



HAL
open science

Analysis of speciation and hybridization in Mediterranean octocorals

Didier Aurelle

► **To cite this version:**

Didier Aurelle. Analysis of speciation and hybridization in Mediterranean octocorals. 2024, 10.5281/zenodo.10966625 . hal-04544432

HAL Id: hal-04544432

<https://hal.science/hal-04544432>

Submitted on 12 Apr 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Analysis of speciation and hybridization in Mediterranean octocorals

Didier AURELLE

Aix-Marseille Université / Institut Pythéas

didier.aurelle@univ-amu.fr

<https://didier-aurelle.pedaweb.univ-amu.fr/>

assembly of transcriptomes with drap and oases

Site of the drap pipeline

<https://www.siginae.org/drap/>

Script on computing cluster

```
#!/bin/bash
#SBATCH -J drap_euni
#SBATCH -o /work/user/daurelle/drap_euni_output.txt
#SBATCH -e /work/user/daurelle/drap_euni_error.txt
#SBATCH -t 10:00:00
#SBATCH --mem=48G
#SBATCH --mail-type=BEGIN,END,FAIL

#Load modules

module load bioinfo/DRAP/1.92
module load bioinfo/Salmon/1.10.0
module load devel/python/Python-3.11.1

cd /work/user/daurelle/Eunicella/

# drap on separate transcriptomes

runDrap --cfg-file /work/user/daurelle/drap.cfg --R1 e-cavol-vil-a_R1.fastq.gz --R2 e-cavol-vil-a_R2.fastq.gz
runDrap --cfg-file /work/user/daurelle/drap.cfg --R1 e-cavol-vil-b_R1.fastq.gz --R2 e-cavol-vil-b_R2.fastq.gz

# runMeta on separate transcriptomes (or assembly on concatenated reads)

runMeta --outdir /work/user/daurelle/Eunicella/Eunicella_meta --drap-dirs /work/user/daurelle/Eunicella/
```

```
module purge
```

sort transcriptomes to remove potential Symbiodiniaceae sequences

First download the sequences to be used as query, here the transcriptome GAKY01.1.fas

Use the `blat_parseil.pl` and `fasta_extract` scripts available here:

https://forgemia.inra.fr/cedric.cabau/drap/-/tree/master/bin?ref_type=heads

```
#!/bin/bash
#SBATCH -p workq
#SBATCH -J blat_popphyl
#SBATCH -o /work/user/daurelle/blat_popphyl.txt
#SBATCH -e /work/user/daurelle/blat_popphyl.txt
```

```
#SBATCH -t 2:00:00 #Acceptable time formats include "minutes", "minutes:seconds", "hours:minutes:seconds"
#SBATCH --mem=48G
#SBATCH --mail-type=BEGIN,END,FAIL
```

```
### SCRIPT FOR GENOBIOINFO
```

```
#Load modules
```

```
module load bioinfo/pblat/2.5.1
```

```
# parallelized blat to look for Symbiodiniaceae sequences
```

```
cd /work/user/daurelle/Eunicella/blat_Eunicella
```

```
pblat -threads=24 /work/user/daurelle/Eunicella/bwa_Eunicella_Symbiodinium/GAKY01.1.fas /work/user/daurelle/Eunicella/blat_Eunicella
```

```
cat Blat_Eunicella_meta_Symb.psl | /work/user/daurelle/blat_parser.pl > ids_Eunicella_meta_Symb.txt
```

```
/work/user/daurelle/fastq_extract_2.pl -f /work/user/daurelle/Eunicella/Eunicella_meta/transcripts_fpkms
```

```
module purge
```

call of SNPs with reads2SNP

bwa2 and Picard SortSam to align raw reads on reference transcriptome objective and prepare bam files for analysis with reads2snps

Link to reads2SNP: <https://kimura.univ-montp2.fr/calcul/software.html>

Example of script

```
#!/bin/bash
#SBATCH -p workq
#SBATCH -J Pic_Eunicella
#SBATCH -o /work/user/daurelle/Pic_Eunicella_NOsymb_output.txt
#SBATCH -e /work/user/daurelle/Pic_Eunicella_NOsymb_error.txt
#SBATCH -t 36:00:00 #Acceptable time formats include "minutes", "minutes:seconds", "hours:minutes:seconds"
#SBATCH --mem=48G
#SBATCH --mail-type=BEGIN,END,FAIL
```

```
### SCRIPT FOR GENOBIOINFO
```

```
#Load modules
```

```
module load bioinfo/bwa-mem2/2.2.1
```

```
module load bioinfo/samtools/1.18
```

```
module load bioinfo/picard-tools/2.20.7
```

```
cd /work/user/daurelle/Eunicella/popphy1/bam_popphy1
```

```
# index reference transcriptome
```

```
bwa-mem2 index -p popphy1_reduit3 transcripts_fpkms_1.fa
```

```

bwa-mem2 mem -t 48 popphyl_reduit3_NO_Symb /work/user/daurelle/Eunicella/popphyl/ECMAR91/ECMAR91.fastq
samtools view -S -b -o ECMAR91_NO_Symb.bam ECMAR91_NO_Symb.sam
samtools flagstat ECMAR91_NO_Symb.bam > ECMAR91_NO_Symb.stats.txt
java -Xmx4g -jar $PICARD SortSam I=ECMAR91_NO_Symb.bam O=ECMAR91_sorted_Picard_NOsymb.bam SORT_ORDER=
# call SNPs with reads2SNP (first prepare bamlist, see documentation)
/work/user/daurelle/reads2snp_2.0.64.bin -nbth 48 -par 1 -tol -th2 0.05 -Fis = 0.5 -bamlist liste_popphyl
module purge

```

demographic inferences with DILS

Site of the DILS software

<https://github.com/popgenomics/DILS>

First prepare the input file and the yaml file as indicated in the documentation.

```
#!/bin/bash
```

```
##### Slurm options #####
```

```
### Job name
```

```
#SBATCH --job-name=DILS_2024_Eunicella
```

```
### Limit run time "days-hours:minutes:seconds"
```

```
#SBATCH --time=24:00:00
```

```
### Requirements
```

```
#SBATCH --partition=fast
```

```
#SBATCH --nodes=1
```

```
#SBATCH --ntasks-per-node=4
```

```
#SBATCH --mem-per-cpu=16GB
```

```
#SBATCH --account=speciation_temperate_gorgonians
```

```
### Email
```

```
#SBATCH --mail-user=didier.aurelle@univ-amu.fr
```

```
#SBATCH --mail-type=ALL
```

```
### Output
```

```
#SBATCH --output=/shared/home/daurelle/DILS2024-%j.out
```

```
#####
```

```
echo '#####'
```

```
echo 'Date:' $(date --iso-8601=seconds)
```

```
echo 'User:' $USER
```

```
echo 'Host:' $HOSTNAME
```

```
echo 'Job Name:' $SLURM_JOB_NAME
```

```
echo 'Job Id:' $SLURM_JOB_ID
```

```
echo 'Directory:' $(pwd)
```

```
echo '#####'
```

```

# modules loading

module load pypy/2.7-5.10.0
module load snakemake/5.3.0
module load r/3.6.3
module load python/2.7

cd /shared/projects/speciation_temperate_gorgonians/

# launch DILS

/shared/home/daurelle/DILS/bin/DILS_2pop.sh /shared/projects/speciation_temperate_gorgonians/cavolini_s

```

structure analysis with snmf (LEA R package)

Site of the LEA R package

<https://www.bioconductor.org/packages/release/bioc/html/LEA.html>

You can use a vcf file produced with reads2SNP

```

# LEA snmf analysis of GBS data on Ophrys Seedek et al
# if required install LEA R package
if (!require("BiocManager", quietly = TRUE))
  install.packages("BiocManager")
BiocManager::install("LEA")

library(LEA)
library(RColorBrewer)
setwd("~/Nextcloud2/Speciation_Eunicella")

# analysis all samples

Eunicella_geno <- vcf2geno("Eunicella_NO_symb_all_0_1kb.recode.vcf")
Eunicella.at = snmf(Eunicella_geno, K = 1:10, ploidy = 2, entropy = TRUE, CPU = 1, repetitions = 10, pr

IndEunicella <- read.csv("liste_pops_Eunicella.txt", sep = "", header = TRUE)

# check cross entropy

plot(Eunicella.at, col="blue4", cex=1.4, pch=19, cex.lab= 1.5)

# plot replicates

par(mfrow=c(2,5))
for(i in 1:10){ qmatrix = Q(Eunicella.at, K = 2, run = i)
barplot(t(qmatrix))}

for(i in 1:10){ qmatrix = Q(Eunicella.at, K = 3, run = i)

```

```

barplot(t(qmatrix))}

for(i in 1:10){ qmatrix = Q(Eunicella.at, K = 4, run = i)
barplot(t(qmatrix))}

for(i in 1:10){ qmatrix = Q(Eunicella.at, K = 5, run = i)
barplot(t(qmatrix))}

# plot best replicate for each K value; example for K = 4

par(mfrow=c(1,1))
par(mar=c(11,4,4,4))
my_colors = c("orange", "blue", "green", "grey")

tiff("snmf_Eunicella_K4.tiff", width = 3000, height = 1500, units = "px", res=300)
bestEunicella = which.min(cross.entropy(Eunicella.at, K=4))
qmatrix = Q(Eunicella.at, K = 4, run = bestEunicella)
barplot(t(qmatrix), names.arg = IndEunicella[,1], cex.names = 1, las = 2, col = c("green", "orange", "grey"),
dev.off()

```

principal component analysis with the adegenet R package

Site of the adegenet R package

<https://cran.r-project.org/web/packages/adegenet/index.html>

You can use a vcf file produced with reads2SNP

PCA on local computer

```

library(ape)
library(ade4)
library(adegenet)
library(pegas)
library(RColorBrewer)

setwd("~/Nextcloud2/Speciation_Eunicella")

# import individuals list and dataset

popsEunicella <- read.csv("popfile_Eunicella.txt", header = FALSE, sep = "")
View(popsEunicella)
Eunicella_vcf <- read.vcf("Eunicella_NO_symb_all_0_1kb.recode.vcf", from = 1, to = 30000)
Eunicella_ind <- loci2genind(Eunicella_vcf)

# define pops in the genind file and check order
pop(Eunicella_ind) <- popsEunicella[,2]
Eunicella_ind$pop

couls <- c("orange", "blue", "green", "grey")

# perform PCA

```

```

# here no missing data
pcaEunicella <- dudi.pca(Eunicella_ind, scale=FALSE, nf = 10, scannf = FALSE)

summary(pcaEunicella)

# plot the results
s.class(dfxy = pcaEunicella$li, fac = pop(Eunicella_ind), xax = 1,
        yax = 2, cstar = 0, cellipse = 0, clabel = 1, cpoint = 6, pch = 19, col = couls, grid = FALSE)

tiff("PCA_Eunicella_axis1-2.tiff", width = 3000, height = 1500, units = "px", res=300)
s.class(dfxy = pcaEunicella$li, fac = pop(Eunicella_ind), xax = 1,
        yax = 2, cstar = 0, cellipse = 0, clabel = 0, cpoint = 6, pch = 21,
        col = couls, grid = FALSE)
dev.off()

tiff("PCA_Eunicella_axis3-4.tiff", width = 3000, height = 1500, units = "px", res=300)
s.class(dfxy = pcaEunicella$li, fac = pop(Eunicella_ind), xax = 3,
        yax = 4, cstar = 0, cellipse = 0, clabel = 0, cpoint = 6,
        col = couls, grid = FALSE)
dev.off()

```