



2 ▼

Mar 21, 2022

Plant assemble - Plant de novo genome assembly: annotation V.2



1

Scott Ferguson¹, Ashley Jones¹, Justin Borevitz¹¹Australian National University

1

dx.doi.org/10.17504/protocols.io.kqdg36d87g25/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.kqdg36d87g25/v2

Scott Ferguson, Ashley Jones, Justin Borevitz 2022. Plant assemble - Plant de novo genome assembly: annotation. **protocols.io**
<https://dx.doi.org/10.17504/protocols.io.kqdg36d87g25/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

59655

Part of collection

[Plant assemble - Plant de novo genome assembly, scaffolding and annotation for genomic studies](#)

Repeat annotate

1

Now that your genome is complete we want to find and annotate it for repeat regions. To do this we use EDTA and RepeatMasker. EDTA is first used to build a transposon (TE) sequence library. Next, RepeatMasker is used to find all TE and simple repeats within the genome.

Repeat mask

```
genome="/path to genome/genome.fasta"  
cpus=XX  
mkdir EDTA  
cd EDTA  
EDTA.pl --genome $genome --anno 1 --threads $cpus  
cd ../  
RepeatMasker -pa $cpus -s -lib EDTA/$(basename  
$genome).mod.EDTA.TElib.fa -dir repeatMask -e ncbi $genome
```

Repeat annotate: soft mask

2

Output from RepeatMasker will include a hard masked genome (all nucleotides within repeat regions will be changed to "N") among other outputs. We will use the repeat regions defined within the out file and BEDTools to create a soft masked version of your genome (all nucleotides within repeat regions will be changed to lowercase, i.e. A becomes a, C becomes c, etc).

First, we create a bed file from the out file. Next BEDTools is used to soft mask our genome using the regions within the bed file.

Soft mask

```
genome="/path to genome/genome.fasta"  
out="/path to repeat results/genome.fasta.out"  
label="speciesName"
```

```
sed -e '1,3d' $out | awk -v OFS='\t' '{print $5, $6-1, $7}' | sort -k1,1 -  
k2,2 -V > ${label}.bed  
bedtools maskfasta -soft -fi $genome -bed ${label}.bed -fo ${label}-  
softMasked.fasta
```

Gene annotate

- 3 Gene annotation can be done using RNA and/or gene homology. The tool we use is BRAKER2, which is capable of using both RNA and protein sequences of known genes. BRAKER2 will use your RNA or gene sequences to train a hidden markov model that scans your genome for regions likely to be genes. It will annotate both introns and exons, producing a gtf or gff3 file from which gene nucleotide and protein sequences can be made.

For gene annotation use a soft masked version of your genome, steps 1-2 (above).

Installing BRAKER2 can be hard and is very specific to your compute environment.

Gene annotate: Gene sequences/homology

- 4 From the NCBI (or another gene repository) find your species genus and download all gene amino acid sequences available. We also like to download the Arabidopsis thaliana genes and concatenate them to your genus's genes. Use this gene dataset for gene finding with BRAKER2.

BRAKER2 homology

```
genome="/path to genome/softmasked.fasta"  
geneSet="/path to gene sets/A_thaliana-genusGenes.faa"  
cpus=XX  
label=XXXXX
```

```
braker.pl --genome=${genome} \  
  --prot_seq=${geneSet} \  
  --epmode \  
  --softmasking \  
  --workingdir=/fast disk/species-annotate \  
  --cores=${cpus} \  
  --species=${label} \  
  --gff3
```

Gene annotate: RNA

- 5 Gene annotation with RNA begins by aligning your RNA to your genome. To do this a split read aware aligner is needed. The two current best aligners for this task are STAR and HISAT2 (<https://daehwankimlab.github.io/hisat2/>), both aligners give good results. We prefer to use STAR.

RNA alignment with STAR may benefit from dataset specific parameters. A very useful manual for STAR can be found on the STAR's github; <https://github.com/alexdobin/STAR>

Set "genomeSAindexNbases" to $\min(14, \frac{\log_2(\text{GenomeLength})}{2} - 1)$

STAR - RNA alignment

```
label=""
genome=""
RNAReads="/path to reads/RNA.fastq"
cpus=XX

# index genome
mkdir index
STAR --runThreadN $cpus \
    --runMode genomeGenerate \
    --genomeSAindexNbases XX \
    --genomeDir index \
    --genomeFastaFiles $genome \
    --outTmpDir /fast temp/disk space/

# align RNA reads
STAR --runThreadN $cpus \
    --runMode alignReads \
    --genomeDir index \
    --readFilesIn $RNAReads \
    --outFileNamePrefix /save dir/${label}-RNA \
    --outTmpDir /fast temp/disk space/ \
    --outSAMunmapped Within
```

- 6 Now that your RNA is aligned to your genome, we proceed to gene annotation with BRAKER2.

BRAKER2 RNA

```
label=""
genome="/path to genome/softmasked.fasta"
bam="/path to STAR output/XXX.bam"
cpus=XX

braker.pl --species=${label} \
--genome=${genome} \
--bam=${bam} \
--softmasking \
--workingdir=/fast disk/species-annotate \
--cores=${cpus}
```

Gene annotate: Create protein sequences

- 7 Whether you did your annotation with RNA or gene homology, BRAKER2 will produce a gtf file which describes the location of genes within your genome (for a description of the gtf file format see: <http://asia.ensembl.org/info/website/upload/gff3.html>).

To create both amino acid and nucleotide sequences of your genes run the following:

Create protein sequences

```
genome="/path to genome/softmasked.fasta"
gtf="/path to braker results/braker.gtf"
label=""

/path to BRAKER2/bin/getAnnoFastaFromJoiningenes.py -g $genome -f
$gtf -o $label
```