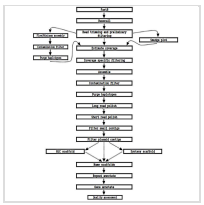

2

Mar 21, 2022

Plant assemble - Plant de novo genome assembly: quality assessment V.2

1

Scott Ferguson[1], Ashley Jones[1], Justin Borevitz[1]

[1]Australian National University

1

dx.doi.org/10.17504/protocols.io.e6nvw578zvmk/v2

Scott Ferguson
Australian National University

With the advancement of long-read sequencing technologies and associated bioinformatics tools, it has now become possible to de novo assemble complex plant genomes with unrivalled contiguity, completeness and correctness. As read lengths can surpass repeat lengths, the ability to assemble genomes de novo has dramatically improved, whereby complex plant genomes of widely variable sizes and repeat content have highly benefited. Despite these improvements, challenges remain in performing de novo assembly, namely in developing a reliable workflow and in tool choice. Here we present a protocol collection of bioinformatic workflows detailing plant genome assembly using Oxford Nanopore Technologies long-reads with a de novo assembler (Canu), syntenic or Hi-C scaffolding, and RNA and/or gene homology-based annotation. We have developed and tested these protocols on multiple plant genomes. Using these protocols with sufficient coverage of long-reads, a highly contiguous, complete, and correct plant genome can be assembled. These genomes can further genomic research into structural variation among groups, and SNP genotyping and association studies among populations.

DOI

dx.doi.org/10.17504/protocols.io.e6nvw578zvmk/v2

Scott Ferguson, Ashley Jones, Justin Borevitz 2022. Plant assemble - Plant de novo genome assembly: quality assessment. **protocols.io** https://dx.doi.org/10.17504/protocols.io.e6nvw578zvmk/v2 Scott Ferguson

Plant assemble - Plant de novo genome assembly, scaffolding and annotation for genomic studies

Updates for publication

protocol ,

Mar 21, 2022

Mar 21, 2022

59657

Part of collection

Plant assemble - Plant de novo genome assembly, scaffolding and annotation for genomic studies

Genome quality assessment

1 Having assembled a genome we want to assess how good it is. The three major considerations here are contiguity, accuracy and completeness.

Contiguity we can assess with numerical statistics:

- N50.
- E-size.
- number of sequences.

N50 and E-size are both ways of trying to express genome contiguity in a single easy to read number. E-size is a better metric but N50 is much more used. A much better way of showing your genome contiguity is with a histogram of sequence sizes, but this is harder to read and communicate.

For more infomation on E-Size see:

- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3290791/
- http://lh3.github.io/2020/04/08/a-new-metric-on-assembly-contiguity

Accuracy and completeness we measure with:

- BUSCO.
- LAI.

BUSCO checks your genome for highly conserved single copy lineage specific genes. Having discovered all BUSCO genes within your genome, this static reports on what percent of these genes have been: found, not-found, duplicated, and fragmented within your assembly. Better genomes have a high percent of single copy genes. LAI scores the proportion of LTR sequences in the genome that are intact, higher is better. To get a feel for this value you will need to read papers that cite this tool and report the genome LAI.

N50

2 Save these commands to N50.sh, make executable, and add to path or place in a directory included in your path. This script works with the .genome file type, see "Long read genome assembly - Step 2".

Citation: Scott Ferguson, Ashley Jones, Justin Borevitz Plant assemble - Plant de novo genome assembly: quality assessment https://dx.doi.org/10.17504/protocols.io.e6nvw578zvmk/v2

To calculate the N50 of a fasta/q file:
1. gne-file.sh XXX.fasta
2. N50.sh XXX.genome

N50.sh

```
#!bin/bash
tmp=`cut -f2 $1 | paste -s -d+ | bc`
half=`echo "${tmp} /2" | bc`
cut -f2 $1 | sort -h | awk -v H=$half 'BEGIN{cumsum = 0} {if(cumsum < H){cumsum = cumsum + $1; last = $1} else {print last; exit}}'
```

E-size

3 Save these commands to ESize.sh, make executable, and add to path or place in a directory included in your path. To calculate E-size call E-size.sh genome.fasta

Save these commands to E-size.sh, make executable, and add to path or place in a directory included in your path. This script works with the .genome file type, see "Long read genome assembly - Step 2".

To calculate the E-size of a fasta/q file:
1. gne-file.sh XXX.fasta
2. E-size.sh XXX.genome

E-size.sh

```
#!/bin/bash
awk 'BEGIN{Lj = 0; Lc = 0} {Lj = Lj + $2; Lc = Lc + ($2 * $2);}
END{printf "%i\n", (Lc/Lj)+0.5}' $1
```

BUSCO

4 To run BUSCO you will need to first download the lineage specific BUSCO set for your species from https://busco-data.ezlab.org/v4/data/lineages/.

Note: newer versions of BUSO will auto download gene sets, but we manually download ours so that we can save them to appropriate disk space. If you use the auto download within BUSCO your commands will be different, see https://busco.ezlab.org/.

Citation: Scott Ferguson, Ashley Jones, Justin Borevitz Plant assemble - Plant de novo genome assembly: quality assessment https://dx.doi.org/10.17504/protocols.io.e6nvw578zvmk/v2

BUSCO

```
BUSCO="/path to BCUSCO gene set/lineage_db"
genome="/path to genome/genome.fasta"
label="species"
cpus=XX

run_busco -l $BUSCO -m geno -c $cpus \
 -i $genome.fasta \
 -o $label
```

LAI

5

LAI

```
genome="/path to genome/genome.fasta"
label="speciesName"
cpus=XX

/path to genome tools/bin/gt suffixerator -db $genome -indexname
$label -tis -suf -lcp -des -ssp -sds -dna
/path to genome tools/bin/gt ltrharvest -index $label -minlenltr 100 -
maxlenltr 7000 -mintsd 4 -maxtsd 6 -motif TGCA -motifmis 1 -similar 85
-vic 10 -seed 20 -seqids yes > ${label}.harvest.scn
/path to LTR Finder/LTR_FINDER_parallel/LTR_FINDER_parallel -seq
$genome -threads $cpus -harvest_out -size 1000000 -time 300
cat ${label}.harvest.scn $(basename $genome).finder.combine.scn >
${label}.rawLTR.scn

/path to LTR_retriever/LTR_retriever -genome $genome -inharvest
${label}.rawLTR.scn -threads $cpus
/path to LTR_retriever/LAI -genome $genome -intact
${genome}.pass.list -all ${label}.out -t $cpus
```

Validation reads

Citation: Scott Ferguson, Ashley Jones, Justin Borevitz Plant assemble - Plant de novo genome assembly: quality assessment https://dx.doi.org/10.17504/protocols.io.e6nvw578zvmk/v2

6 An additional quality check we like to perform is to align our unused long reads to our finalised genome and look at the mapping percent and quality. A very high proportion of reads will align with good quality scores to a high quality genome.

The validation read set is made from all reads not used in assembly with an average quality score of 7 or greater. We don't wish to align low quality reads as these will not align well due to sequencing errors and be a poor indicator of genome quality.

Create validation reads fastq

```
allReads="/path to reads/all.fastq.gz"
assemblyReads="/path to reads/assembly.fastq.gz"

# clean up reads, only remove reads smaller than 200.
zcat $allReads | \
   /path to Nano Pack/NanoLyse -r /path to DNA control
strand/DNA_CS.fasta | \
   /path to Nano Pack/NanoFilt --headcrop 200 --tailcrop 200 | \
   /path to Nano Pack/NanoFilt -q 7 -l 200 | gzip > q7.fastq.gz

# get list of read names, and filter allReads to get validation set
bioawk -c fastx '{$name}' q7.fastq.gz > q7.lst
bioawk -c fastx '{$name}' $assemblyReads > assembly.lst
zgrep -vwf assembly.lst < q7.lst > validationReads.lst
seqtk subseq q7.fastq.gz validationReads.lst | gzip >
validationReads.fastq.gz
```

7 Align validation reads to genome and validate alignment quality with Qualimap.

Citation: Scott Ferguson, Ashley Jones, Justin Borevitz Plant assemble - Plant de novo genome assembly: quality assessment https://dx.doi.org/10.17504/protocols.io.e6nvw578zvmk/v2

Align and validate genome

```
genome="/path to genome/genome.fasta"
reservedReads="/path to reads/validationReads.fastq.gz"
cpus=XX

minimap2 -t $cpus -ax map-ont $genome $reservedReads \
   | samtools sort -@ $cpus > sorted_mapped_reads.bam

qualimap bamqc -bam sorted_mapped_reads.bam -outdir
qualimap_results -outformat PDF:HTML --java-mem-size=15G -nt $cpus
```

Qualimap will produce a pdf formatted report. Use this to get statistics on

- number of reads
- mapped reads
- unmapped reads
- average coverage
- standard deviation of coverage
- mean mapping quality

These statistics can be used to assess genome quality. Average coverage should ~= read coverage (to calculate see below), and the proportion and quality of mapping should be high.

Calculate read coverage

```
genomeSize=5000
reservedReads="/path to reads/validationReads.fastq.gz"

gne-file.sh $reservedReads
echo $(cut -f2 $(basename $reservedReads .fastq.gz).genome | paste -s
-d+ | bc)/${genomeSize} | bc
```


Citation: Scott Ferguson, Ashley Jones, Justin Borevitz Plant assemble - Plant de novo genome assembly: quality assessment https://dx.doi.org/10.17504/protocols.io.e6nvw578zvmk/v2