##

(c) 2015-2017 Merly Escalona merlyescalona@uvigo.es

Phylogenomics Lab. University of Vigo.

#

Description:

==========

Pipelines for data simulation for variant calling assesment

Running @ft2.cesga.es

##
#!/bin/bash -l
##

Previous to running the wrapper I had

to set up the perl env.

< o conf mbuildpl_arg '--install_base /home/uvi/be/mef/perl'
cpan> o conf commit
cpan> q
cpan install Math::GSL
MODULE_INSTALL_PERL
##
####

Folder paths

##
####
source $HOME/vc-benchmark-cesga/src/vcs.variables.sh
simphyReplicateID=3
##
####

0. Folder structure

##
####

git clone https://merlyescalona@github.com/merlyescalona/vc-benchmark-cesga.git $HOME/vc-benchmark-cesga

mkdir $folderDATA $folderOUTPUT $folderERROR $folderINFO

##
####

STEP 1. SimPhyvc

##
####
step1JOBID=$(sbatch -a $simphyReplicateID $folderJOBS/vcs.1.simphy.sh | awk '{ print $4}')
##
####

STEP 2. INDELible wrapper

##
####

After the running of SimPhy, it is necessary to run the INDELIble_wrapper

to obtain the control files for INDELible. Since, is not possible to

run it for all the configurations, it is necessary to modify the name of the

output files in order to keep track of every thing

```
################################################################################
####
step2JOBID=$(sbatch -a $simphyReplicateID --dependency=afterok:$step1JOBID
$folderJOBS/vcs.2.wrapper.sh | awk '{ print $4}')
################################################################################
####
```

3. INDELIBLE CALLS

```
################################################################################
####
```

Need to figure out the folder from where I'll call indelilble

Need to filter the species tree replicates that do not have ninds % 2==0

```
numJobs=$(wc -l $HOME/vc-benchmark-cesga/files/${pipelinesName}.$(printf "%05g"
$simphyReplicateID).indelible.folders.txt | awk '{ print $1}')
step3JOBID=$(sbatch -a 1-$numJobs $folderJOBS/vcs.3.indelible.array.sh $simphyReplicateID |
awk '{ print $4}')
################################################################################
####
```

4. ngsphy

```
################################################################################
####
step4JOBID=$(sbatch -a $simphyReplicateID --dependency=afterok:$step3JOBID
$folderJOBS/vcs.4.ngsphy.sh)
```

Possible - Generate Folder structure for art

##
####

4.1 DATA TRANSFERENCE

##
####
rsync -rP $replicateFOLDER/
merly@triploid.uvigo.es:/home/merly/data/$pipelinesName.$replicateID
rsync -rP $LUSTRE/data/ngsphy.data/NGSphy_$pipelinesName.$replicateID/
merly@triploid.uvigo.es:/home/merly/data/NGSphy_$pipelinesName.$replicateID
##
####

4.2 DATA COMPRESSION LUSTRE

##
####
simphyReplicateID=1
replicateID=$(printf "%05g" $simphyReplicateID)
pipelinesName="ssp"
replicateFOLDER="$LUSTRE/data/$pipelinesName.$replicateID"

replicateFOLDER="/home/merly/data/ssp.00001"

for replicate in $(find $replicateFOLDER -maxdepth 1 -mindepth 1 -type d | sort); do
echo "$replicate"
for tree in $(find $replicate -name "g_trees*.trees" | sort); do cat $tree >> $replicate/g_trees.all done echo "Gzipped trees file" gzip $replicate/g_trees.all echo "Removing all g_trees*.trees"
find $replicate -name "g_trees*.trees" | xargs rm
done

```
for replicate in $(find $replicateFOLDER -maxdepth 1 -mindepth 1 -type d | sort); do
echo "$replicate"
mkdir $replicate/FASTA $replicate/TRUE_FASTA
cd $replicate
mv *_TRUE.fasta TRUE_FASTA
mv *.fasta FASTA
tar -czf TRUE_FASTA.tar.gz TRUE_FASTA
tar -czf FASTA.tar.gz FASTA
rm -rf $replicate/TRUE_FASTA
rm -rf $replicate/FASTA
done

##############################################################################
####
```

5. Reference Loci Selection

@ triploid

```
##############################################################################
####
step3.1JOBID=$(qsub -t $simphyReplicateID $HOME/vc-benchmark-
cesga/jobs/vcs.3.1.references.sh | awk '{ print $4}')

##############################################################################
####
```

4. 0

```
#-----------------------------------------------------------------------
```

Compress gene tree files of the replicates into a single gtrees file.

The file will be a tab separated file with the id and the gtree

```
##############################################################################
####
simphyReplicateID=1
replicateID=$(printf "%05g" $simphyReplicateID)
pipelinesName="ssp"
replicateFOLDER="$LUSTRE/data/$pipelinesName.$replicateID"
```

replicateFOLDER="/home/merly/data/ssp.00001"

```
for replicate in $(find $replicateFOLDER -maxdepth 1 -mindepth 1 -type d | sort); do
echo "$replicate"
for tree in $(find $replicate -name "g_trees.trees" | sort); do cat $tree >> $replicate/g_trees.all
done echo "Gzipped trees file" gzip $replicate/g_trees.all echo "Removing all g_trees.trees"
find $replicate -name "g_trees*.trees" | xargs rm
done
for replicate in $(find $replicateFOLDER -maxdepth 1 -mindepth 1 -type d | sort); do
echo "$replicate"
mkdir $replicate/FASTA $replicate/TRUE_FASTA
cd $replicate
mv *_TRUE.fasta TRUE_FASTA
mv *.fasta FASTA
tar -czf TRUE_FASTA.tar.gz TRUE_FASTA
tar -czf FASTA.tar.gz FASTA
rm -rf $replicate/TRUE_FASTA
rm -rf $replicate/FASTA
done

##############################################################################
####
```

4.1 ART

##
####

Need to split the command file. This is because the slurm sysmtem does not

allow me to launch jobs over 1K.

##
####
<<SPLIT_COMMANDS

If staying at LUSTRE, LUSTRE does not allow to launch more than 1000 jobs.

So,if I had to split the files and wait for all the jobs to finish to launch

the following 1000 jobs.

In any case, I'm moving things to triploid,

Way better and faster to run on triploid sequentially

SPLIT_COMMANDS
<<RSYNC

This takes like an hour

rsync -rP $LUSTRE/data/ngsphy.data/NGSphy_ssp.00002/
merly@triploid.uvigo.es:/home/merly/data/NGSphy_ssp.00002

Had to change the names of the paths for the files that were used, since I'm no longer at cesga

cat ssp.00002.sh | sed
's/\/mnt\/lustre\/scratch\/home\/uvi\/be\/mef\/data\/ngsphy.data/\/home\/merly\/data/g' | sed
's/\/home\/uvi\/be\/mef\/vc-benchmark-cesga\/files/\/home\/merly\/csNGSProfile/g' >
ssp.00002.triploid.sh

RSYNC
##
####

Run 1 - PE 150 bp with custom profile

##
####
replicateNum=1
pipelinesName="ssp"
replicatesNumDigits=5
replicateID="$pipelinesName.$(printf "%0${replicatesNumDigits}g" $replicateNum)"
cat ${replicateID}.sh | sed
's/\/mnt\/lustre\/scratch\/home\/uvi\/be\/mef\/data\/ngsphy.data/\/home\/merly\/data/g' | sed
's/\/home\/uvi\/be\/mef\/vc-benchmark-cesga\/files/\/home\/merly\/csNGSProfile/g' >
${replicateID}.triploid.sh
triploidART="/home/merly/data/NGSphy_${replicateID}/scripts/${replicateID}.triploid.sh"
split -l 10000 -d -a 2 ${replicateID}.triploid.sh ${replicateID}.art.commands.
for file in $(ls ${replicateID}.art.commands*); do mv $file "$file.sh"; done for item in $(find*

/home/merly/data/NGSphy_${replicateID}/scripts/ -name "${replicateID}.art.commands" | sort); do
echo $item
qsub $HOME/jobs/vcs.5.art.split.sh $item;
done
##
####

Run 1 - PE 150 bp with custom profile

##
####
replicateNum=1
pipelinesName="ssp"
replicatesNumDigits=5
replicateID="$(printf "%0${replicatesNumDigits}g" $replicateNum)"

ngsphyReplicatePath="$LUSTRE/data/ngsphy.data/NGSphy_${pipelinesName}.${replicateID}"

ngsphyReplicatePath="$HOME/data/NGSphy_${pipelinesName}.${replicateID}"
cat $ngsphyReplicatePath/scripts/${pipelinesName}.${replicateID}.sh | sed
's/\/mnt\/lustre\/scratch\/home\/uvi\/be\/mef\/data\/ngsphy.data/\/home\/merly\/data/g' | sed 's/--
qprof1 \/home\/uvi\/be\/mef\/vc-benchmark-cesga\/files\/csNGSProfile_hiseq2500_1.txt --qprof2
\/home\/uvi\/be\/mef\/vc-benchmark-cesga\/files\/csNGSProfile_hiseq2500_2.txt/ -ss HS25/g' >
$ngsphyReplicatePath/scripts/${pipelinesName}.${replicateID}.triploid.HS25.sh
triploidART="/home/merly/data/NGSphy_${replicateID}/scripts/${replicateID}.triploid.HS25.sh"
cd /home/merly/data/NGSphy_${pipelinesName}.${replicateID}/scripts/
split -l 10000 -d -a 2 ${pipelinesName}.${replicateID}.triploid.HS25.sh
${pipelinesName}.${replicateID}.art.commands.
for file in $(ls
/home/merly/data/NGSphy_${pipelinesName}.${replicateID}/scripts/${pipelinesName}.${replicateI
D}.art.commands*); do mv $file "$file.sh"; done for item in $(find
/home/merly/data/NGSphy_${pipelinesName}.${replicateID}/scripts/ -name
"${pipelinesName}.${replicateID}.art.commands*" | sort); do
echo $item
qsub $HOME/jobs/vcs.5.art.split.sh $item;

```
done

################################################################################
####
```

ORganization of individual reads

```
################################################################################
####
simphyReplicateID=2
ngsphyReplicatePath=$HOME/data/NGSphy_${pipelinesName}.$(printf "%05g"
$simphyReplicateID)

replicates=($(ls $ngsphyReplicatePath/reads))
for replicateST in ${replicates[*]}; do
qsub -t $simphyReplicateID $HOME/vc-benchmark-
cesga/jobs/vcs.6.organization.fq.individuals.sh PE150OWN PAIRED $replicateST
done

################################################################################
###
```

STEP 9. FASTQC

```
################################################################################
####

fqFiles="$fqReadsFolder/${pipelinesName}.allfiles.fastq"
find $fqReadsFolder -name *.fq | xargs cat > $fqFiles

st=1
echo -e "#! /bin/bash
#$ -o $outputFolder/$pipelinesName.8.$st.o
#$ -e $outputFolder/$pipelinesName.8.$st.e
#$ -N $pipelinesName.8.$st

INPUTBASE=$(basename $fqFiles .fastq)

cd $qcFolder/\$INPUTBASE
```

```
$fastqc $fqFiles -o $qcFolder/$INPUTBASE

"> $scriptsFolder/$pipelinesName.8.$st.sh
qsub -l num_proc=1,s_rt=0□00,s_vmem=2G,h_fsize=1G,arch=haswell
$scriptsFolder/$pipelinesName.8.$st.sh
```