# ORTHOSKIM: *in silico* gene capture from genomic and transcriptomic libraries

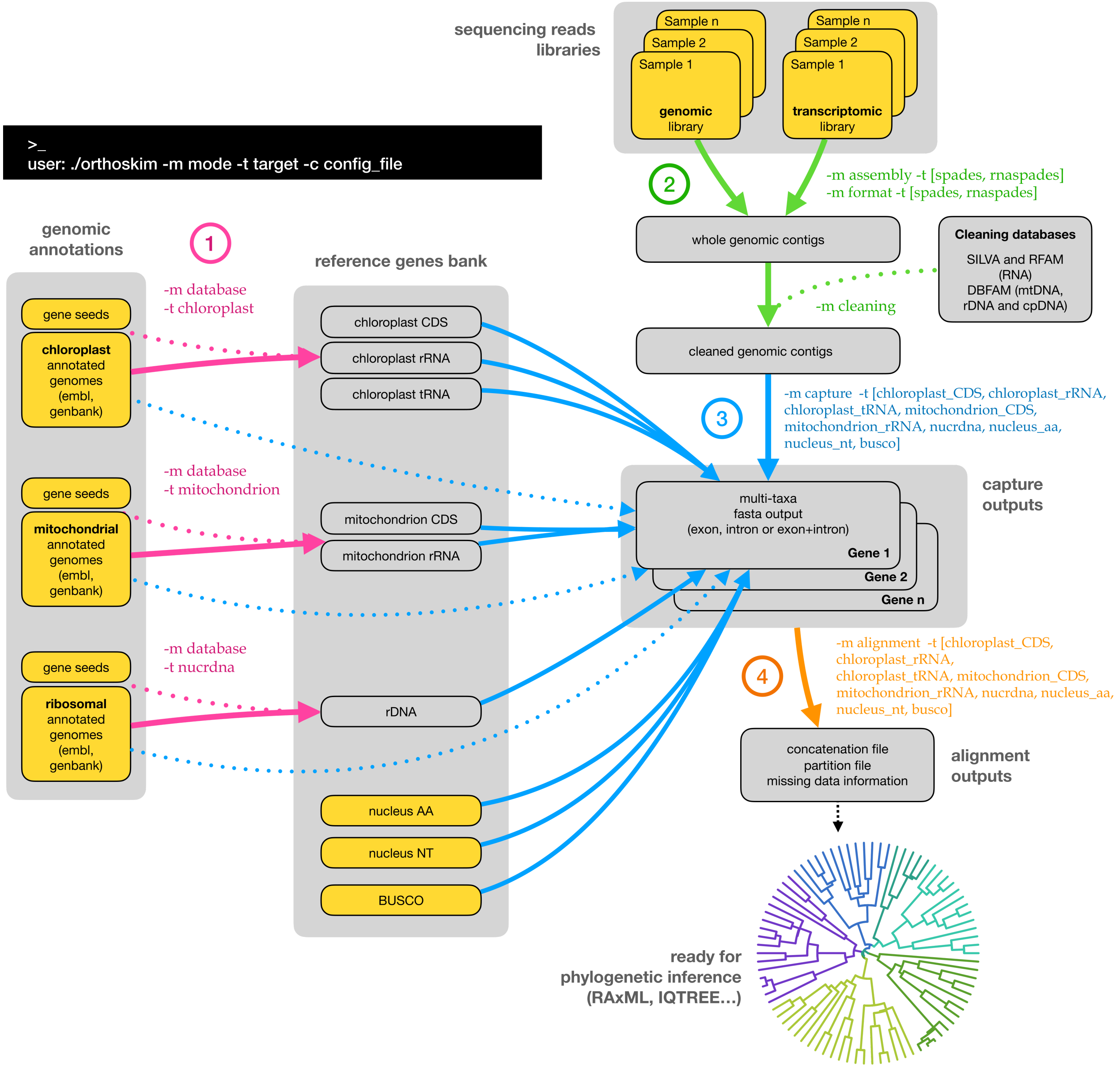
ORTHOSKIM is a pipeline providing different tools to capture targeted genes from genomic and transcriptomic libraries, and to produce phylogenetic matrices for these genes.

This software was developed under the PhyloAlps project.

ORTHOSKIM is a command-line program, that needs to be run from a terminal/console, by calling different modes along with specific targets (see Figure 1), to:

1. produce the gene references database (purple arrrows in Figure 1)
2. perform the contigs assemblies and cleaning from sequencing reads (green arrows)
3. capture the targeted genes (step 3, blue arrows)
4. get taxa alignment of these genes for phylogenetic inference (orange arrows).

### ORTHOSKIM flowchart



```
>_
user: ./orthoskim -m mode -t target -c config_file
```

sequencing reads libraries
Sample n
Sample 2
Sample 1
genomic library
Sample n
Sample 2
Sample 1
transcriptomic library
2
-m assembly -t [spades, rnaspades]
-m format -t [spades, rnaspades]
whole genomic contigs
Cleaning databases
SILVA and RFAM (RNA)
DBFAM (mtDNA, rDNA and cpDNA)
-m cleaning
cleaned genomic contigs
genomic annotations
1
reference genes bank
gene seeds
chloroplast annotated genomes (embl, genbank)
-m database -t chloroplast
chloroplast CDS
chloroplast rRNA
chloroplast tRNA
3
-m capture -t [chloroplast_CDS, chloroplast_rRNA, chloroplast_tRNA, mitochondrion_CDS, mitochondrion_rRNA, nucrdna, nucleus_aa, nucleus_nt, busco]
gene seeds
mitochondrial annotated genomes (embl, genbank)
-m database -t mitochondrion
mitochondrion CDS
mitochondrion rRNA
capture outputs
multi-taxa fasta output (exon, intron or exon+intron)
Gene 1
Gene 2
Gene n
gene seeds
ribosomal annotated genomes (embl, genbank)
-m database -t nucrdna
rDNA
4
-m alignment -t [chloroplast_CDS, chloroplast_rRNA, chloroplast_tRNA, mitochondrion_CDS, mitochondrion_rRNA, nucrdna, nucleus_aa, nucleus_nt, busco]
concatenation file
partition file
missing data information
alignment outputs
nucleus AA
nucleus NT
BUSCO
ready for phylogenetic inference (RAxML, IQTREE...)

> **Fig. 1. ORTHOSKIM workflow**. Yellow boxes represents data that needs to be provided by users. To capture any of the chloroplast, ribosomal or mitochondrial genes, users have to provide each of the three/two annotation genome files if plant/non-plant models are analyzed (see Pipeline description section).

**Applications:** ORTHOSKIM can be run on genomes skims libraries to capture chloroplast, mitochondrial and ribosomal genes. This pipeline can also be run to capture nuclear genes and BUSCO markers from transcriptomic or target sequences capture libraries.

**Citation:**

Pouchon et al. *in prep.* ORTHOSKIM: in silico gene capture from genomic and transcriptomic libraries for phylogenomic and barcoding applications.

## Table of contents

- Installation
- Input files
  - Configuration file
  - Dependencies
  - Sample file
  - References files (database)
- Pipeline description
  - Database (optional)
  - Global assemblies and cleaning
  - genomic/transcriptomic assembly
  - assemblies cleaning
  - Gene capture
  - Selection
    - gene selection
    - contig selection
  - Exon/intron gene prediction
  - gene extraction
  - Summary statistics
  - Alignment of taxa
- Running ORTHOSKIM
  - ORTHOSKIM arguments
  - ORTHOSKIM tutorials
  - databases
  - assemblies and filtering
  - gene capture
  - alignments
- Additional modes for PhyloDB users
  - Sample file
  - List of genes files
  - phyloDB database of references
  - phyloDB extraction from annotations
- Funding
- Support

## 1. Installation

ORTHOSKIM is tested on Unix environment and requires:

- EXONERATE
- SPADES
- DIAMOND
- BLAST
- MAFFT
- TRIMAL
- Needs Awk, Python

Some python libraries are also required, and can be installed via conda *install*:

- ete3==3.0.0b35
- joblib==0.16.0
- numpy==1.19.1
- Bio==0.3.0

ORTHOSKIM is installed from the source code:

```
wget https://github.com/cpouchon/ORTHOSKIM/archive/master.zip
unzip master.zip
cd OrthoSkim-master/
```

## 2. Input files

ORTHOSKIM required a sample file, a config file, and references sequences for targeted regions.

### 2.1. Configuration file

Users have to modify the *config_orthoskim.txt* file provided before running the pipeline. Default values are set for filtering and assembly steps. Indications about the parameters are given for each respective parts in the section 3.

```
nano config_orthoskim.txt
```

```
# ORTHOSKIM (v.1.0) config file
# Global parameters -----------------------------------------------------------------------------------------------------
TOOLS=~/OrthoSkim-master/tools.sh                                                      ## [1] path to file with differebt
RES=~/run_orthoskim                                                                    ## [2] output directory for all OR
EVALUE=0.00001                                                                         ## [3] evalue threshold for mappin
THREADS=15                                                                             ## [4] number of threads to use fo
VERBOSE=0                                                                              ## [5] set verbose to TRUE (1) or
PLANT_MODEL=yes                                                                        ## [6] plants analyzed (yes/no)

# preprocessing the data ------------------------------------------------------------------------------------------------
LIST_FILES=~/OrthoSkim-master/ressources/listSamples.tab                               ## [7] samples table. Specific for

# [assembly] mode -------------------------------------------------------------------------------------------------------
MEMORY=30                                                                              ## [8] max memory used in assembly
KMER=55                                                                                ## [9] Kmer size used in assembly,

# [filtering] mode: Filtering for contaminants in assemblies
SIMILARITY_CONTA_THSLD=65                                                              ## [10] similarity threshold (%) u
MAPPING_CONTA_LENGTH=50                                                                ## [11] minimal value of mapping.
TAXONOMIC_PHYLUM_EXPECTED=Embryophyta                                                  ## [12] taxonomic phylum expected

# [database] mode: sequences of reference -------------------------------------------------------------------------------
MITO_ANNOTATIONS=~/OrthoSkim-master/data/mitochondrion_viridiplantae.gb                ## [13] file with mitochondrial an
NRDNA_ANNOTATIONS=~/OrthoSkim-master/data/nucrdna_viridiplantae.gb                     ## [14] file with nucrdna annotati
CHLORO_ANNOTATIONS=~/OrthoSkim-master/data/chloroplast_viridiplantae.gb                ## [15] file with chloroplast anno
MITO_DB_FMT=genbank                                                                    ## [16] database format: genbank,e
```

```
NRDNA_DB_FMT=genbank                                                                 ## [17] database format: genbank,e
CHLORO_DB_FMT=genbank                                                                ## [18] database format: genbank,e
MITO_SIZE_MIN=200000                                                                 ## [19] minimal size of mitochondr
MITO_SIZE_MAX=1000000                                                                ## [20] maximal size of mitochondr
NRDNA_SIZE_MIN=2000                                                                  ## [21] minimal size of nuclear ri
NRDNA_SIZE_MAX=9000                                                                  ## [22] maximal size of nuclear ri
CHLORO_SIZE_MIN=140000                                                               ## [23] minimal size of chloroplas
CHLORO_SIZE_MAX=200000                                                               ## [24] maximal size of chloroplas
SEEDS_THRESHOLD=0.8                                                                  ## [25] minimal percent of seed co

# [capture] mode: extraction steps from mapping assemblies into a reference -------------------------------------------
MINLENGTH=90                                                                         ## [26] minimal length of alignmen
REFPCT=0.4                                                                           ## [27] minimal fraction of the re
COVERAGE=3                                                                           ## [28] minimal contig coverage (i
MINCONTLENGTH=500                                                                    ## [29] minimal contigs length all
EXO_SCORE=50                                                                         ## [30] minimal score of mapping i
COVCUTOFF=on                                                                         ## [31] coverage cut-off ption for
ORFCOV=0.8                                                                           ## [32] minimal fraction of captur

#---------  [busco] target --------------------------------------------------------------------------------------------
BUSCO_REF=~/OrthoSkim-master/data/BUSCO_viridiplantae.fa                             ## [33] BUSCO sequences (ancestral
BUSCO_TYPE=exon                                                                      ## [34] type of sequence captured:

#---------  [nuclear] target ------------------------------------------------------------------------------------------
NUC_NT_REF=~/OrthoSkim-master/data/nucleusNT_unaligned.fa                            ## [35] nuclear genes of reference
NUC_AA_REF=~/OrthoSkim-master/data/nucleusAA_unaligned.fa                            ## [36] nuclear genes of reference
NUC_TYPE=exon                                                                        ## [37] type of sequence captured:

#---------  [mitochondrion] target ------------------------------------------------------------------------------------
SEEDS_MITO_CDS=~/OrthoSkim-master/ressources/mitoCDS.seeds                           ## [38] mitochondrial CDS seeds se
SEEDS_MITO_rRNA=~/OrthoSkim-master/ressources/mitorRNA.seeds                         ## [39] mitochondrial rRNA seeds s
MITO_REF_CDS=~/OrthoSkim-master/data/mit_CDS_unaligned.fa                            ## [40] mitochondrial coding (CDS)
MITO_REF_rRNA=~/OrthoSkim-master/data/mit_rRNA_unaligned.fa                          ## [41] mitochondrial rRNA non-cod
MITO_TYPE=exon                                                                       ## [42] type of structure extracte

#--------- [chloroplast] target ---------------------------------------------------------------------------------------
SEEDS_CHLORO_CDS=~/OrthoSkim-master/ressources/chloroCDS.seeds                       ## [43] chloroplast CDS seeds sequ
SEEDS_CHLORO_rRNA=~/OrthoSkim-master/ressources/chlororRNA.seeds                     ## [44] chloroplast rRNA seeds seq
SEEDS_CHLORO_tRNA=~/OrthoSkim-master/ressources/chlorotRNA.seeds                     ## [45] chloroplast tRNA seeds seq
CHLORO_REF_CDS=~/OrthoSkim-master/data/chloro_CDS_unaligned.fa                       ## [46] chloroplast coding gene re
CHLORO_REF_rRNA=~/OrthoSkim-master/data/chloro_rRNA_unaligned.fa                     ## [47] chloroplast rRNA gene refe
CHLORO_REF_tRNA=~/OrthoSkim-master/data/chloro_tRNA_unaligned.fa                     ## [48] chloroplast tRNA gene refe
CHLORO_TYPE=exon                                                                     ## [49] type of sequence captured:

#--------- [nucrdna] target -------------------------------------------------------------------------------------------
NRDNA_REF=~/OrthoSkim-master/data/nucrdna_rRNA_unaligned.fa                          ## [50] ribosomal rRNA gene refere
SEEDS_NRDNA=~/OrthoSkim-master/ressources/nucrdna.seeds                              ## [51] ribosomal rRNA seeds seque
NRDNA_TYPE=exon                                                                      ## [52] type of sequence captured:

# [alignment] mode ----------------------------------------------------------------------------------------------------
SELECTION=on                                                                         ## [53] selection of taxa option b
TAXALIST=~/OrthoSkim-master/ressources/selTaxa_Primulaceae.tab                       ## [54] list of taxa to select if 
TRIMMING=on                                                                          ## [55] alignment trimming option 
MISSING_RATIO=1.0                                                                    ## [56] maximal threshold of missi
GENES_TO_CONCAT=~/OrthoSkim-master/ressources/listGenes_To_Concat.tab                ## [57] list of genes which are co

# [checking] mode -----------------------------------------------------------------------------------------------------
BARCODES=( matK rbcL )                                                               ## [58] list of genes used for tax
BARCODES_TYPE=chloroplast_CDS                                                        ## [59] gene location subdirectory
DB_LOCAL=off                                                                         ## [60] option to perform a blast 
BLAST_NT_DB=~/path_to_ntdb/nt                                                        ## [61] location of local NCBI nt 
BLAST_NT_ACCESSION_TAXID=/bettik/pouchon/blastDB/nucl_gb.accession2taxid             ## [62] list of matches between NC
TAXALIST=~/OrthoSkim-master/ressources/selTaxa_Primulaceae.tab                       ## [63] list of taxa for which tax
FAMILIES_LOCAL=off                                                                   ## [64] option to directely use a 
CORRESPONDING_FAMILIES=ecofind_out.tab                                               ## [65] table with query taxid and

# only for phyloskims users -------------------------------------------------------------------------------------------
CHLORO_GENES=~/OrthoSkim-master/ressources/listGenes.chloro                          ## [66] list of chloroplast genes.
MITO_GENES=~/OrthoSkim-master/ressources/listGenes.mito                              ## [67] list of mitochondrial gene
NRDNA_GENES=~/OrthoSkim-master/ressources/listGenes.rdna                             ## [68] list of rdna nuclear genes
```

## 2.2. Dependencies

The path to all dependencies which are required in ORTHOSKIM must be supplied in the *tools.sh* file, using following command:

```
nano tools.sh
```

```
#!/bin/bash

SPADES=/Users/pouchonc/PhyloAlps/OrthoSkim/TOOLS/SPAdes-3.13.0-Darwin/bin/spades.py
DIAMOND=/Users/pouchonc/miniconda2/bin/DIAMOND
EXONERATE=/usr/local/bin/EXONERATE
BLASTDB=/Users/pouchonc/miniconda2/bin/makeBLASTdb
BLASTN=/Users/pouchonc/miniconda2/bin/BLASTn
MAFFT=/path/to/MAFFT
TRIMAL=/path/to/TRIMAL
```

## 2.3. Sample file

A sample file must be supplied in the **$LIST_FILES** tab file (line 7 in *config_orthoskim.txt*).
This tab must contain for each sample the following columns :

- the sample name following *Genus_species_taxid_sampleid_otherids*
- the file-path to forward reads
- the file-path reverse reads

```
head ~/OrthoSkim/ressources/listSamples.tab

Veronica_crassifolia_996476_CAR009639_BGN_NFI   /Users/pouchonc/PhyloAlps/CDS/Veronica_crassifolia:996476/BGN_NFIOSW_4_
Androsace_helvetica_199610_CLA000520_BGN_ETA    /Users/pouchonc/PhyloAlps/CDS/Androsace_helvetica:199610/BGN_ETAOSW_2_1
```

## 2.4. References files (database)

ORTHOSKIM uses a multi-taxa bank of references to capture targeted genes into assemblies for all the different targets of the pipeline (see *3. Pipeline description* below part).

This bank is created in ORTHOSKIM for the *nucrdna*, *chloroplast* and *mitochondrion* targets (purple arrows in Fig. 1), directly from genomic annotations collected by users for each genomic compartment (genbank or embl format required, a single file is set in the config file at lines 13-15). These annotations can be collected directly from the NCBI for example. To achieve this, seeds are required for each type of gene (CDS, rRNA + tRNA for chloroplast) to correctly identify each targeted genes with a standard name (header) as following: `>genename_taxid_Genus_species_other-arguments"` (*e.g.* *>cox1_3702_Arabidopsis_thaliana* for cox1 gene). Location of seeds is given in lines 38-39, 43-45 and 51 of the config file.

ORTHOSKIM creates next a multi-fasta file by given the amino acid sequences for the coding regions (CDS), and nucleotidic sequences for the non-coding regions (*i.e.* rRNA + tRNA only for *chloroplast* target). Location of these output files are set in the *config_orthoskim.txt* file at lines 40-41, 46-48 and 50.

> **NOTE:** As a selection on assemblies is done (see *3.3.1.b.* section), users have to collect all three mitochondrion, chloroplast and nucrdna genome annotations before to run ORTHOSKIM if plant models are analyzed (l.6), or both mitochondrion and nucrdna genomes for other models. All seeds are also required for corresponding regions. Moreover, as a taxonomic selection is done according to the query taxon, we recommend to include as many divergent taxa as possible in the annotations.

Here, an output example of CDS bank generated from mitonchondrial annotations (*i.e.* using the mode `-m database` and the target `-t mitochondrion`).

```
head ~/OrthoSkim/data/mit_CDS_unaligned.fa

>cox2_103999_Codonopsis_lanceolata
MRELEKKNTHDFILPAPADAAEPWQLGFQDGATPIMQGIIDLHHDIFFFLIMILVLVLWILVRALWLFSSKRNPIPQRIVHGTTIEILRTIFPSIILMFIAIPSFALLYSMDEVVVDPA
>cox2_104537_Roya_obtusa
MILKSLFQVVYCDAAEPWQLGFQDAATPMMQGIIDLHHDIMFFITIIITFVLWMLVRVLWHFHYKKNPIPQRFVHGTTIEIIWTIIPSIILMFIAIPSFALLYSMDEVVDPAITIKAIG
>cox2_111617_Ulva_fasciata
MKNFSFSYCILITLFNISVISSCDAPLSATSAMLDRFGFQEPASPLMEGLIALHSDIWAIMLFVAGFVLYMMCAILYNFSASSSEISYKVHHHSLIEIVWTTIPALILCVIAIPSFTLL
>cox1_112509_Hordeum_vulgare_subsp._vulgare
MTNLVRWLFSTNHKDIGTLYFIFGAIAGVMGTCFSVLIRMELARPGDQILGGNHQLYNVLITAHAFLMIFFMVMPAMIGGFGNWFVPILIGAPDMAFPRLNNISFWLLPPSLLLLLSSA
>nad1_119543_Anomodon_attenuatus
MRLYIIGILAKILGIIIPLLLGVAFLVLAERKIMASMQRRKGPNVVGLFGLLQPLADGLKLMIKEPILPSSANLFIFLMAPVMTFMLSLVAWAVIPFDYGMVLSDLNVGILYLFAISSL
```

For the *nucleus_aa* and *nucleus_nt* targets, users have to provide the multi-fasta files of genes, and set their location in the config file to the corresponding sections (lines 35-36 of the config file). The gene name restrictions have to be respected.
For the *busco* target, the multi-fasta file must contain the BUSCO dataset of ancestral sequences in amino acid sequences, called *ancestral_variants* in datasets. The location of this database is given in line 33 of the config file).

Here, an overview of the busco sequences needed:

```
head ~/OrthoSkim/data/BUSCO_viridiplantae.fa

>10018_0
IASVVSEIGLGSEPAFKVPEYDFRSPVDKLQKATGIPKAVFPVLGGLAVGLIALAYPEVLYWGFENVDILLESRPKGLSADLLLQLVAVKIVATSLCRASGLVGGYYAPSLFIGAATGM
>10018_1
VASVVSEIGLGSEPAFKVPEYDFRSAVDSLKKTLGLPKAVLPALGGLIVGLIALAYPEVLYWGFENVDILLESRPRGLSAELLLQLVAVKVVATSLCRASGLVGGYYAPSLFIGAATGM
...
```

By default, ORTHOSKIM is supplied with sequences for plants containing the BUSCO plant set (viridiplantaeae_odb10), 353 UCE designed for angiosperms (Johnson et al., 2018) and a subset of annotations for chloroplast, mitochondrion and nucrdna genomes (in *data/* directory). More annotations can be downloaded as shown in the *4.2 ORTHOSKIM tutorials* section.
Users can easily adapted the files for other models by respecting the recommendations (see documentation).

## 3. Pipeline description

The gene capture is driven on genomic or transcriptomic global assemblies. This allowed to capture from a single assembly run different targeted genes (*e.g.* chloroplast, mitochondrial and ribosomal genes) thanks to mapping of contigs into gene database.

ORTHOSKIM pipeline uses different mode to compute the databases, capture targeted regions, align them between taxa, or to check assemblies (see Figure 1).

> **NOTE**: A *mode_done.log* file is created containing samples that were correctly processed, whereas unprocessed samples were added into *mode_error.log* file. This file could be used to remove processed samples from the initial sample file if the script has to be rerun. Command lines are also print if users want to rerun specific commands on samples.

### 3.1. Database (optional)

ORTHOSKIM provides a mode to create gene database for the mitochondrial, chloroplast and ribosomal regions with `-m database` mode along with `-t mitochondrion, chloroplast, nucrdna` targets (purple arrows in Fig. 1). To do this, genomic annotations for these compartments has to be collected across taxa in a single file and set into the config file.

ORTHOSKIM will then extract all notified CDS, rRNA and tRNA genes and align them into given seeds thanks to *EXONERATE* to keep a standard gene name. Output files (l. 40-41, 46-48 and 50) are created containing a bank of genes, all well identified. Only genes given for the seeds will be included.

> **NOTE**: Users have to collect all three genomes and corresponding seeds to run ORTHOSKIM (or two for non plant model) as a selection is done on contigs thanks to the different genomes (see 3.3.1.b. section). If users want to capture nuclear or busco markers, this step is skipped. In such case, users have to collected genes of reference for these markers into the *config_orthoskim.txt* file, by following instructions for the sequence header.

We also supplied with ORTHOSKIM a function, *SortDB.py*, to reduce the reference datasets of genes and genomes by family (as whole genomes are mapped during the contigs selection step), in order to reduce the computational time of capture (see section 4.2.1).

## 3.2. Global assemblies and cleaning

### 3.2.1. genomic/transcriptomic assembly

Global assemblies are performed for each taxon of the taxa file (l.7) by using SPAdes and have to be run using the `-m assembly -t spades` or `-m assembly -t rnaspades` target, according to the type of library (green arrows in Fig. 1). SPAdes will be run by using the assembly options (**$THREADS**,**$MEMORY**,**$KMER**) specified in the config file (l. 4, 8-9).

ORTHOSKIM will then output a *samplename/* subdirectory into the **${RES}/Assembly/SPADES/** or **${RES}/Assembly/RNASPADES/** given per sample included in the taxa file.

After SPAdes runs, ORTHOSKIM has to preprocess SPAdes scaffolding contigs by renaming the file according to the same sample name provided in the taxa file and ordering them into **${RES}/Assembly/Samples/unfiltered/** directory. This is made under `-m format` mode and `-t spades` or `-t rnaspades` targets according to the version used.

### 3.2.2. assemblies cleaning

The capture of genes will be run only on cleaned assemblies after running the `-m cleaning` mode. This step identifies contigs which are not expected in the assembly dataset and removes them.

To do this, all contigs are mapped with BLAST against rRNA databases SILVA and RFAM supplied in sortmerna (v.4.2.0), composed of the 5S, 5.8S, 16S, 23S, 18S and 28S genes for bacteria, archaea and eukarya. Moreover, contigs are also mapped against to own DBFAM database including a subset of chloroplast, mitochondria and nucrdna genomes for eukarya. The best hits are identified for each contigs, and only contigs mapping onto the expected taxonomy are kept. The expected taxonomy is set by the user at the line 12 (**$TAXONOMIC_PHYLUM_EXPECTED**).

> **NOTE:** Please check the taxonomy provided in the ~/OrthoSkim-master/ressources/rRNA_database_taxonomy.txt file to set a correct phylum (*e.g.* "Embryophyta", "Eumetazoa","Arthropoda","Annelida" etc). We also recommend to keep low values for parameters of **$SIMILARITY_CONTA_THSLD** and **$MAPPING_CONTA_LENGTH** (l. 10-11) as a taxonomic comparison is done between entries in the database.

## 3.3. Gene capture

The capture of targeted genomic regions is made using the `-m capture` mode (blue arrows in Fig. 1), according to three steps:

### 3.3.1. Selection

### 3.3.1.a. gene selection

For all targets (with the exception of BUSCO), ORTHOSKIM will first select the closest reference for each gene and for each taxa from the given database of references.

To achieve this, the selection is made according to the NCBI taxonomy thanks to the taxid number given in the sample name. If the taxid is not valid, ORTHOSKIM will use seeds as references for chloroplast, mitochondrion and nucrdna targets, or the longest sequences for other targets.

For the BUSCO, no selection is made as ancestral variants sequences (already aligned) are used for the reference.

After this, a DIAMOND database is created for each amino acid sequences provided in the retained sequences if CDS are targeted. Otherwise, a BLAST database (*makeBLASTdb* program) is formatted.

### 3.3.1.b. contig selection

Cleaned contigs are selected to reduce the computational time of the following alignments and to correctly identify the right genomic origin of the targeted genes.

To achieve this, for the mitochondrion, chloroplast and nucrdna targets, we identified the contigs by mapping them with BLAST directly on the five closest genomes from the provided annotations for all three genomes in plant models (or both mitochondrion and nucrdna genomes for others). For example, if a contig align more on chloroplast than on mitochondrion or nucrdna, it will be identified as chloroplast. Only genomes with a minimal/maximal size given in **$[MITO,CHLORO,NRDNA]_SIZE** arguments will be considered (lines 19-24 of the config file).

As near to 35% of the ancestral plastid genomes has been estimated to be transferred and conserved in to mitochondrial genomes (Park et al., 2020), this step allows to avoid capturing a mitochondrial copy of a targeted chloroplast gene leading to taxonomic mis-positioning, and *vice versa*. It allows also to attribute the right RNA gene copy to its original cellular compartment.

For the other targets, the selection is performed by mapping the contigs directly on the selected genes by using DIAMOND or BLAST if the sequences are proteic or nucleotidic. A threshold on the kmer coverage (**$COVERAGE**), the contig length (**$MINCONTLENGTH**) and the minimal evalue (**$EVALUE**) is set by users to exclude all contigs below these values for the following step.

### 3.3.2. Exon/intron gene prediction

Gene prediction is conducted from alignment of the selected contigs on the selected genes from EXONERATE by incorporating all the appropriate gaps and frameshifts, and by modelling introns. The *protein2genome* mode is used when CDS are targeted or the *genome2genome* mode for other targets. A *gff* output table is created in **${RES}/Mapping/[nucleus,mitochondrion,chloroplast]/** folder for each sample. Only sequences with a mapping score above the **$EXO_SCORE** value are kept (l. 30 of the config file).
By default we set this score at 50. We recommend to not set too high values (if the gene length is short) as a selection in alignment scores is next performed. Otherwise short genes could be skipped.

> **Note:** Concerning plant models, we performed a second control during the gene alignment to ensure the right origin of organelle. To achieve this, for example, during the chloroplast capture, we align the mitochondrial seeds on selected chloroplast contigs to check if a contig position best align on selected genes than on seeds. This allows to verify if chimeric organelle contig were assembled on the conserved regions and thus wrongly pass the selection of contigs. Seeds of both mitochondrion and chloroplast have to be done by users even if only chloroplast genes will be captured.

### 3.3.3. gene extraction

Targeted genes sequences are extracted from the gff table by identifying the best alignment for each gene, and stored in a fasta file (*e.g.* ycf1.fa). Users can choose to extract exons, introns or both (specified in the config file). A first control is performed by checking that the longest open reading frame (ORF) from the extracted exons of each gene covers at least a minimal fraction of the capture sequence set by users (l. 32 of the contig file; *e.g.* ORFCOV=0.8, meaning that the ORF must cover 80% of the extracted sequence). This step allows taking into account for variations or errors in gene predictions as alternative start codon in proteic sequence of the reference. If such condition is not filled (*e.g.* due to pseudogenes or prediction errors), the sequence is tagged as a gene-like sequence (*e.g.* ycf1-like), and stored apart (*e.g.* ycf1-like.fa file). A coverage cut-off option is also implemented to remove all possible organelle contaminant contigs (*e.g.* alien sequenced DNA resulting in plant-plant contamination), by using a weighted mean standard deviations approach for the contigs coverage adjusted by the reconstructed sizes. We recommend using this option only for genomic libraries and organelles targets. For the nucrdna target, ITS1 and ITS2 barcodes are extracted from the intronic regions of rRNA probes designed during the database step.

Output gene files are created in the **${RES}/Extraction/[mitochondrion,chloroplast]_[CDS,tRNA,rRNA]/** or **${RES}/Extraction/[nucleus_aa,nucleus_nt,nucrdna,busco,uce]/** as following:

```
ls -l ~/RES/Extraction/busco/

-rw-r--r--  1 pouchonc  staff  1758  5 jui 11:11 10104.fa
-rw-r--r--  1 pouchonc  staff  1964  5 jui 11:11 10521.fa
-rw-r--r--  1 pouchonc  staff  5071  5 jui 11:11 10785.fa
-rw-r--r--  1 pouchonc  staff  1400  5 jui 11:11 11487.fa
-rw-r--r--  1 pouchonc  staff  2040  5 jui 11:11 11505.fa
-rw-r--r--  1 pouchonc  staff  1778  5 jui 11:11 1504.fa
```

> **Note:** Once genes were captured, users can use the *checking* mode (-m) on some genes to check the family rank found for these genes for each queried taxa. A BLAST is done on NCBI database and a comparison is made according to the given taxid. Please see required parameters on the config file.
> Users have to download and unzip the corresponding file between accesions and taxids as following:

```
wget ftp://ftp.ncbi.nlm.nih.gov/pub/taxonomy/accession2taxid//nucl_gb.accession2taxid.gz
```

A subdirectory is created **${RES}/Errors/** with a *ValidationSamples.out* file. This file indicates for each taxa and for each gene if the checking is TRUE/FALSE/NA, as following:

```
Abies_alba_45372_PHA000002_RSZ_RSZAXPI000687-79    TRUE    TRUE
Abies_balsamea_90345_TROM_V_43901_CDM_AOZ    TRUE    TRUE
Abies_sibirica_97169_TROM_V_97238_CDM_AVE    TRUE    TRUE
```

> If users want to combine chloroplast_tRNA (e.g. trnL-UAA) and CDS genes (e.g. matK and rbcL), a new directory must be created in the **${RES}/Extraction/** subdirectory with gene files inside; users have next to set the name of this directory in the config file (l. 56).

We also recommend to investigate as well as the reconstructed size and the number of contigs for which targeted genes were extracted to identify spurious taxa (see following section 3.4.b).

## 3.4. Summary statistics

#### a. on assemblies

ORTHOSKIM allows to output summary statistic on cleaned assemblies by using the `-m statistic_assembly` mode.

The output *assemblies_statistics.txt* tab is generated in **${RES}/Statistics/** folder, giving details on the assembly over:

- the taxa name
- the number of cleaned contigs
- the total reconstructed size
- the N50 (*i.e.* the sequence length of the shortest contig at 50% of the total genome length)
- the L50 (*i.e.* the smallest number of contigs whose length sum makes up half of genome size)
- the GC content

```
head ~/RES/Statistics/assemblies_statistics.txt
Actinidia_sp_1927898_FAM000131_BGN_MGF  14691   4768612 600.0   14691   38.05
Adenophora_liliifolia_361368_PHA000132_BGN_NR   106586  17274304        231.0   106586  41.05
Agrostis_canina_218142_TROM_V_92449_BXA_ASB     672     197898  2941.0  672     44.07
Agrostis_vinealis_247443_TROM_V_47532_BXA_ARG   24475   6458884 278.0   24475   36.29
```

Moreover, statistics over contaminant contig identified and removed from assemblies are given in the

**b. on capture**

ORTHOSKIM allows to get statistic from the gene capture by using the `-m statistic_capture` mode for sequences for the different targets (multiple targets can be supplied, *e.g.* `-t mitochondrion_CDS`). The pipeline output a *report.tab* into **${RES}/Statistics/** containing:

- the gene name (gene_name)
- the taxa coverage (taxa)
- the mean length (mean)
- the minimal length of sequence found (minlen)
- the maximal length of sequence found (maxlen)
- the standard deviation (std)
- the 25th percentil (pct25)
- the 50th percentil (pct50)
- the 75th percentil (pct75)

```
head ~/RES/Statistics/chloroplast_CDS_report.log
gene    taxa  mean  min   max   std   pct25   pct50   pct75
rpoC2    7   3316  1831  4152  880   2743    3561    4093
rps19    7   280   273   309   11    276     276     276
ycf1     6  2026  378   5607  1769  820     1346    2462
rpoC1    7   1842  945   2121  413   1795    2058    2092
psbA     7  1059  1059  1059  0     1059    1059    1059
atpI     7  741   741   744   1     741     741     741
rpl2     7  763   483   828   115   792     801     825
ndhH     7  1179  1179  1179  0     1179    1179    1179
rbcL     7  1425  1425  1425  0     1425    1425    1425
```

> **Note**: The full summary statistics of gene capture, as shown in our paper, can be obtained by using the *FullStat.py* function provided in the src/ directory as following:

```
~/OrthoSkim-master/src/FullStat.py -pfind -p Extraction/chloroplast_CDS/ -t chloroplast_CDS_done.log > stat_cp.txt
```

with -p: path where genes are extracted and -t: list of taxa to compute statistics

Moreover, when analyzing genome skims (*i.e.* by targeting chloroplast, mitochondrion or ribosomal genes), we also strongly recommend to investigate the summary statistics of contigs for which genes were captured once the capture is done, by using the function *StatContigs.py* as following:

```
StatContigs.py --path ${RES}/Mapping/ --taxa taxalist --mode [all,chloroplast,mitochondrion,nucrdna] > statistics_captu
```

This function outputs for each taxa and each genomic compartment (according to the `--mode`) the number of contigs assembled along with the total reconstructed size and the mean coverage. By using the `--mode all`, the first three columns of the output table correspond to the chloroplast, the next three to the mitochondrion and the last three to the nucrdna.

```
head statistics_captured_contigs.log
Primula_acaulis_175104_PHA007169_RSZ_RSZAXPI000864-106    26  141628  614.67
Primula_integrifolia_175074_PHA007216_BGN_LG    6   125017  125.8
Primula_kitaibeliana_184184_CLA007221_BGN_MQI    6   126871  309.78
Primula_kitaibeliana_184184_CLA007222_BGN_NND    5   126339  117.18
Primula_latifolia_152139_PHA007223_BGN_LS    5   125006  139.46
Primula_magellanica_175079_CLA010550_GWM_1236    5   126155  172.52
Primula_marginata_175080_PHA007227_BGN_ID    5   124986  192.91
```

This can provides an indication about contaminant that can not be identified during the assembly cleaning (*e.g.* plant-plant contaminant, or host-parasite DNA contaminant). Indeed, for a 150kb chloroplast genome, we except to have a reconstructed size over 125Kb (*i.e.* with only one inverted repeat) as following. In the above example, `Primula_acaulis_175104_PHA007169_RSZ_RSZAXPI000864-106` is doutbut as it shows an higher reconstructed size and number of chloroplast contigs thant what expected. In such case, user can check all genes captured for this sample before to include it on the alignment procedure.

## 3.5. Alignment of taxa

ORTHOSKIM provides a mode to align taxa for each captured genes by using the `-m alignment` mode. We use MAFFT to align each gene individually with the '--adjustdirectionaccurately' option. This alignment can be filtered if the option is chosen by users using TRIMAL with the heuristic 'automated1' method (*on/off* at line 55 of the config file).
In addition, users can choose which taxa will be aligned according to the selection option (*on/off* at line 53 of the config file). In such case, a list of taxa to align has to be given (list stated in l. 54 of the config file).

ORTHOSKIM will output the concatenated alignment of genes along with a partition file under a RAxML-style format suitable for phylogenetic inferences. For such needs, users have to choose which genes will be concatenated (list stated in l. 57 of the config file). A tab with information about gappy or missing data is also produced by sample.

```
-rw-r--r--    1 pouchonc  staff        1341  5 mai 10:41 concatenated.fa
-rw-r--r--    1 pouchonc  staff          21  5 mai 10:41 concatenated.info
-rw-r--r--    1 pouchonc  staff         101  5 mai 10:41 concatenated.missingdata
-rw-r--r--    1 pouchonc  staff          19  5 mai 10:41 concatenated.partitions
```

```
head ~/PATH/concatenated.fa
>Carex_elongata_240685_PHA001842_BGN_MAS
CTTACTATAAATTTCATTGTTGTCGATATTGACATGTAGAAT-GGACTCTCTCTTTATTCTCGTTTGATTTATCA-TCATTTTTTCAATCTAACAAACTCTAAAATGAATAAAATAAAT
>Dipsacus_fullonum_183561_TROM_V_159792_CDM_BFO
CTTACTAAAAATTTCATTGTTGCCGGTATTGACATGTAGAATGGGACTCTATCTTTATTCTCGTCCGATTAATCAGTTCTTCAAAAGATCTATCAGACTATGGAGT-------------
```

```
head ~/PATH/concatenated.info
1    625 trnL-UAA    part1
```

```
head ~/PATH/concatenated.missingdata
Carex_elongata_240685_PHA001842_BGN_MAS    0.0096
Dipsacus_fullonum_183561_TROM_V_159792_CDM_BFO    0.1808
```

```
head ~/PATH/concatenated.partition
DNA, part1 = 1-625
```

> **NOTE**: we recommend visualising gene alignments to be sure that homolog regions were well captured, in particular for plant mtDNA for which some genes have divergent copies. Additional software, such as PREQUAL or SPRUCEUP, can

> be used directly by users to check for correct homology assignment of captured genes.

# 4. Running ORTHOSKIM

ORTHOSKIM uses a command line interface (CLI) that can be accessed through a terminal. Please use the -help (-h) flag to see a description of the main arguments.

```
./orthoskim -h
```

ORTHOSKIM is called step by step. Recommendations about steps are given in the previous description (section 3). After edition of the *tools.sh* and *config_orthoskim.txt* files (with all required files and formats), ORTHOSKIM is called by using the different modes.

We detail instructions here through the description of arguments and the tutorials below.

## 4.1. ORTHOSKIM arguments

**-c (config file):** config file edited by users. See instructions above.

**-m (mode):** different modes encoded in ORTHOSKIM.

> - **alignment:**: Give taxa alignment of selected genes. Each gene are aligned individually with MAFFT and then concatenated. Multiple targets (-t) can be set. A selection of taxa can be performed to decide to which taxa will be align. Alignments can also be trimmed or not.
>   A concatenation and a partition file are generated.

> - **database:** compute the reference bank of gene database for the chloroplast, mitochondrion and nucrdna targets. Annotation needs to be collected in a single file in genbank/embl format. Seeds are required from one organism for each targeted genes using a standard gene name. CDS genes are given in proteic sequences and others in nucleotidic sequences.

> - **capture:** Capture of genes from targeted markers. A selection of the closest reference is made for each gene according to the taxonomy. If errors occurred during this step, ORTHOSKIM will use seeds as reference (exception for busco and nuclear targets). Users has to collected seeds for the targeted genes. Users choose to capture exonic, intronic or both regions.

> - **checking:** Checking of the family rank capture for a given gene with BLAST into the NCBI database.

> - **cleaning:** Cleaning of contigs according BLAST mapping into RNA databases and DBFAM databases. An expected taxonomic level is required to consider as "good" contigs when the best-hit corresponds to this level.

> - **assembly:** Perform global assembly using SPADES assembler. Specificities for assembly are given in the config file (Kmer, memory, threads).

> - **format:** Extract and format the scaffold fasta file for each taxa. A *Samples/* subdirectory is generated containing all taxa contig files.

> - **statistic_assembly:** Compute summary statistics of cleaned assemblies. Informations over the contigs number, the contigs size, the GC content, the N50 value are generated.

> - **statistic_capture:** Compute summary statistics of extraction. A file (target_report.log) is generated including the taxa recovery, the mean size and the range size by gene. Multiple targets (-t) can be set.

**-t (targets):** targeted regions accordint to the mode (-m) used.

> For *database* mode:
>
> - **chloroplast** (creation of chloroplast database containing CDS, rRNA, trnL-UAA genes)
> - **mitochondrion** (creation of mitochondrial database containing CDS and rRNA genes)
> - **nucrdna** (creation of ribosomal database containing rRNA genes and probes for internal spacer regions)

> For *alignment*, *capture* and *stat_capture* modes:
>
> - **busco** (BUSCO markers)
> - **chloroplast_CDS** (coding sequence of chloroplast)
> - **chloroplast_rRNA** (non coding chloroplast rRNA genes)
> - **chloroplast_tRNA** (only tRNA trnL-UAA gene for now)
> - **mitochondrion_CDS** (coding sequence of mitochondrion)
> - **mitochondrion_rRNA** (non coding mitochondrial rRNA genes)
> - **nucleus_aa** (nuclear genes with proteic sequences in references)
> - **nucleus_nt** (nuclear genes with nucleotidic sequences in references)

> For *assembly* and *format* modes:
>
> - **spades** (use of SPADES software to compute genomic assemblies)
> - **rnaspades** (use RNA version of SPADES software to compute transcriptomic assemblies)

## 4.2. ORTHOSKIM tutorials

In this section, we describe a tutorial to capture chloroplast, mitochondrial and ribosomal genes for genome skimming libraries.

### 4.2.1. databases

To begin, users have to install all dependencies, create a sample file, edit the *config_orthoskim.txt* and the *tools.sh* files and collect annotations for the targeted compartments. By default, subsets of genomic annotations are given for *Viridiplantae* to quickly run the software.

Here, we show an example to collect chloroplast annotations for plants from the NCBI.

```
wget -m -np -nd 'ftp://ftp.ncbi.nlm.nih.gov/refseq/release/plastid/' -A.genomic.gbff.gz
gunzip *.genomic.gbff.gz
cat *.genomic.gbff >> plastid.genomic.gb
rm *.genomic.gbff
```

We supplied with ORTHOSKIM a function *AnnotFilter.py* to filter annotations according to taxonomy (*e.g.* viridiplantae). Here, we collected all annotations of viridiplantae.

```
~/OrthoSkim-master/src/AnnotFilter.py -i plastid.genomic.gb -f genbank -l viridiplantae -o ~/OrthoSkim-master/data/chlo
Filtering annotations on taxonomy
1 level(s) of taxonomy set: viridiplantae
      parsing annotations [............................................................] 100 %
4869 / 5201 annotations selected on taxonomy
```

> **NOTE:** the output (given with **-o**) has to be the same which is set in the config file (line 15: **CHLORO_ANNOTATIONS=~/OrthoSkim-master/data/chloroplast_plants.gb**). Morevover, multiple taxonomic levels can be given in -l with a coma separator (*e.g.* -l asteraceae,helianthae).

Once all annotations and respective seeds are collected, we compute the database for the three targets using the `-m database` mode.

```
./orthoskim -m database -t chloroplast -c config_orthoskim.txt
./orthoskim -m database -t mitochondrion -c config_orthoskim.txt
./orthoskim -m database -t nucrdna -c config_orthoskim.txt
```

> **NOTE:** We supplied with ORTHOSKIM a python function SortDB.py allowing to select a subset of lineages by family in gene or genome databases. It allows to reduce the computational time of capture steps by reducing the number of sequences and keeping a taxonomic diversity within the database. This function can be run directly on outputs as following:

```
SortDB.py -i chloroplast_CDS.fa -f fasta -l 3 -o selected_chloroplast_CDS.fa -m gene
SortDB.py -i chloroplast_ncbi.gb -f genbank -l 5 -o selected_chloroplast_CDS.embl -m genome
```

with -i input genes/genomes file; -l number of queried lineages by family; -f input file format (embl/ genbank/fasta); -o output name (format fasta for genes or embl for genomes); -m mode (gene/genome)

### 4.2.2. assemblies and filtering

We next perform global assemblies and format the outputs. After that, assemblies were cleaned by removing all potential contaminants.

```
./orthoskim -m assembly -t spades -c config_orthoskim.txt
./orthoskim -m format -t spades -c config_orthoskim.txt
./orthoskim -m cleaning -c config_orthoskim.txt
```

> **Note:** For the cleaning step, we set the expected phyllum at "Embryophyta" (l.12 of the config file).

If you want to get summary statistics of assemblies, users can run the following command:

```
./orthoskim -m statistic_assembly -c config_orthoskim.txt
```

### 4.2.3. gene capture

The next step consists on capture all targeted genes into these assemblies. To do this, we run the `-m capture` mode with different targets.

```
./orthoskim -m capture -t chloroplast_CDS -c config_orthoskim.txt
./orthoskim -m capture -t chloroplast_rRNA -c config_orthoskim.txt
./orthoskim -m capture -t chloroplast_tRNA -c config_orthoskim.txt
./orthoskim -m capture -t mitochondrion_CDS -c config_orthoskim.txt
./orthoskim -m capture -t mitochondrion_rRNA -c config_orthoskim.txt
./orthoskim -m capture -t nucrdna -c config_orthoskim.txt
```

> **Note**: in this example for the chloroplast tRNA, we change the CHLORO_TYPE (l.49) from "exon" to "intron", to capture the intron of the trnL-UAA.

Summary statistics about the capture can be obtained by using the following mode:

```
./orthoskim -m statistic_capture -t chloroplast_CDS -t chloroplast_rRNA -t chloroplast_tRNA -t mitochondrion_CDS -t mit
```

> **NOTE:** Here, multiple targets (-t) are given in the command same line.

### 4.2.4. alignments

Finally, we compute a supermatrix by aligning captured genes (here on chloroplast CDS and rRNA) useful for phylogenetic inferences, by using the `-m alignment` mode.

```
./orthoskim -m alignment -t chloroplast_CDS -t chloroplast_rRNA -c config_orthoskim
```

> **NOTE:** all outputs are detailed in the previous sections.

# 5. Additional modes for PhyloDB users

Additional modes were implemented for PhyloDB users (*i.e.* for PHA, PHN, PHC member project) to use ORTHOSKIM along with annotations performed under these projects with Org.Asm assembler. Users can easily use all modes supplied in ORTHOSKIM in complement.

## 5.1. Sample file

Sample file can be created directly from samples location into the GriCAD infrastructures on the */bettik/LECA/phyloskims/release/* folder. This tab is produced by the `-m phyloskim_indexing` mode. This allowed to screen each sample that will be used for the gene extraction from `-p path/to/seek/files/`. Unwanted samples must be removed from the list before processing other modes.

```
./orthoskim -m indexing -c config_orthoskim.txt -p /bettik/LECA/phyloskims/release/
```

## 5.2. List of genes files

The extraction of orthologous regions and the creation of databases from annotations are based on a given list of genes. This list is supplied in **$CHLORO_GENES**, **$MITO_GENES** and **$NRDNA_GENES** (lines [63-65] of the config file) and must contain:

- the type of gene (*e.g.* CDS,rRNA,tRNA,misc_RNA)
- the gene name

```
head ~/OrthoSkim/ressources/listGenes.chloro

tRNA    trnV
tRNA    trnA
tRNA    trnN
rRNA    rrn16S
rRNA    rrn23S
rRNA    rrn4.5S
rRNA    rrn5S
CDS     psbA
CDS     matK
CDS     rps16
CDS     psbK
```

By default, ORTHOSKIM provided a list for tRNA, rRNA and CDS genes in chloroplast (see **$CHLORO_GENES** and **$MITO_GENES**). For the ribosomal complex, the gene type correspond to rRNA (*i.e.* for rrn18S, 5.8S rRNA, rrn28S) and misc_RNA (*i.e.* ITS1 and ITS2) (see **$NRDNA_GENES**) as annotated in Org.Asm assembler.

## 5.3. phyloDB database of references

ORTHOSKIM provides a mode to create a database from the all annotations performed within the project by using the `-m phyloskim_database` mode. To do this, all genes found in these annotations files are extracted with the header restrictions. Output files are created according to the name and the path set in the config file (**$CHLORO_REF_CDS**, **$CHLORO_REF_rRNA**, **$CHLORO_REF_tRNA** and **$NRDNA_REF** at lines 42-44 and 46 of the config file).

> **Note:** For chloroplast annotations, only genes found in single and circular contig will be extracted.

### 5.4. phyloDB extraction from annotations

For each sample of the sample file, ORTHOSKIM will perform genes extraction directly from annotation with `-m phyloskim_extraction_targeted` mode, according to a list of genes for `-t [chloroplast, nucrdna]` targets.

Results are output in **RES/** directory by creating subdirectories for each compartment and gene type, including a multifasta file per gene. For example, for chloroplast CDS provided in **$CHLORO_GENES**, ORTHOSKIM will output **RES/chloroplast_CDS/** subdirectory with CDS gene files.

```
ls -l ~/RES/chloroplast_CDS/

-rw-r--r--  1 pouchonc  staff   4874 16 avr 10:40 accD.fa
-rw-r--r--  1 pouchonc  staff   4952 16 avr 10:40 atpA.fa
-rw-r--r--  1 pouchonc  staff   4853 16 avr 10:40 atpB.fa
-rw-r--r--  1 pouchonc  staff   1580 16 avr 10:40 atpE.fa
-rw-r--r--  1 pouchonc  staff   2057 16 avr 10:40 atpF.fa
```

## 6. Funding

The PhyloAlps data collection was largely funded from the European Research Council under the European Community's Seventh Framework Programme FP7/2007-2013 grant agreement 281422 (TEEMBIO), the Sixth European Framework Programme (GOCE-CT-2007-036866), the Swiss SNF (Grant 31003A_149508/1), the ANR DIVERSITALP Project (ANR-07-BDIV-014), ANR project Origin-Alps (ANR-16-CE93-0004), France Génomique (ANR-10-INBS-09-08) and the NextBarcode project (Institut Français de Bioinformatique).

## 7. Support

For questions and comments, please contact: contact@orthoskim.org