Command lines used for Genome assembly and annotation:

The provided information offers a detailed account of the steps and commands used in the genome assembly and annotation process. It includes the commands used for cleaning Illumina reads, correcting PacBio subreads, assembling with MaSuRCA, and annotating the genome with Maker2.

Cleaning of Illumina reads using Trimmomatic-v0.36:

- Shotgun library:

```
java -jar trimmomatic-0.36.jar PE -phred33 4-shotgunlibrary_1.fastq 4-
shotgunlibrary_2.fastq 4-shotgun_PE1 4-shotgun_PE1_unpaired 4-shotgun_PE2
4-shotgun_PE2_unpaired LEADING:3 TRAILING:3 SLIDINGWINDOW:4:20 MINLEN:36
```

- Mate-pair libraries

```
java -jar trimmomatic-0.36.jar PE -phred33 1and6-matepair3kb_1.fastq
1and6-matepair3kb_2.fastq 1and6-matepair3kb_PE1 1and6-
matepair3kb_PE1_unpaired 1and6-matepair3kb_PE2 1and6-
matepair3kb_PE2_unpaired LEADING:3 TRAILING:3 SLIDINGWINDOW:4:15 MINLEN:36

java -jar trimmomatic-0.36.jar PE -phred33 3and7-matepair8kb_1.fastq
3and7-matepair8kb_2.fastq 3and7-matepair8kb_PE1 3and7-
matepair8kb_PE1_unpaired 3and7-matepair8kb_PE2 3and7-
matepair8kb_PE2_unpaired LEADING:3 TRAILING:3 SLIDINGWINDOW:4:15 MINLEN:36
```

Pacbio subreads correction using canu-v1.8:

```
unzip Adult_4.zip Adult_4_2.zip Adult_4_3.zip

#Rename Adulto_4_subreads.fasta from each unzipped file to
Adult4_subreads_SMRT1.fasta Adult4_subreads_SMRT2.fasta
Adult4_subreads_SMRT3.fasta, respectively

#Combine subreads fasta files

cat Adult4_subreads_SMRT1.fasta Adult4_subreads_SMRT1.fasta
Adult4_subreads_SMRT1.fasta > All_pacbio_SMRT.fasta

#subreads correction – generates the output file:
allpacbio.correctedReads.fasta.gz

canu useGrid=remote -correct genomeSize=350m -p allpacbio -d allpacbio -
pacbio-raw All_pacbio_SMRT.fasta
```

Assembly using MaSuRCA-v3.2.6:

MaSuRCA was run in the advanced mode, which was recommended for projects involving multiple Illumina runs and long-read data. This mode requires a configuration file comprising two sections – DATA and PARAMETERS. The default parameters were used.

```
#Create a configuration file

#Run masurca to generate the assemble.sh file
MaSuRCA-v3.2.6/bin.masurca configuration_file.txt

#Run the assembly
./assemble.sh
```

Example of a configuration file with *T. diversipes* data:

```
# DATA is specified as type {PE,JUMP,OTHER,PACBIO} and 5 fields:
# 1)two_letter_prefix 2)mean 3)stdev 4)fastq(.gz)_fwd_reads
# 5)fastq(.gz)_rev_reads. The PE reads are always assumed to be
# innies, i.e. --->.<---, and JUMP are assumed to be outties
# <---.--->. If there are any jump libraries that are innies, such as
# longjump, specify them as JUMP and specify NEGATIVE mean. Reverse reads
# are optional for PE libraries and mandatory for JUMP libraries. Any
# OTHER sequence data (454, Sanger, Ion torrent, etc) must be first
# converted into Celera Assembler compatible .frg files (see
# http://wgs-assembler.sourceforge.com)
DATA
#Illumina paired end reads supplied as <two-character prefix> <fragment
mean> <fragment stdev> <forward_reads> <reverse_reads>
#if single-end, do not specify <reverse_reads>
#MUST HAVE Illumina paired end reads to use MaSuRCA
PE= pe 350 50 4-shotgun_PE1  4-shotgun_PE2
#Illumina mate pair reads supplied as <two-character prefix> <fragment
mean> <fragment stdev> <forward_reads> <reverse_reads>
JUMP= m1 3000 450 1and6-matepair3kb_PE1 1and6-matepair3kb_PE1
JUMP= m2 8000 1200 3and7-matepair8kb_PE1 3and7-matepair8kb_PE1
#pacbio OR nanopore reads must be in a single fasta or fastq file with
absolute path, can be gzipped
#if you have both types of reads supply them both as NANOPORE type
PACBIO= allpacbio.correctedReads.fasta.gz
#NANOPORE=/FULL_PATH/nanopore.fa
#Other reads (Sanger, 454, etc) one frg file, concatenate your frg files
into one if you have many
#OTHER=/FULL_PATH/file.frg
#synteny-assisted assembly, concatenate all reference genomes into one
reference.fa; works for Illumina-only data
#REFERENCE=/FULL_PATH/nanopore.fa
END

PARAMETERS
#PLEASE READ all comments to essential parameters below, and set the
parameters according to your project
```

```
#set this to 1 if your Illumina jumping library reads are shorter than
100bp
EXTEND_JUMP_READS=0
#this is k-mer size for deBruijn graph values between 25 and 127 are
supported, auto will compute the optimal size based on the read data and
GC content
GRAPH_KMER_SIZE = auto
#set this to 1 for all Illumina-only assemblies
#set this to 0 if you have more than 15x coverage by long reads (Pacbio or
Nanopore) or any other long reads/mate pairs (Illumina MP, Sanger, 454,
etc)
USE_LINKING_MATES = 0
#specifies whether to run the assembly on the grid
USE_GRID=0
#specifies grid engine to use SGE or SLURM
GRID_ENGINE=SGE
#specifies queue (for SGE) or partition (for SLURM) to use when running on
the grid MANDATORY
GRID_QUEUE=all.q
#batch size in the amount of long read sequence for each batch on the grid
GRID_BATCH_SIZE=500000000
#use at most this much coverage by the longest Pacbio or Nanopore reads,
discard the rest of the reads
#can increase this to 30 or 35 if your reads are short (N50<7000bp)
LHE_COVERAGE=25
#set to 0 (default) to do two passes of mega-reads for slower, but higher
quality assembly, otherwise set to 1
MEGA_READS_ONE_PASS=0
#this parameter is useful if you have too many Illumina jumping library
mates. Typically set it to 60 for bacteria and 300 for the other organisms
LIMIT_JUMP_COVERAGE = 300
#these are the additional parameters to Celera Assembler.  do not worry
about performance, number or processors or batch sizes -- these are
computed automatically.
#CABOG ASSEMBLY ONLY: set cgwErrorRate=0.25 for bacteria and
0.1<=cgwErrorRate<=0.15 for other organisms.
CA_PARAMETERS =  cgwErrorRate=0.15
#CABOG ASSEMBLY ONLY: whether to attempt to close gaps in scaffolds with
Illumina  or long read data
CLOSE_GAPS=1
#number of cpus to use, set this to the number of CPUs/threads per node
you will be using
NUM_THREADS = 16
#this is mandatory jellyfish hash size -- a safe value is
estimated_genome_size*20
JF_SIZE = 200000000
#ILLUMINA ONLY. Set this to 1 to use SOAPdenovo contigging/scaffolding
module.
#Assembly will be worse but will run faster. Useful for very large
(>=8Gbp) genomes from Illumina-only data
SOAP_ASSEMBLY=0
#If you are doing Hybrid Illumina paired end + Nanopore/PacBio assembly
ONLY (no Illumina mate pairs or OTHER frg files).
```

```
#Set this to 1 to use Flye assembler for final assembly of corrected mega-
reads.
#A lot faster than CABOG, AND QUALITY IS THE SAME OR BETTER.
#Works well even when MEGA_READS_ONE_PASS is set to 1.
#DO NOT use if you have less than 15x coverage by long reads.
FLYE_ASSEMBLY=0
END
```

Annotation using Maker2

The annotation process was made following the tutorials below (mostly the 1st one):

1. https://gist.github.com/darencard/bb1001ac1532dd4225b030cf0cd61ce2#repeat-annotation
2. http://weatherby.genetics.utah.edu/MAKER/wiki/index.php/MAKER_Tutorial_for_WGS_Assembly_and_Annotation_Winter_School_2018#Repeat_Masking

Following the 1st tutorial, steps 1 and 2 for creating a species-specific repeat library were skipped. Instead, `model_org=all` was used for the "Repeat Masking" session.

Step 3 – Initial MAKER Analysis

Data files used for the 1st round:

1. *Tetrapedia diversipes* genome assembly
2. Transcriptome of *T. diversipes* – available at: https://github.com/pkfsantos/Diapause_Tetrapedia_diversipes
3. Bee protein sequenced from RefSeq database (April 2019)

`est2genome` and `protein2genome` are set to `1` so that MAKER gene predictions are based on the aligned transcripts and proteins.

The gene models generated in the 1st round are used for the training of Augustus (within BUSCO) and SNAP software in Step 4 (Training Gene Prediction Software).

Step 5 – MAKER with *Ab Initio* Gene Predictors
Data files used for the 2nd round:

1. Transcript, protein, and repeat annotation files generated in the 1st round in replacement of the transcriptome, protein files of RefSeq database, and the `model_org=all` parameter in the Repeat Masking
2. Files generated by SNAP and Augustus in Step 4 – SNAP HMM and the species name for Augustus.

Switch `est2genome` and `protein2genome` to `0` so that gene predictions are based on the Augustus and SNAP gene models.

Steps 4 and 5 were repeated three times. The annotation results from the 3rd round were chosen as the best (larger average gene length and more predicted proteins when compared with the BUSCO Hymenoptera database).

Below is the maker_opts.log file used in the 3rd round of annotation:

```
#-----Genome (these are always required)
genome=illumina_cleaned_pacbio_corrected_2019.fasta #genome sequence
(fasta file or fasta embeded in GFF3 file)
organism_type=eukaryotic #eukaryotic or prokaryotic. Default is eukaryotic

#-----Re-annotation Using MAKER Derived GFF3
maker_gff= #MAKER derived GFF3 file
est_pass=0 #use ESTs in maker_gff: 1 = yes, 0 = no
altest_pass=0 #use alternate organism ESTs in maker_gff: 1 = yes, 0 = no
protein_pass=0 #use protein alignments in maker_gff: 1 = yes, 0 = no
rm_pass=0 #use repeats in maker_gff: 1 = yes, 0 = no
model_pass=0 #use gene models in maker_gff: 1 = yes, 0 = no
pred_pass=0 #use ab-initio predictions in maker_gff: 1 = yes, 0 = no
other_pass=0 #passthrough anyything else in maker_gff: 1 = yes, 0 = no

#-----EST Evidence (for best results provide a file for at least one)
est= #set of ESTs or assembled mRNA-seq in fasta format
altest= #EST/cDNA sequence file in fasta format from an alternate organism
est_gff=illumina_cleaned_pacbio_corrected_round2.all.maker.est2genome.gff
#aligned ESTs or mRNA-seq from an external GFF3 file
altest_gff= #aligned ESTs from a closly relate species in GFF3 format

#-----Protein Homology Evidence (for best results provide a file for at
least one)
protein=  #protein sequence file in fasta format (i.e. from mutiple
oransisms)
protein_gff=illumina_cleaned_pacbio_corrected_round2.all.maker.protein2gen
ome.gff  #aligned protein homology evidence from an external GFF3 file

#-----Repeat Masking (leave values blank to skip repeat masking)
model_org= #select a model organism for RepBase masking in RepeatMasker
rmlib= #provide an organism specific repeat library in fasta format for
RepeatMasker
repeat_protein= #provide a fasta file of transposable element proteins for
RepeatRunner
rm_gff=illumina_cleaned_pacbio_corrected_round2.all.maker.repeats.gff
#pre-identified repeat elements from an external GFF3 file
prok_rm=0 #forces MAKER to repeatmask prokaryotes (no reason to change
this), 1 = yes, 0 = no
softmask=1 #use soft-masking rather than hard-masking in BLAST (i.e. seg
and dust filtering)

#-----Gene Prediction
```

snaphmm=illumina_cleaned_pacbio_corrected_round2.zff.length50_aed0.25.hmm
#SNAP HMM file
gmhmm= #GeneMark HMM file
augustus_species=Tetrapedia_diversipes #Augustus gene prediction species model
fgenesh_par_file= #FGENESH parameter file
pred_gff= #ab-initio predictions from an external GFF3 file
model_gff= #annotated gene models from an external GFF3 file (annotation pass-through)
est2genome=0 #infer gene predictions directly from ESTs, 1 = yes, 0 = no
protein2genome=0 #infer predictions from protein homology, 1 = yes, 0 = no
trna=0 #find tRNAs with tRNAscan, 1 = yes, 0 = no
snoscan_rrna= #rRNA file to have Snoscan find snoRNAs
unmask=0 #also run ab-initio prediction programs on unmasked sequence, 1 = yes, 0 = no

#-----Other Annotation Feature Types (features MAKER doesn't recognize)
other_gff= #extra features to pass-through to final MAKER generated GFF3 file
#-----External Application Behavior Options
alt_peptide=C #amino acid used to replace non-standard amino acids in BLAST databases
cpus=1 #max number of cpus to use in BLAST and RepeatMasker (not for MPI, leave 1 when using MPI)
#-----MAKER Behavior Options
max_dna_len=300000 #length for dividing up contigs into chunks (increases/decreases memory usage)
min_contig=1 #skip genome contigs below this length (under 10kb are often useless)

pred_flank=200 #flank for extending evidence clusters sent to gene predictors
pred_stats=0 #report AED and QI statistics for all predictions as well as models
AED_threshold=1 #Maximum Annotation Edit Distance allowed (bound by 0 and 1)
min_protein=0 #require at least this many amino acids in predicted proteins
alt_splice=0 #Take extra steps to try and find alternative splicing, 1 = yes, 0 = no
always_complete=0 #extra steps to force start and stop codons, 1 = yes, 0 = no
map_forward=0 #map names and attributes forward from old GFF3 genes, 1 = yes, 0 = no
keep_preds=0 #Concordance threshold to add unsupported gene prediction (bound by 0 and 1)

split_hit=20000 #length for the splitting of hits (expected max intron size for evidence alignments)
single_exon=0 #consider single exon EST evidence when generating annotations, 1 = yes, 0 = no
single_length=250 #min length required for single exon ESTs if 'single_exon is enabled'

```
correct_est_fusion=0 #limits use of ESTs in annotation to avoid fusion
genes

tries=2 #number of times to try a contig if there is a failure for some
reason
clean_try=0 #remove all data from previous run before retrying, 1 = yes, 0
= no
clean_up=0 #removes theVoid directory with individual analysis files, 1 =
yes, 0 = no
TMP= #specify a directory other than the system default temporary
directory for temporary files
```