Jun 11, 2024

Genome assembly (Nanopore and Illumina reads)

DOI
dx.doi.org/10.17504/protocols.io.kxygxywyol8j/v1

Rafael Rodrigues Ferrari[1], Thiago Mafra Batista[1]

[1]Universidade Federal do Sul da Bahia

bioinfo

Thiago Mafra Batista
Universidade Federal do Sul da Bahia

DOI: dx.doi.org/10.17504/protocols.io.kxygxywyol8j/v1

Protocol Citation: Rafael Rodrigues Ferrari, Thiago Mafra Batista 2024. Genome assembly (Nanopore and Illumina reads). **protocols.io https://dx.doi.org/10.17504/protocols.io.kxygxywyol8j/v1**

Manuscript citation:
Ferrari RR, Batista TM, Zhou QS, Hilário HO, Orr MC, Luo A, Zhu CD. The Whole Genome of *Colletes collaris* (Hymenoptera: Colletidae): An Important Step in Comparative Genomics of Cellophane Bees. Genome Biol Evol. 2023 May 5;15(5):evad062. doi: 10.1093/gbe/evad062. PMID: 37075227; PMCID: PMC10159585.

Protocol status: Working
We use this protocol and it's working

Created: June 11, 2024

Last Modified: June 11, 2024

Protocol Integer ID: 101556

Keywords: long reads, nanopore reads, genome assembly, genome polishment, bioinformatics

Abstract

This protocol offers detailed, step-by-step instructions for students and researchers to assemble nuclear genomes using long reads generated by Nanopore technology. Before assembling the genome, we will align the reads against a bacterial genome database to eliminate potential contamination. The assembled contigs will then be polished using Illumina short reads.

SEQUENCING QUALITY CHECK

1 ****LongQC (https://github.com/yfukasawa/LongQC)****

Prepare a .pbs file to run the analysis remotely on Sagarana

```
python /home/fafinha/bin/LongQC/longQC.py sampleqc -x ont-ligation
-c /tmp/LongQC_run/reads_trim.fq \
-p 64 -o /tmp/LongQC_run
/home/fafinha/colletes_collaris/reads/genomic_reads/longreads_rawdata_collaris.fq

mv /tmp/LongQC_run/ /home/fafinha/colletes_collaris/
```

CROSS-SPECIES CONTAMINATION FILTERIN

2 ****Magic-BLAST (https://ncbi.github.io/magicblast/)****

Index the database

```
$~/bin/ncbi-magicblast-1.7.0/bin/makeblastdb -in
refseq_release_215_bacteria.fna -dbtype nucl
```

ONT whole-genome sequencing

Prepare a .pbs file to run the analysis remotely on Sagarana

```
magicblast -db
/databases/ref_prok_rep_genomes_out20/ref_prok_rep_genomes \
-query
/home/fafinha/collaris/reads/genomic_reads/reads/genomic_reads/ONT_longreads_rawdata_collaris.fq \
-out_unaligned ONT_longreads_unaligned_in_refseq_prok_collaris.fa
-num_threads 80 -infmt fastq -unaligned_fmt fasta > output.sam
```

Illumina whole-genome sequencing

Prepare a .pbs file to run the analysis remotely on Sagarana

```
magicblast -db
/databases/ref_prok_rep_genomes_out20/ref_prok_rep_genomes -query
/home/fafinha/collaris/reads/genomic_reads/Illumina_shortreads.R1.
fastq \
-query_mate
/home/fafinha/collaris/reads/genomic_reads/Illumina_shortreads.R2.
fastq \
-paired -no_discordant -infmt fastq -unaligned_fmt sam -
num_threads 128 \
-out_unaligned
/home/fafinha/collaris/mafra/descontamination/illumina_reads/illum
ina_unaligned_in_refseq_prok.sam \
-out
/home/fafinha/collaris/mafra/descontamination/illumina_reads/illum
ina_aligned_in_refseq_prok.sam
```

Convert output file

```
$/programs/samtools-1.12/bin/samtools view -Sb -@12
illumina_unaligned_in_refseq_prok.sam >
illumina_unaligned_in_refseq_prok.bam

$/programs/samtools-1.12/bin/samtools sort
illumina_unaligned_in_refseq_prok.bam -o
illumina_unaligned_in_refseq_prok_sorted.bam -@12

$/programs/samtools-1.12/bin/samtools fastq -1 paired1.fq -2
paired2.fq -n illumina_unaligned_in_refseq_prok_sorted.bam -@12
```

GENOME SIZE ESTIMATION

3 ****Jellyfish (https://github.com/gmarcais/Jellyfish)****

Counting k-mers

Prepare a .pbs file to run the analysis remotely on Sagarana

```
/programs/jellyfish/jellyfish-2.3.0 count -C -m 21 -s 10G -t 36
/home/fafinha/collaris/reads/genomic_reads/D2015099C_L4_304X04.R1.
fastq \
/home/fafinha/collaris/reads/genomic_reads/D2015099C_L4_304X04.R2.
fastq -o /home/fafinha/collaris/Jellyfish/reads.jf

/programs/jellyfish/jellyfish-2.3.0 histo -t 36
/home/fafinha/collaris/Jellyfish/reads.jf >
/home/fafinha/collaris/Jellyfish/reads.histo
```

Size estimation

/////STRATEGY #1: GenomeScope (on my PC)\\\\\

Go to the directory where reads.histo is located

```
$/home/rafael/genomescope2.0/genomescope.R -i reads.histo -o
output -k 21
```

/////STRATEGY #2: R (on my PC)\\\\\

Go to the directory where reads.histo is located

```
$R

$library ("findGSE")

$findGSE(histo="reads.histo", sizek=21, outdir="21mer")
```

GENOME ASSEMBLY

4 ****NextDenovo (https://github.com/Nextomics/NextDenovo)****

Prepare an 'input.fofn' file

```
$ls
/home/fafinha/collaris/reads/genomic_reads/ONT_longreads_rawdata_c
ollaris.fq > input.fofn
```

Prepare a 'run.cfg' file

```
[General]
job_type = local
job_prefix = nextDenovo
task = all
rewrite = yes
deltmp = yes
parallel_jobs = 24
input_type = raw
read_type = ont
input_fofn = /home/fafinha/collaris/NextDenovo_run/input.fofn
workdir = 01_rundir

[correct_option]
read_cutoff = 1k
genome_size = 300m
sort_options = -m 20g -t 8
minimap2_options_raw = -t 8
pa_correction = 3
correction_options = -p 15

[assemble_option]
minimap2_options_cns = -t 8
nextgraph_options = -a 1
```

Prepare a .pbs file to run the analysis remotely on Sagarana

```
/home/fafinha/bin/NextDenovo/nextDenovo /home/fafinha/collaris/NextDenovo_run/run.cfg
```

4.1 GENOME ASSEMBLY STATISTICS

****scaffolds_stats****

Compare two runs and include the stats into a .txt

```
$scaffold_stats.pl -f run1/assembly.fasta run2/assembly.fasta -N 1 -t 1000 10000 | tee stats.txt
```

****BBMap (https://jgi.doe.gov/data-and-tools/bbtools/bb-tools-user-guide/bbmap-guide/)****

#BBMap is part of BBTools (https://jgi.doe.gov/data-and-tools/bbtools/bb-tools-user-guide/)

Run the script stats.sh using the main output produced by Flye (on my PC)

```
$bash /mnt/d/Genomics/bbmap/stats.sh in=nd.asm.fasta
out=nd.asm.fasta_stats.txt
```

****BUSCO (https://busco.ezlab.org/)****

Run BUSCO (using a docker)

```
$docker run --rm -e USERID=$UID -u $UID -v /home/rferrari/:/home/rferrari/ -w
/home/rferrari/projetos/collaris/BUSCO_run/genome/post_polishment/NextDenovo/run4_RF_f
inal/SRs ezlabgva/busco:v5.2.2_cv1 busco -i
/home/rferrari/projetos/collaris/BUSCO_run/genome/post_polishment/NextDenovo/run4_RF_f
inal/SRs/genome.nextpolish.fasta -l hymenoptera_odb10 --augustus_species Apis_mellifera -o
run1 -m geno -c 12
```

#To see list of available reference datasets

```
$docker run --rm -e USERID=$UID -u $UID -v /home/rferrari/:/home/rferrari/ -w /home/rferrari
ezlabgva/busco:v5.2.2_cv1 busco --list-datasets
```

GENOME POLISHMENT

5 ****NextPolish (https://github.com/Nextomics/NextPolish)****

Using only short reads

Prepare a 'sgs.fofn' file

```
$ls
/home/fafinha/collaris/reads/genomic_reads/Illumina_shortreads.R1.
fastq
/home/fafinha/collaris/reads/genomic_reads/Illumina_shortreads.R2.
fastq > sgs.fofn
```

Create a 'run.cfg' file

```
[General]
job_type = local
job_prefix = nextPolish
task = best
rewrite = yes
rerun = 3
parallel_jobs = 6
multithread_jobs = 5
genome = /home/fafinha/collaris/mafra/flye_run/run1/assembly.fasta
genome_size = auto
workdir = /home/fafinha/collaris/Nextpolish_run/01_rundir
polish_options = -p {multithread_jobs}

[sgs_option]
sgs_fofn = /home/fafinha/collaris/Nextpolish_run/sgs.fofn
sgs_options = -max_depth 100 -bwa
```

Prepare a .pbs file to run the analysis remotely on Sagarana

```
/programs/NextPolish_n005/nextPolish
/home/fafinha/collaris/Nextpolish_run/run.cfg
```

Using both short and long reads

Prepare a 'sgs.fofn' file

```
$ls
/home/fafinha/collaris/reads/genomic_reads/Illumina_shortreads.R1.
fastq
/home/fafinha/collaris/reads/genomic_reads/Illumina_shortreads.R2.
fastq > sgs.fofn
```

Prepare a 'lgs.fofn' file

```
$ls
/home/fafinha/collaris/reads/genomic_reads/ONT_longreads_rawdata_c
ollaris.fq > lgs.fofn
```

Create a 'run.cfg' file

```
[General]
job_type = local
job_prefix = nextPolish
task = best
rewrite = yes
rerun = 3
parallel_jobs = 8
multithread_jobs = 8
genome = /home/fafinha/collaris/mafra/flye_run/run1/assembly.fasta
genome_size = 300m
workdir =
/home/fafinha/collaris/NextPolish_run/run5/long_short_reads/01_rundir
polish_options = -p {multithread_jobs}

[sgs_option]
sgs_fofn =
/home/fafinha/collaris/NextPolish_run/run5/short_reads/sgs.fofn
sgs_options = -max_depth 100 -bwa

[lgs_option]
lgs_fofn =
/home/fafinha/collaris/NextPolish_run/run5/long_short_reads/lgs.fofn
lgs_options = -min_read_len 1k -max_depth 100
lgs_minimap2_options = -x map-ont
```

Prepare a .pbs file to run the analysis remotely on Sagarana

```
/programs/NextPolish_n005/nextPolish
/home/fafinha/collaris/NextPolish_run/run5/long_short_reads/run.cfg
```