#####Genome survey and assembly#####

1. Quality Control

fastp -q 10 -u 50 -y -g -Y 10 -e 20 -l 100 -b 150 -B 150 -i Unknown\_AW313-01R0002\_good\_1.fq.gz -I Unknown\_AW313-01R0002\_good\_2.fq.gz -o output\_R1.fq -O output\_R2.fq

2. Assembling the mitochondrial genome

get\_organelle\_from\_reads.py -1 Unknown\_AW313-01R0002\_good\_1.fq.gz -2 Unknown\_AW313-01R0002\_good\_2.fq.gz -R 10 -k 21,45,65,85,105 -F animal\_mt -o z189\_mt\_out

3. Genome survey

jellyfish count -C -m 19 -s 2G -t 10 -o mer\_counts.jf input\_R1.fq input\_R2.fq

4. Genome assembly

hifiasm -l 2 -n 4 -o z189 -t 15 ~/z189/hifi/BMK220516-AW313-02P0002/cell/BMK220516-AW313-02P0002.ccs.fastq.gz

sh gfa2fasta.sh z189.bp.p\_ctg.gfa > z189.fasta

quast z189.fasta

5. Redundancy removal

minimap2 -x asm20 –t 50 z189.fasta ~/z189/hifi/BMK220516-AW313-02P0002/cell/BMK220516-AW313-02P0002.ccs.fastq.gz | gzip -c - > z189.paf.gz

pbcstat z189.paf.gz

calcuts PB.stat > cutoffs 2> calcults.log

split\_fa z189.fasta > asm.split

minimap2 -t 50 -xasm5 -DP asm.split asm.split | gzip -c - > asm.split.self.paf.gz

purge\_dups -2 -T cutoffs -c PB.base.cov asm.split.self.paf.gz > dups.bed 2> purge\_dups.log

get\_seqs -e dups.bed z189.fasta

quast purged.fa

6. Integrity Assessment

busco -i genome.nextpolish.fasta -l ~/z189/02genome\_assemble/without-hic/fasta\_asm/nextpolish/01\_rundir/eukaryota\_odb10/ -m genome -o busco\_out\_fly -c 30

busco -i genome.nextpolish.fasta -l ~/z189/02genome\_assemble/without-hic/fasta\_asm/nextpolish/01\_rundir/metazoa\_odb10/ -m genome -o busco\_out\_fly -c 30

#####Genome annotation##########

1. Repetitive sequences

trf ../purged.fa 2 7 7 80 10 50 500 -f -d -h -r

perl ~/scripts/repeat\_to\_gff.pl purged.fa.2.7.7.80.10.50.500.dat

BuildDatabase -name z189 -engine ncbi ../purged.fa

RepeatModeler -pa 20 -database z189 -engine ncbi

RepeatMasker ../purged.fa -lib z189-families.fa -e ncbi -pa 40 -poly -html -gff -dir ./denove/

python famdb.py -i Libraries/RepeatMaskerLib.h5 lineage -ad Anthozoa

python famdb.py -i Libraries/RepeatMaskerLib.h5 families -f embl -a -d Anthozoa > Anthozoa.embl

buildRMLibFromEMBL.pl Anthozoa.embl > Anthozoa.fasta

cat ~/miniconda3/envs/buscopy3.9/share/RepeatMasker/Anthozoa.fasta ../z189-families.fa > repeat\_for\_anno.fasta

RepeatMasker -a -xsmall -nolow -norna -html -gff -dir ./xsmall\_output -lib repeat\_for\_anno.fasta -e ncbi -pa 50 -poly ../../purged.fa

2. Noncoding RNAs with three methods

gunzip Rfam.cm.gz

cmpress Rfam.cm

cmscan -Z 1650 --cut\_ga --rfam --nohmmonly --fmt 2 --tblout sample.tblout -o sample.result --clanin Rfam.clanin Rfam.cm ../purged.fa.masked

perl infernal-tblout2gff.pl --cmscan --fmt2 sample.tblout >sample.infernal.ncRNA.gff3

awk 'BEGIN{OFS="\t";}{if(FNR==1) print "target\_name\taccession\tquery\_name\tquery\_start\tquery\_end\tstrand\tscore\tEvalue"; if(FNR>2 && $20!="=" && $0!~/^#/) print $2,$3,$4,$10,$11,$12,$17,$18; }' sample.tblout >sample.tblout.xls

cat rfam.txt | awk 'BEGIN {FS=OFS="\t"}{split($3,x,";");class=x[2];print $1,$2,$3,$4,class}' > rfam\_anno.txt

awk 'BEGIN{OFS=FS="\t"}ARGIND==1{a[$2]=$5;}ARGIND==2{type=a[$1]; if(type=="") type="Others"; count[type]+=1;}END{for(type in count) print type, count[type];}' rfam\_anno.txt sample.tblout.xls >sample.ncRNA.statistic

grep "miRNA" rfam\_anno.txt |cut -f1 >miRNA.tem

grep -f miRNA.tem sample.tblout.xls >miRNA.txt

awk '{sum += (int($5) - int($4) >= 0 ? int($5) - int($4) : int($4) - int($5)) + 1} END {print sum}' miRNA.txt

perl /project/tianzhenWu/software/rnammer/rnammer -S euk -multi -m lsu,ssu,tsu -gff rRNA.gff2 -f rRNA.fasta -h rRNA.hmmreport -xml rRNA.xml /project/tianzhenWu/z189/03genome\_annotation/03ncrna/purged.fa.masked

tRNAscan-SE -E -o tRNA.out -f tRNA.ss -m tRNA.stats ../purged.fa.masked

3. Protein-coding genes with three methods

##RNA-seq–based##

hisat2-build ../purged.fa.masked z189.genome.index

hisat2 -p 20 --dta --no-mixed -x z189.genome.index -1 output.R1.fq -2 output.R2.fq --no-unal -S z189.sam 2>z189.summary.txt

samtools view -b z189.sam -o z189.bam

samtools sort z189.bam -o z189.sort.bam

stringtie -p 20 -o 02out.gtf z189.sort.bam

gtf\_genome\_to\_cdna\_fasta.pl 02out.gtf ../purged.fa.masked > transcripts.fasta

gtf\_to\_alignment\_gff3.pl 02out.gtf > 02out.gff3

TransDecoder.LongOrfs -t transcripts.fasta

wget ftp://ftp.uniprot.org/pub/databases/uniprot/current\_release/knowledgebase/complete/uniprot\_sprot.fasta.gz

gunzip uniprot\_sprot.fasta.gz

diamond makedb --in uniprot\_sprot.fasta --db uniprot\_sprot.fasta

diamond blastp -d uniprot\_sprot.fasta -q transcripts.fasta.transdecoder\_dir/longest\_orfs.pep --evalue 1e-5 --max-target-seqs 1 > blastp.outfmt6

TransDecoder.Predict -t transcripts.fasta --retain\_blastp\_hits blastp.outfmt6

cdna\_alignment\_orf\_to\_genome\_orf.pl transcripts.fasta.transdecoder.gff3 02out.gff3 transcripts.fasta > transcripts.fasta.transdecoder.genome.gff3

##homology-based##

miniprot -t8 -d ref.mpi ../purged.fa.masked

miniprot -t8 -I --gff ref.mpi protein.faa > out.gff3

python ~/scripts/miniprot\_GFF\_2\_EVM\_GFF3.py out.gff3 > out\_evm.gff3

GeMoMa GeMoMaPipeline threads=10 outdir=`pwd` GeMoMa.Score=ReAlign AnnotationFinalizer.r=NO o=true t=scaffold.fa a=Genome1.gff g=Genome1.fa

java -jar GeMoMa-1.9.jar CLI GeMoMaPipeline threads=4 outdir=output GeMoMa.Score=ReAlign AnnotationFinalizer.r=NO o=true t=target.fa a=ref.gff g=ref.fa

sh ~/miniconda3/pkgs/gemoma-1.9-hdfd78af\_0/share/gemoma-1.9-0/pipeline.sh tblastn purged.fa.masked.fas genomic.gff dgig.fna 5 ./result FR\_UNSTRANDED ../../02Transcriptome/z189.bam

perl ~/scripts/GeMoMa\_gff\_to\_gff3.pl final\_annotation.gff > final\_annotation\_evm.gff

##de novo##

gffread genomic.gff -g GCF\_004324835.1\_DenGig\_1.0\_genomic.fna -y dgig.pep

gff2gbSmallDNA.pl genomic.gff GCF\_004324835.1\_DenGig\_1.0\_genomic.fna 1000 gene.raw.gb

etraining --species=generic --stopCodonExcludedFromCDS=false gene.raw.gb 2> train.err

cat train.err | perl -pe 's/.\*in sequence (\S+): .\*/$1/' >badgenes.lst

filterGenes.pl badgenes.lst gene.raw.gb > genes.gb

grep '/gene' genes.gb |sort |uniq |sed 's/\/gene=//g' |sed 's/\"//g' |awk '{print $1}' >geneSet.lst

seqkit grep -f geneSet.lst dgig.pep >geneSet.lst.fa

makeblastdb -in geneSet.lst.fa -dbtype prot -parse\_seqids -out geneSet.lst.fa

blastp -db geneSet.lst.fa -query geneSet.lst.fa -out geneSet.lst.fa.blastp -evalue 1e-5 -outfmt 6 -num\_threads 8

awk '$3 > 70 && $1 != $2 {print $2}' geneSet.lst.fa.blastp | sort | uniq > filtered\_lines.txt

awk 'NR==FNR{a[$0];next} !($0 in a)' filtered\_lines.txt genomic.gff > gene\_filter.gff3

gff2gbSmallDNA.pl gene\_filter.gff3 GCF\_004324835.1\_DenGig\_1.0\_genomic.fna 1000 genes.gb.filter

randomSplit.pl genes.gb.filter 100

new\_species.pl --species=z189

etraining --species=z189 genes.gb.filter.train

augustus --species=z189 genes.gb.filter.test | tee firsttest.out

augustus --species=nematostella\_vectensis genes.gb.filter.test | tee firsttest\_nvec.out

augustus --species=human genes.gb.filter.test | tee firsttest\_human.out

etraining --species=z189 --CRF=1 genes.gb.filter.train

augustus --species=z189 genes.gb.filter.test | tee secondtest.out.withCRF

augustus --species=z189 --gff3=on purged.fa.masked >augustus\_z189.gff

perl ~/scripts/augustus\_gff3\_to\_evm\_gff3.pl augustus\_z189.gff > augustus\_z189\_gene.gff

4. Integration and sequence extraction

perl -p -i -e 's/^#.\*//s' gene\_prediction.gff3 transcript\_alignments.gff3 protein\_alignments.gff3

cat transcript\_alignments.gff3 gene\_prediction.gff3 > gene\_prediction\_combined.gff3

perl EVidenceModeler-master/EvmUtils/partition\_EVM\_inputs.pl --partition\_dir ./partition --genome ../purged.fa.masked --gene\_predictions gene\_prediction\_combined.gff3 --protein\_alignments protein\_alignments.gff3 --segmentSize 100000 --overlapSize 10000 --partition\_listing partitions\_list.out

perl EVidenceModeler-master/EvmUtils/write\_EVM\_commands.pl --partitions partitions\_list.out --genome ../purged.fa.masked --gene\_predictions gene\_prediction\_combined.gff3 --protein\_alignments protein\_alignments.gff3 --output\_file\_name evm.out --weights `pwd`/weights.txt > commands.list"

perl EVidenceModeler-master/EvmUtils/execute\_EVM\_commands.pl commands.list | tee evm\_run.log

perl EVidenceModeler-master/EvmUtils/recombine\_EVM\_partial\_outputs.pl --partitions partitions\_list.out --output\_file\_name evm.out

perl EVidenceModeler-master/EvmUtils/convert\_EVM\_outputs\_to\_GFF3.pl --partitions partitions\_list.out --output\_file\_name evm.out --genome ../purged.fa.masked

find . -regex ".\*evm.out.gff3" -exec cat {} \; | bedtools sort -i - > EVM.all.gff

conda install gffread

conda install bioconda::bioawk

gffread EVM.all.gff -g ../purged.fa.masked -y tr\_cds.fa

bioawk -c fastx 'length($seq) < 50 {print $name}' tr\_cds.fa | sed 's/evm.model.//g' > short\_aa\_gene\_list.txt

grep -v -w -f short\_aa\_gene\_list.txt EVM.all.gff > z189.gff

5. BUSCO assessment

busco -i tr\_cds.fa -l ~/z189/02genome\_assemble/without-hic/fasta\_asm/nextpolish/01\_rundir/eukaryota\_odb10/ -m prot -o busco\_out -c 40

busco -i tr\_cds.fa -l ~/z189/02genome\_assemble/without-hic/fasta\_asm/nextpolish/01\_rundir/metazoa\_odb10/ -m prot -o busco\_out\_m -c 40

6. Functional annotation

wget -c ftp://ftp.ncbi.nih.gov/pub/taxonomy/accession2taxid/prot.accession2taxid.gz

wget -c https://ftp.ncbi.nih.gov/pub/taxonomy/taxdump.tar.gz

tar -zxvf taxdump.tar.gz

cp names.dmp ~/.taxonkit

cp nodes.dmp ~/.taxonkit

taxonkit list --ids 6073 --indent "" > cnidaria.taxid.txt

wc -l cnidaria.taxid.txt

zcat prot.accession2taxid.gz |csvtk -t grep -f taxid -P cnidaria.taxid.txt |csvtk -t cut -f accession.version > cnidaria.taxid.acc.txt

seqkit grep -f cnidaria.taxid.acc.txt nr -o cnidaria.fas

diamond makedb --in cnidaria.fas -d cnidaria\_diamond.dmnd

diamond blastp --db cnidaria\_diamond.dmnd --query z189.pep.fa --out nr.tab --outfmt 6 --sensitive --max-target-seqs 1 --evalue 1e-5 --id 30 --block-size 20 --index-chunks 1

wget ftp://ftp.uniprot.org/pub/databases/uniprot/current\_release/knowledgebase/complete/uniprot\_sprot.fasta.gz ./

gzip -d uniprot\_sprot.fasta.gz

diamond makedb --in uniprot\_sprot.fasta -d uniprot\_diamond.dmnd

diamond blastp --db uniprot\_diamond.dmnd --query ../05function\_annotation/z189.pep.fa --out swissprot.tab --outfmt 6 --sensitive --max-target-seqs 1 --evalue 1e-5 --id 30 --block-size 20 --index-chunks 1

wget https://ftp.ncbi.nih.gov/pub/COG/KOG/fun.txt

wget https://ftp.ncbi.nih.gov/pub/COG/KOG/kog

wget https://ftp.ncbi.nih.gov/pub/COG/KOG/kyva

wget https://ftp.ncbi.nih.gov/pub/COG/KOG/twog

diamond makedb --in kyva.fas -d kog\_diamond.dmnd

diamond blastp --db kog\_diamond.dmnd --query ../05function\_annotation/z189.pep.fa --out kog.tab --outfmt 6 --sensitive --max-target-seqs 1 --evalue 1e-5 --id 30 --block-size 20 --index-chunks 1

#####PSMC analysis##########

bwa index ~/z189/03genome\_annotation/purged.fa

minimap2 -ax map-hifi ../purged.fa BMK220516-AW313-02P0002.ccs.fastq.gz -t 8 > aln.sam

samtools sort -@ 12 -m 20G -o sample\_sort.bam z189.hifi.sam

bcftools mpileup -C 50 -f ../purged.fa sample\_sort.bam | bcftools call -c - | vcfutils.pl vcf2fq -d 3 -D 100| gzip > z189.diploid.fq.gz

fq2psmcfa -q20 z189.diploid.fq.gz > diploid.psmcfa

splitfa diploid.psmcfa > split.psmcfa

seq 100 | xargs -i echo psmc -N25 -t15 -r5 -b -p "4+25\*2+4+6" -o round-{}.psmc split.psmcfa | sh

psmc\_plot.pl -u 3.58e-8 -g 35 -G -p z189 combined.psmc

mash dist purged.fa GCA\_035772405.1\_ASM3577240v1\_genomic.fna > log.txt

#####Phylogenetic analysis#####

1. orthofinder

orthofinder -f pep\_files/ -t 36 -S diamond

2. Sequence alignment

java -jar -Xmx600m macse.jar -prog alignSequences -seq cds.fas -allow\_NT "?"

trimal -in macsed.fas -out after-trimal -automated1

perl ./script/02.lst2gene.pl after-trimal.fas 4DTV.gff

3. Phylogenetic reconstruction

iqtree -s connect4Dsites.fa -m MFP -b 100

4. Divergence time

MCMCtree mcmc.ctl

######Control files used by MCMCtree analyses.

###The first run calculates branch length and Hessian information

"

seed = -1

seqfile = connect4Dsites.fa

treefile = input.trees

mcmcfile = mcmc.txt

outfile = out.txt

ndata = 1

seqtype = 0 \* 0: nucleotides; 1:codons; 2:AAs

usedata = 1 \* 0: no data; 1:seq like; 2:normal approximation; 3:out.BV (in.BV)

clock = 2 \* 1: global clock; 2: independent rates; 3: correlated rates

RootAge = '<1.364' \* safe constraint on root age, used if no fossil for root.

model = 4 \* 0:JC69, 1:K80, 2:F81, 3:F84, 4:HKY85

alpha = 0.5 \* alpha for gamma rates at sites

ncatG = 5 \* No. categories in discrete gamma

cleandata = 0 \* remove sites with ambiguity data (1:yes, 0:no)?

BDparas = 1 1 0.1 \* birth, death, sampling

kappa\_gamma = 6 2 \* gamma prior for kappa

alpha\_gamma = 1 1 \* gamma prior for alpha

rgene\_gamma = 2 20 1 \* gammaDir prior for rate for genes

sigma2\_gamma = 1 10 1 \* gammaDir prior for sigma^2 (for clock=2 or 3)

finetune = 1: .1 .1 .1 .1 .1 .1 \* auto (0 or 1): times, musigma2, rates, mixing, paras, FossilErr

print = 1 \* 0: no mcmc sample; 1: everything except branch rates 2: everything

burnin = 2000

sampfreq = 10

nsample = 100000

\*\*\* Note: Make your window wider (100 columns) before running the program.

"

###The second run calculates the divergence time

"

seed = -1

seqfile = connect4Dsites.fa

treefile = input.trees

mcmcfile = mcmc.txt

outfile = out.txt

ndata = 1

seqtype = 0 \* 0: nucleotides; 1:codons; 2:AAs

usedata = 1 \* 0: no data; 1:seq like; 2:normal approximation; 3:out.BV (in.BV)

clock = 2 \* 1: global clock; 2: independent rates; 3: correlated rates

RootAge = '<1.364' \* safe constraint on root age, used if no fossil for root.

model = 4 \* 0:JC69, 1:K80, 2:F81, 3:F84, 4:HKY85

alpha = 0.5 \* alpha for gamma rates at sites

ncatG = 5 \* No. categories in discrete gamma

cleandata = 0 \* remove sites with ambiguity data (1:yes, 0:no)?

BDparas = 1 1 0.1 \* birth, death, sampling

kappa\_gamma = 6 2 \* gamma prior for kappa

alpha\_gamma = 1 1 \* gamma prior for alpha

rgene\_gamma = 2 20 1 \* gammaDir prior for rate for genes

sigma2\_gamma = 1 10 1 \* gammaDir prior for sigma^2 (for clock=2 or 3)

finetune = 1: .1 .1 .1 .1 .1 .1 \* auto (0 or 1): times, musigma2, rates, mixing, paras, FossilErr

print = 1 \* 0: no mcmc sample; 1: everything except branch rates 2: everything

burnin = 2000000

sampfreq = 100

nsample = 1000000

\*\*\* Note: Make your window wider (100 columns) before running the program.

"

#####Selection pressure analysis#####

codeml codeml.ctl

######Control files used by oneratio model

"

seqfile = ../../../gene/cds.fas

treefile = ../../../tree/tree.txt

outfile = tree\_out

noisy = 0

verbose = 1

runmode = 0

seqtype = 1

CodonFreq = 2

clock = 0

aaDist = 0

aaRatefile = dat/jones.dat

model = 0

NSsites = 0

icode = 0

Mgene = 0

fix\_kappa = 0

kappa = 2

fix\_omega = 0

omega = 0.5

fix\_alpha = 1

alpha = 0.

Malpha = 0

ncatG = 8

getSE = 0

RateAncestor = 1

Small\_Diff = .5e-6

cleandata = 0

"

######Control files used by tworatio model

"

seqfile = ../../../gene/cds.fas

treefile = ../../../tree/tree.label

outfile = tree\_out

noisy = 0

verbose = 1

runmode = 0

seqtype = 1

CodonFreq = 2

clock = 0

aaDist = 0

aaRatefile = dat/jones.dat

model = 2

NSsites = 0

icode = 0

Mgene = 0

fix\_kappa = 0

kappa = 2

fix\_omega = 0

omega = 0.5

fix\_alpha = 1

alpha = 0.

Malpha = 0

ncatG = 8

getSE = 0

RateAncestor = 1

Small\_Diff = .5e-6

cleandata = 0

"

######Control files used by branchsite Ma model

"

seqfile = ../../../gene/cds.fas

treefile = ../../../tree/tree.label

outfile = tree\_out

noisy = 9

verbose = 2

runmode = 0

seqtype = 1

CodonFreq = 2

clock = 0

aaDist = 0

aaRatefile = dat/jones.dat

model = 2

NSsites = 2

icode = 0

Mgene = 0

fix\_kappa = 0

kappa = 2

fix\_omega = 0

omega = 1

fix\_alpha = 1

alpha = 0.

Malpha = 0

ncatG = 8

getSE = 0

RateAncestor = 1

Small\_Diff = .5e-6

cleandata = 0

"

######Control files used by branchsite Ma0 model

"

seqfile = ../../../gene/cds.fas

treefile = ../../../tree/tree.label

outfile = tree\_out

noisy = 9

verbose = 2

runmode = 0

seqtype = 1

CodonFreq = 2

clock = 0

aaDist = 0

aaRatefile = dat/jones.dat

model = 2

NSsites = 2

icode = 0

Mgene = 0

fix\_kappa = 0

kappa = 2

fix\_omega = 1

omega = 1

fix\_alpha = 1

alpha = 0.

Malpha = 0

ncatG = 8

getSE = 0

RateAncestor = 1

Small\_Diff = .5e-6

cleandata = 0

"

#####Gene family analysis#####

grep "UTREE 1 =" FigTree.tre | sed -E -e "s/\[[^]]\*\]//g" -e "s/[ \t]//g" -e "/^$/d" -e "s/UTREE1=//" > tree.txt

awk -v OFS="\t" '{$NF=null;print $1,$0}' Orthogroups.GeneCount.tsv |sed -E -e 's/Orthogroup/desc/' -e 's/\_[^\t]+//g' > orthomcl2cafe.tab

perl orthoMCL2cafe.pl Orthogroups.txt > orthomcl2cafe.tab

python ~/scripts/cafetutorial\_clade\_and\_size\_filter.py -i orthomcl2cafe.tab -o gene\_family\_filter.txt -s

cafe5 -i orthomcl2cafe.tab -t tree.txt -p -k 2 -l 0.0001 -o k2p

cafe5 -i orthomcl2cafe.tab -t tree.txt -p -k 3 -l 0.0001 -o k3p

cafe5 -i orthomcl2cafe.tab -t tree.txt -p -k 4 -l 0.0001 -o k4p

cafe5 -i orthomcl2cafe.tab -t tree.txt -p -k 5 -l 0.0001 -o k5p

cat ../Gamma\_change.tab |cut -f1,37|grep "+[1-9]" > cr.expanded

grep "<35>\\*" ../Gamma\_asr.tre > cr\_significant\_trees.tre

grep -E -o "OG[0-9]+" cr\_significant\_trees.tre > cr\_significant.ogs

awk '$2 <0.05 {print $1}' ../Gamma\_family\_results.txt >p0.05\_significant.ogs

grep -f cr\_significant.ogs p0.05\_significant.ogs > cr\_p0.05\_significant.ogs

grep -f cr\_p0.05\_significant.ogs cr.expanded |cut -f1 > cr.expanded.significant

###Extract the list of significantly expanded genes, for example z189

grep -f cr.expanded.significant ~/2023-redo-cafe/OrthoFinder/Results\_Aug27/Orthogroups/Orthogroups.txt|sed "s/ /\n/g"|grep "amil" |sort -k 1.3n |uniq > cr.amil.expanded.significant.genes

seqkit grep -f ../cr.amil.expanded.significant.genes /home/tianzhen/2023-redo-cafe/amil.fasta > cr.amil.expanded.significant.pep.fas

cat \*.fas > all\_sequences\_for\_anno.fas