembly.

Version 5, published Jun 8, 2024. Originally published Apr 11, 2023.

Arcadia Science DOI: 10.57844/arcadia-75v2-nj84

Purpose

We want to ensure that we assemble high-quality genomes in a reproducible manner. We built a Nextflow workflow, reads2genome, to assemble sequencing reads from Illumina, Nanopore, or PacBio HiFi technologies for a single organism and produce quality control statistics for the resulting assembly. The product of this pipeline is an assembly, mapped reads, and interactive visualizations reported with MultiQC. The final HTML report addresses assembly quality, lineage-specific checks, and mapping statistics that will help us make more informed decisions about downstream curation and functional annotation efforts.

We built this pipeline using open-source software and tools, and we hope others will shape and extend this resource to fit their needs.

- This pub is part of the **project**, "Useful computing at Arcadia." Visit the project narrative for more background and context.

- The **reads2genome Nextflow pipeline** is available at this GitHub repository.
- We've included a **sample report** of assemblies that we generated from 24 microorganisms sequenced with PacBio HiFi technology in the "Food safety and infectious microbes" dataset.

The resource

The problem

Running the commands for assembly and quality control (QC) checks from sequencing efforts of single organisms can be fairly straightforward but repetitive, depending on the desired outcomes. We want to quickly generate assemblies and resulting statistics to decide if further curation is needed before moving forward with downstream steps.

Our solution

We previously developed the "hifi2genome" workflow (see the earlier version of this pub) for automating genome assembly and QC for PacBio HiFi sequencing efforts. Since releasing that workflow, we've expanded our genome sequencing efforts to include Illumina and Nanopore technologies. Therefore, we developed a computational resource that automates genome assembly and quality control checks from Illumina, Nanopore, or PacBio Hifi technologies, called reads2genome.

An overview of the reads2genome workflow

The reads2genome pipeline injects a sample sheet that includes the sample name and the local path, URL, or URI of the reads in FASTQ format (Figure 1).

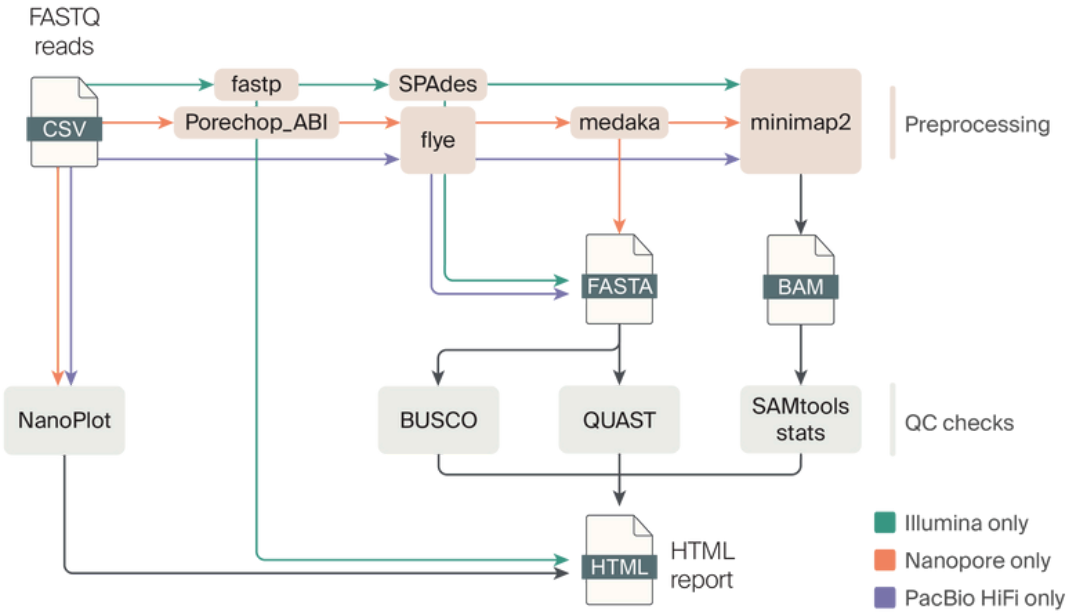


Figure 1. **Computational steps in the reads2genome workflow**.

We designed the pipeline to separately process Illumina, Nanopore, or PacBio HiFi reads from a single organism, and therefore researchers cannot currently use reads2genome for hybrid assembly or scaffolding approaches. We made this decision based on our most common internal use cases, which have evolved from solely using PacBio HiFi for sequencing genomes from single organisms (the previous version of this pub was limited to this use case). The user must therefore select the corresponding technology with the `--platform` flag with either `--platform illumina`, `--platform nanopore` or `--platform pacbio` when launching the workflow.

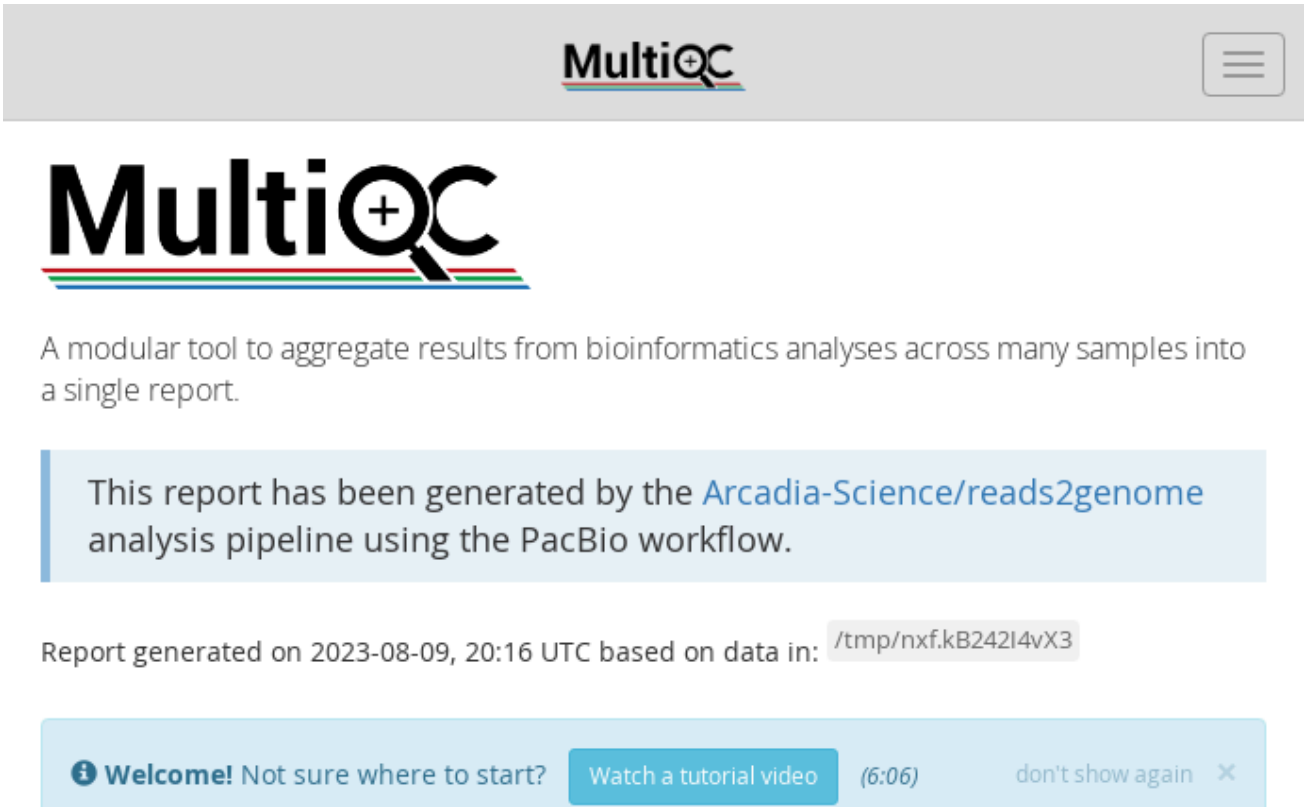
Key functions

Most of the tools downstream of read QC and assembly are the same for all technologies. We describe tools specific to Illumina, Nanopore, or PacBio HiFi technologies below. After inputting reads in FASTQ format, the pipeline performs basic read QC, adapter removal, and assembly, and then maps the reads back to the assembly with minimap2 [1]. Subsequent assembly checks run in parallel and generate QC statistics. Currently, the checks include 1) lineage-specific QC marker checks with BUSCO [2], 2) assembly quality statistics with QUAST [3], and 3) mapping rate stats with `samtools stats` [4][5].

In addition to the sample sheet containing the path to the reads and the corresponding `--platform` selection, the only other input the user must provide is the closest BUSCO lineage of the target organism for calculating lineage-specific completeness and redundancy statistics.

The final step of the pipeline aggregates the results from QUAST, BUSCO, `samtools stats`, and the information about the pipeline run and software versions into an HTML report with MultiQC [5] (Figure 1). MultiQC can generate an HTML report from the log files of numerous bioinformatics programs, and you can use it with or without running a Nextflow pipeline. The MultiQC report currently outputs general information about the assemblies and mapping statistics, as well as more detailed information about each assembly from QUAST, including the distribution of sizes of contigs that were assembled, BUSCO lineage assessment results, and outputs from `samtools stats`, including percentages of the reads that mapped to each corresponding assembly and alignment metrics. The resulting MultiQC HTML report is emailed to the end user if SMTP credentials for the pipeline are configured.

View an example of the MultiQC HTML report from the pipeline below from a run on a publicly available dataset of PacBio HiFi sequencing of 24 microorganisms from the "Food safety and infectious microbes" dataset with `nextflow run main.nf -input samplesheet.csv -outdir microbial_hifi_assemblies -profile docker -lineage bacteria`:



Download the sample report:

`reads2genome_multiqc_report.html` Download

Illumina-specific tools

When the user launches the workflow using `--platform illumina`, reads2genome filters each set of paired-end reads with fastp [6] and assembles them with SPAdes [7].

Nanopore- and PacBio HiFi-specific tools

When the user launches the workflow using `--platform pacbio`, reads2genome summarizes the quality stats of the reads with NanoPlot [8] and assembles them with Flye [9] using the `--pacbio-hifi` flag. When the user launches the workflow using `--platform nanopore`, it similarly summarizes reads with NanoPlot, but includes an adapter-trimming step with Porechop_ABI [10] before assembling with Flye using the `--nano-hq` flag. After assembly, the workflow polishes contigs using Medaka with default parameters [11].

Deployment

We deploy the pipeline with continuous integration testing using subsampled reads for each sequencing technology, ensuring proper execution of the workflow as we add new features.

We are currently deploying all of our Nextflow workflows, including reads2genome, through Nextflow Tower using our AWS Batch setup [12]. The pipeline is still fully executable locally via the command line and works on diverse compute infrastructure setups.

We found that for organisms with small genomes, such as bacteria and archaea, reads2genome assembles the reads fairly quickly, and can run these jobs on interruptible AWS EC2 spot instances and complete successfully. However, for higher-order eukaryotes with larger genomes, like humans and ticks [13], which might take multiple days to assemble, we needed to reconfigure the Nextflow Tower queue directive settings so that assemblies are run via on-demand instances and are not interrupted.

Next steps

This version of the reads2genome pipeline is a simple way to assemble reads obtained from a single organism using either Illumina, Nanopore, or PacBio HiFi technologies and to provide QC stats for the resulting assembly. In the future, we would like to:

- Provide the user with the option to use other assembly algorithms (such as Hifiasm) in place of or concurrently with Flye to compare assembly outputs for technologies such as PacBio HiFi.
- Add an optional endosymbiont detection sub-workflow for pulling out contigs that do not belong to the host genome and are likely symbiont(s) sequences.
- Configuring Medaka to run on GPU instances.

For these efforts, we have created GitHub issues in the reads2genome GitHub repository and welcome outside suggestions and contributions through pull

requests!

Contributors (A–Z)

- **Feridun Mert Celebi**: Software, Supervision, Validation
- **Megan L. Hochstrasser**: Editing, Visualization
- **Elizabeth A. McDaniel**: Conceptualization, Software, Visualization, Writing
- **Taylor Reiter**: Critical Feedback, Validation
- **Peter S. Thuy-Boun**: Critical Feedback

References

1. Li H. (2018). Minimap2: pairwise alignment for nucleotide sequences. https://doi.org/10.1093/bioinformatics/bty191
2. Simão FA, Waterhouse RM, Ioannidis P, Kriventseva EV, Zdobnov EM. (2015). BUSCO: assessing genome assembly and annotation completeness with single-copy orthologs. https://doi.org/10.1093/bioinformatics/btv351
3. Gurevich A, Saveliev V, Vyahhi N, Tesler G. (2013). QUAST: quality assessment tool for genome assemblies. https://doi.org/10.1093/bioinformatics/btt086
4. Li H, Handsaker B, Wysoker A, Fennell T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R, Subgroup 1GPDP. (2009). The Sequence Alignment/Map format and SAMtools. https://doi.org/10.1093/bioinformatics/btp352
5. Ewels P, Magnusson M, Lundin S, Käller M. (2016). MultiQC: summarize analysis results for multiple tools and samples in a single report. https://doi.org/10.1093/bioinformatics/btw354
6. Chen S, Zhou Y, Chen Y, Gu J. (2018). fastp: an ultra-fast all-in-one FASTQ preprocessor. https://doi.org/10.1093/bioinformatics/bty560
7. Prjibelski A, Antipov D, Meleshko D, Lapidus A, Korobeynikov A. (2020). Using SPAdes De Novo Assembler. https://doi.org/10.1002/cpbi.102
8. De Coster W, D'Hert S, Schultz DT, Cruts M, Van Broeckhoven C. (2018). NanoPack: visualizing and processing long-read sequencing data.

https://doi.org/10.1093/bioinformatics/bty149

9. Kolmogorov M, Yuan J, Lin Y, Pevzner PA. (2019). Assembly of long, error-prone reads using repeat graphs. https://doi.org/10.1038/s41587-019-0072-8

10. Bonenfant Q, Noé L, Touzet H. (2022). Porechop_ABI: discovering unknown adapters in ONT sequencing reads for downstream trimming. https://doi.org/10.1101/2022.07.07.499093

11. https://github.com/nanoporetech/medaka

12. Celebi FM, McDaniel EA, Reiter T. (2023). Creating reproducible workflows for complex computational pipelines. https://doi.org/10.57844/arcadia-cc5j-a519

13. Chou S, Poskanzer KE, Rollins M, Thuy-Boun PS. (2023). De novo assembly of a long-read Amblyomma americanum tick genome. https://doi.org/10.57844/arcadia-9b6j-q683