METHOD ARTICLE

REVISED How to build phylogenetic species trees with OMA [version 2; peer review: 2 approved with reservations]

David Dylus[1-3*], Yannis Nevers[1-3*], Adrian M. Altenhoff[1,4], Antoine Gürtler[2,3], Christophe Dessimoz[1-3,5,6], Natasha M. Glover[1-3]

[1]Swiss Institute of Bioinformatics, Lausanne, 1015, Switzerland
[2]Department of Computational Biology, University of Lausanne, Lausanne, 1015, Switzerland
[3]Center for Integrative Genomics, University of Lausanne, Lausanne, 1015, Switzerland
[4]Department of Computer Science, ETH Zurich, Zurich, 8092, Switzerland
[5]Department of Genetics, Evolution and Environment, University College London, London, WC1E 6BT, UK
[6]Department of Computer Science, University College London, London, WC1E 6BT, UK

[*] Equal contributors

v2 **First published:** 04 Jun 2020, **9**:511
https://doi.org/10.12688/f1000research.23790.1

Latest published: 28 Feb 2022, **9**:511
https://doi.org/10.12688/f1000research.23790.2

Abstract

Knowledge of species phylogeny is critical to many fields of biology. In an era of genome data availability, the most common way to make a phylogenetic species tree is by using multiple protein-coding genes, conserved in multiple species. This methodology is composed of several steps: orthology inference, multiple sequence alignment and inference of the phylogeny with dedicated tools. This can be a difficult task, and orthology inference, in particular, is usually computationally intensive and error prone if done *ad hoc*. This tutorial provides protocols to make use of OMA Orthologous Groups, a set of genes all orthologous to each other, to infer a phylogenetic species tree. It is designed to be user-friendly and computationally inexpensive, by providing two options: (1) Using only precomputed groups with species available on the OMA Browser, or (2) Computing orthologs using OMA Standalone for additional species, with the option of using precomputed orthology relations for those present in OMA. A protocol for downstream analyses is provided as well, including creating a supermatrix, tree inference, and visualization. All protocols use publicly available software, and we provide scripts and code snippets to facilitate data handling. The protocols are accompanied with practical examples.

Keywords
phylogenetics, phylogenomics, species tree, OMA, Orthologous Matrix

Open Peer Review

Approval Status ? ?

	1	2
version 2 (revision) 28 Feb 2022		
version 1 04 Jun 2020	? view	? view

1. **Jianbo Xie**, National Engineering Laboratory for Tree Breeding, College of Biological Sciences and Technology, Beijing Forestry University, Beijing, China

2. **Denis Baurain**, University of Liège, Liège, Belgium

Any reports and responses or comments on the article can be found at the end of the article.

This article is included in the Bioinformatics gateway.

This article is included in the The OMA collection collection.

Corresponding author: Natasha M. Glover (natasha.glover@sib.swiss)

Author roles: Dylus D: Conceptualization, Formal Analysis, Investigation, Methodology, Project Administration, Software, Visualization, Writing – Original Draft Preparation; **Nevers Y**: Data Curation, Formal Analysis, Investigation, Methodology, Validation, Writing – Original Draft Preparation, Writing – Review & Editing; **Altenhoff AM**: Conceptualization, Data Curation, Methodology, Resources, Software, Writing – Review & Editing; **Gürtler A**: Formal Analysis, Investigation, Methodology; **Dessimoz C**: Funding Acquisition, Project Administration, Supervision, Writing – Review & Editing; **Glover NM**: Conceptualization, Project Administration, Supervision, Validation, Writing – Original Draft Preparation, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: This work was funded by a Service and Infrastructure grant from the Swiss Institute of Bioinformatics and a Swiss National Science Foundation Professorship grant (Grant 183723).
The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

How to cite this article: Dylus D, Nevers Y, Altenhoff AM *et al.* **How to build phylogenetic species trees with OMA [version 2; peer review: 2 approved with reservations]** F1000Research 2022, **9**:511 https://doi.org/10.12688/f1000research.23790.2

First published: 04 Jun 2020, **9**:511 https://doi.org/10.12688/f1000research.23790.1

REVISED **Amendments from Version 1**

There are no major differences between this version and the previous version. We mainly clarified the points in the text (as suggested by the reviewers), and updated Figure 3 and Figure 4 to reflect bootstrap values. We added a list of the species used in this analysis in the figshare archive, and updated the python scripts to make it compatible with the latest version of its dependency.

Any further responses from the reviewers can be found at the end of the article

Introduction

Inferring accurate and complete species phylogenies is a fundamental problem in biology[1]. Traditionally, species trees have been inferred using ubiquitous marker genes such as the small subunit ribosomal RNA (SSU rRNA 16S/18S) gene[2]. However, as there are fewer sites to sample, using only one gene per species limits the resolution of the inference; the phylogeny of the gene may not necessarily reflect the evolutionary history of the entire species (due to, for example, incomplete lineage sorting, "hidden" duplications followed by loss of one copy, horizontal gene transfer, etc.[3]). Additionally, phylogenies based on one gene are not always sufficient to obtain statistical support for difficult nodes due to their limited number of characters. For this reason, species phylogenies are now overwhelmingly inferred from multiple genes[4]. As long as one takes the necessary precautions, notably selecting true orthologs for their comparisons (see 4,5 for common pitfalls in phylogenomics), multilocus phylogenies are better resolved and more robust[6]. Recently, multiple protein-coding genes were used to infer a tree comprising ~3000 species, but this was still limited to a small number of concatenated ribosomal genes[7]. With the rise of next-generation sequencing, many hundreds of genes can now be considered when building species trees, which tremendously increases the available information for the inference.

To make use of all available genes, one needs to identify groups of genes that emerged from a common ancestral gene solely through speciation. These sets of genes, in which all pairs of genes are orthologs[8], are commonly referred to as **Orthologous Groups (OGs)**. Another term used for these types of groups are marker genes, or **phylogenetic marker genes**. The inference of OGs is non-trivial due to additional evolutionary events such as gene duplications, gene losses or horizontal gene transfers[9]. Furthermore, there are numerous algorithms for inferring orthology which can result in different OG composition, further complicating matters.

In this tutorial, we focus on OGs obtained from Orthologous MAtrix (OMA). Alternatively called "**OMA Groups**," they are stringently computed orthologs, they make use of all the available species in OMA, and are specifically designed for species tree inference. OMA Groups are defined as gene families that contain genes which are all orthologous to each other, with a maximum of one gene per species[10]. If recently diverged in-paralogs are inferred (i.e., co-orthologs), only one of the copies will be selected for the OG. Thus, all members of the OG are still orthologous to each other. This type of Orthologous Group is provided by only a few orthology databases such as BUSCO[11] and OMA[12,13] and, to our knowledge, only OMA allows for use of both precomputed and user-computed OGs. Moreover, OMA Groups have repeatedly been shown to produce reliable trees[12] and its underlying algorithm was shown to accurately infer OGs in a large-scale benchmark[14]. It is sometimes possible to rely on existing marker genes used in large-scale studies (for example[15], but they are generally available only for a subset of species and do not include newly sequenced species.

The OMA algorithm is freely available as an open source software tool (OMA Standalone[12]) that integrates well with the public OMA Browser (https://omabrowser.org), which is a database that provides orthology information among more than 2400 genomes across the tree of life, selected to maximize taxon coverage and users' needs[16]. To date (December 2021 release) there are 1719 Bacteria, 155 Archaea, and 622 Eukaryotes. In this protocol, we show how to leverage the publicly available orthology data to infer phylogenetic species trees.

First, we set up the prerequisites and explain how to search for precomputed OGs for species of interest in the OMA database. Then, we show how to infer a species tree under two scenarios: (1) using only species that are present in the public OMA database, or (2) using species in OMA in addition to other proteomes not available in the database, e.g. a proteome obtained from sequencing a new species. By proteome we mean the protein sequences of all protein-coding genes annotated in a genome. Finally, we show how to do downstream processing and tree inference. Each of these steps is illustrated by an example on real data.

Materials

The tools needed for this tutorial can be found in Table 1. All commands can be run from the command line and/or with python scripts. We reference four tree inference software tools, but there are many other alternatives (see List of phylogenetics software).

Two examples will be used in this tutorial, one to illustrate Protocol 1, and another to illustrate Protocol 2. Both examples can be downloaded from FigShare[17]. Instructions on how to obtain the data from the OMA Browser for Protocol 1 are described in the next section, so it is not required to download anything to complete this protocol. However, Protocol 2 demonstrates how to add external proteomes. For this example we chose two proteomes available in the OMA Browser, but we set them aside after downloading the data. We then re-add them as external data (FOMPI.fa and YEAST.fa). For reproducibility, these proteomes can be found in the /data/AddedGenomes subdirectory of Protocol 2. The rest of the data included in the example tarball are OGs and alignment files needed to compute the trees, which are also included as results.

The tree computations on these data have been performed using both RAxML 8.2.12 and IQ-TREE 1.7.beta17, as specified in the PDF accompanying the examples.

Protocols

Phylogenetic tree inference using OMA is done in three steps: getting OGs data, aligning all sequences of every OG and combining them into a supermatrix, and finally, using tree inference tools on the supermatrix. Depending

Table 1. Computational tools needed for making a phylogenetic species tree using OMA. Note that four phylogenetic tree inference software packages are given, but only one (user preference) is needed to complete this tutorial.

Tool	Use case	How to get it
Command line	Mandatory to run the commands written in this tutorial	Installed by default on Unix and Mac
Python 3	Python language interpreter. Mandatory for running the scripts used in this paper	https://www.python.org/downloads/
OMA Browser	Needed to import orthology relations used for tree inference	https://omabrowser.org
OMA standalone	Needed to infer orthology for data not available in the OMA Browser	https://omabrowser.org/standalone/#downloads
MAFFT	Multiple sequence alignment software	https://mafft.cbrc.jp/alignment/software/
High Performance Computing (HPC)	Needed if a high amount of computation is involved	Institutional infrastructure
IQ-Tree	Phylogenetic tree software	http://www.iqtree.org/#download
RAxML	Phylogenetic tree software	https://cme.h-its.org/exelixis/web/software/raxml/index.html
Phylobayes	Phylogenetic tree software	http://www.atgc-montpellier.fr/phylobayes/
PhyML	Phylogenetic tree software	http://www.atgc-montpellier.fr/phyml/
Phylo.io	Phylogenetic tree visualization website	http://phylo.io

on the species requirement, two options are available to obtain OG data, they are detailed in Protocol 1 and 2 subsections. Protocol 1 is the fastest and can be used if all species of interest are available on the OMA Browser. Alternatively, Protocol 2 is for the cases when new proteomes must be added, or when solely using data computed by OMA Standalone. Later steps are the same for both cases and are addressed in Protocol 3.

Protocol 1: Export marker genes to make a phylogeny of species found in OMA

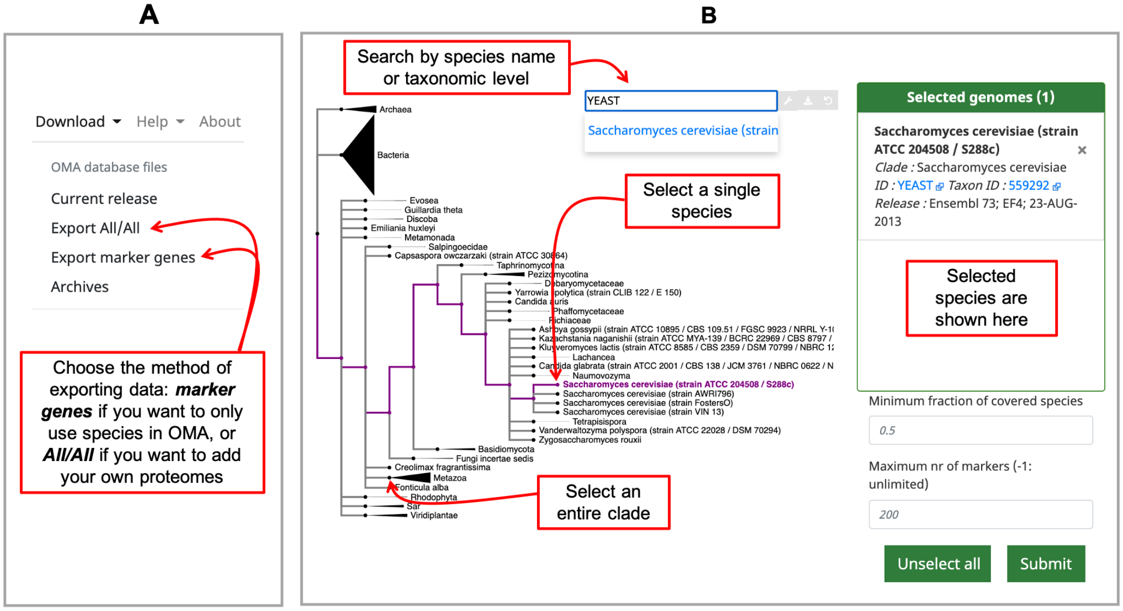
This method is the quickest way to obtain data to build a phylogenetic species tree, but is only useful if one is interested in making a tree from species already in the OMA database. To do this, the *Export marker genes* function in the browser takes advantage of the precomputed OMA Groups. As mentioned in the Introduction, OMA Groups are a specific type of OG which contain sets of genes that are all orthologous to one another. This implies that there is at most one gene from each species in a group.

Finding species of interest in OMA. The OMA public database and all related information are accessed through the OMA browser (https://omabrowser.org/). One can search for species of interest in the OMA database by browsing through the available data in OMA using the release info page (from the menu in the upper right corner: *Explore -> Release information*). Two browsing options are available, the default one is through an interactive tree, with colors indicating domains of life: bacteria are blue, archaea are green and eukaryotes are red. The other option is a table viewer featuring a search bar and both can be accessed through the *Select species browser widget* icons in the top right of the *Species Information* visualization.

Export the relevant data from OMA. The way to obtain OGs with only species present in the OMA database is by using the *Download -> Export marker genes* option (Figure 1A) in the top right menu. This will open a page which allows the user to select species. Species can be searched by name or clade. A whole clade can be selected by clicking on the node (*select all species*). A single species can be selected by clicking on the leaf (*select species*). All selected species will be displayed in the right box with additional species information (release info, taxon id, etc.) (Figure 1B).

Specifying the Minimum Species Coverage and Maximum Number of Markers parameters. After species selection, exported OGs will depend on the **minimum fraction of covered species** and the **maximum number of markers** parameters:

- **Minimum species coverage: the lowest acceptable proportion of selected species that are present in any given OG in order to be exported.**


A
B
Download
Help
About
OMA database files
Current release
Export All/All
Export marker genes
Archives
YEAST
Saccharomyces cerevisiae (strain
Archaea
Bacteria
Evosea
Guillardia theta
Discoba
Emiliania huxleyi
Metamonada
Salpingoecidae
Capsaspora owczarzaki (strain ATCC 30864)
Taphrinomycotina
Pezizomycotina
Debaryomycetaceae
Yarrowia lipolytica (strain CLIB 122 / E 150)
Candida auris
Phaffomycetaceae
Pichiaceae
Ashbya gossypii (strain ATCC 10895 / CBS 109.51 / FGSC 9923 / NRRL Y-1
Kazachstania naganishii (strain ATCC MYA-139 / BCRC 22969 / CBS 8797 /
Kluyveromyces lactis (strain ATCC 8585 / CBS 2359 / DSM 70799 / NBRC 1
Lachancea
Candida glabrata (strain ATCC 2001 / CBS 138 / JCM 3761 / NBRC 0622 / N
Naumovozyma
Saccharomyces cerevisiae (strain ATCC 204508 / S288c)
Saccharomyces cerevisiae (strain AWRI796)
Saccharomyces cerevisiae (strain FostersO)
Saccharomyces cerevisiae (strain VIN 13)
Tetrapisispora
Vanderwaltozyma polyspora (strain ATCC 22028 / DSM 70294)
Zygosaccharomyces rouxii
Basidiomycota
Fungi incertae sedis
Creolimax fragrantissima
Metazoa
Fonticula alba
Rhodophyta
Sar
Viridiplantae
Selected genomes (1)
Unselect all
Submit

Choose the method of exporting data: ***marker genes*** if you want to only use species in OMA, or ***All/All*** if you want to add your own proteomes

Search by species name or taxonomic level

Select a single species

Select an entire clade

Selected species are shown here

Saccharomyces cerevisiae (strain ATCC 204508 / S288c) ×
Clade : Saccharomyces cerevisiae
ID : YEAST *Taxon ID :* 559292
Release : Ensembl 73; EF4; 23-AUG-2013

Minimum fraction of covered species
0.5

Maximum nr of markers (-1: unlimited)
200

Figure 1. Exporting data from OMA for building a species tree. A) Choose which type of data to export from the *Download* tab on the right hand side of the home page. **B**) Select your proteomes from those in the OMA database by using the interactive species tree, which is based on the NCBI taxonomy.

A more permissive (lower) minimum species coverage will result in a higher number of exported groups. Choosing this parameter depends on the number of and how closely related are the selected species. For instance, consider the *Drosophila* clade versus chordates clade (20 and 116 species in the January 2020 release, respectively). If one selects the 20 *Drosophila* genomes and sets the minimum species coverage to 0.5, only OGs with at least 10 *Drosophila* species will be exported. In the January 2020 release, this results in 11,855 OGs which meet this criteria. If using the same 0.5 minimum species coverage for the chordates, it results in 14,357 OGs exported. On the other hand, for a 0.8 minimum species coverage, 7,886 and 6,329 OGs are exported for *Drosophila* and chordates clades, respectively.

- **Maximum number of markers: the maximum number of OGs/marker genes to return.** To consider as much information as possible in the tree inference, remove any limit by setting this parameter to -1, in which case all OGs fulfilling the minimum species coverage parameter will be returned. To speed up the tree inference, set this value to below 1000 genes. When the number of markers is limited in this way, OGs with the highest coverage will be prioritized.

After filling in the parameters and submitting the request, the browser will return a compressed archive (“tarball”) that contains a fasta file with unaligned sequences for each OG. Depending on the size of the request, it may take a few minutes for this operation to complete.

As an example for Protocol 1, we performed an analysis on 20 yeast species, using only OGs shared by all species (*Minimum species coverage* : 1) and no limit to the number of OGs retrieved (*Maximum number of markers* : -1). We obtained 169 OGs with this query. The corresponding data, including a list of the 20 species used, can be found at FigShare[17], in the Protocol_1 folder.

Upon exporting the marker genes, i.e. OGs, from OMA, the data can be used to make a phylogenetic species tree (skip to Protocol 3: Downstream processing and tree inference).

Protocol 2: Export precomputed OMA all-against-all data as a backbone to add your own genomes and use with OMA standalone

Orthology computation first starts with an all-against-all alignment phase—comparing all proteins in every species of interest to each other. If genomes to be included in the species phylogeny are not present in OMA (hereafter referred to as “added genomes”), it is necessary to first compute orthology predictions for the combined set of species (those in OMA plus the added genomes). This approach is computationally more expensive and requires that computations are performed on a local machine or high performance computing cluster (HPC). However, by using the OMA Browser’s *Download -> Export All/All* option, one can take advantage of the precomputed all-against-all data for those species in OMA, saving time. The following protocol describes how to make use of this data and run OMA Standalone, the software for running the OMA algorithm on added genomes. In the case where the user wants to only use genomes unavailable in OMA, skip to the “Running OMA Standalone” section.

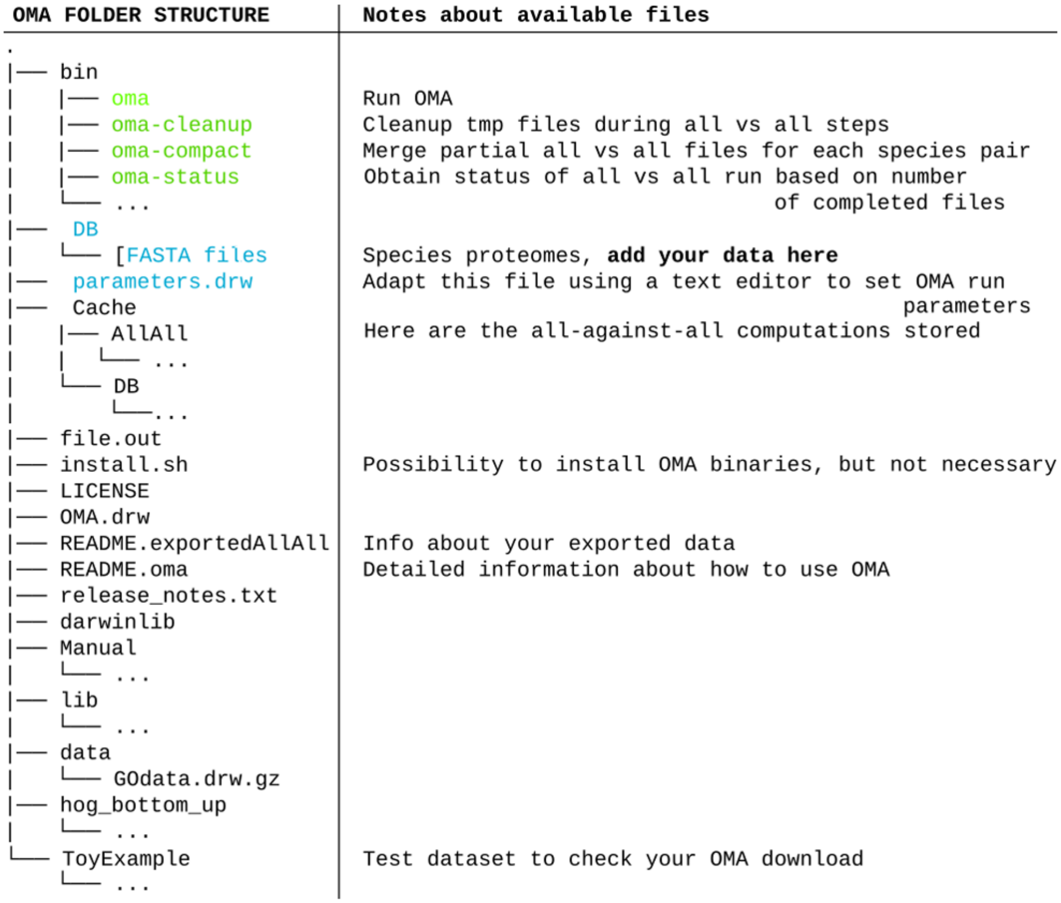
Export the all-against-all from OMA. Choose species which you want to combine with your own genomes by choosing *Download -> Export All/All* from the top right menu (Figure 1A). This will lead to an interactive species tree of all the species in OMA, for which you can choose your species of interest to export (Figure 1B).

After selecting species and clicking submit, the OMA Browser will export a tarball (described in Figure 2) which contains:

- The all-against-all alignments of the selected species, found in the folder “Cache.”
- All exported genomes, in the format of protein fasta files, found in the folder “DB.”
- The full OMA standalone software tool. No need to download it separately.

Combining the added genomes with exported OMA data. Next, the added genomes data must be combined with the OMA data. For this procedure, the added genomes data must fulfill certain conditions:

- Each additional dataset is in the form of a fasta file, containing *protein* sequences of all coding genes in the corresponding genome. Please note that OMA Standalone can work on nucleic coding sequences when starting from scratch, however for compatibility with pre-computed OMA data, only protein sequences may be used when combining new and exported data.



```
OMA FOLDER STRUCTURE          | Notes about available files
.                             |
|── bin                       |
|    |── oma                  | Run OMA
|    |── oma-cleanup          | Cleanup tmp files during all vs all steps
|    |── oma-compact          | Merge partial all vs all files for each species pair
|    |── oma-status           | Obtain status of all vs all run based on number
|    └── ...                  |                                 of completed files
|──  DB                       |
|    └── [FASTA files         | Species proteomes, add your data here
|──  parameters.drw           | Adapt this file using a text editor to set OMA run
|──  Cache                    |                                         parameters
|    |── AllAll               | Here are the all-against-all computations stored
|    |   └── ...              |
|    └── DB                   |
|        └──...               |
|── file.out                  |
|── install.sh                | Possibility to install OMA binaries, but not necessary
|── LICENSE                   |
|── OMA.drw                   |
|── README.exportedAllAll     | Info about your exported data
|── README.oma                | Detailed information about how to use OMA
|── release_notes.txt         |
|── darwinlib                 |
|── Manual                    |
|    └── ...                  |
|── lib                       |
|    └── ...                  |
|── data                      |
|    └── GOdata.drw.gz        |
|── hog_bottom_up             |
|    └── ...                  |
└── ToyExample                | Test dataset to check your OMA download
     └── ...                  |
```

Figure 2. Tree organization of the tarball downloaded through the OMA Browser after exporting an all-against-all of selected species. The important files and folders are colored. In green, the executable files mentioned in the course of the tutorial. In blue are the files and folder that will need to be modified. Other files and folders (in black) will not be used in the course of the tutorial. Files and folders not shown are represented by three dots.

- The name of the fasta file should identify the species clearly and uniquely. The exported genomes from OMA use for example UniProt's mnemonic five-letter species codes. The filename must end with a ".fa" suffix and must not contain any whitespace characters. The filename without the ".fa" suffix is used as the species name throughout the process and result files.

- Each sequence in the fasta file has a clear and unique identifier. We suggest not to use special characters such as brackets, dots, or a pipe character. The reason is that many programs use them for special purposes, e.g. brackets are used in the newick format for tree representation, and the pipe character is often used to separate ids and annotations.

If these conditions are fulfilled, these fasta files must be put into the DB folder with the other exported OMA genomes (Figure 2), where they will be considered as a unique dataset for the following steps.

Setting the parameters for OMA standalone. Before starting the computation, it is wise to adjust the parameters file, called "parameters.drw" (Figure 2), which can be edited with any text editor. If the goal is to only generate a dataset for species phylogeny inference (and not to keep other unrelated orthology inferences, such as Hierarchical Orthologous Groups[10], which better represent individual genes' evolutionary histories but take time to compute), one can avoid doing computations and generating output files that are not needed by the following:

- Uncomment (remove the # from) all the lines starting with `WriteOutput` EXCEPT `#WriteOutput_OrthologousGroupsFasta := false`. By keeping that one commented, OMA standalone will produce one fasta file for each inferred OG.

- Deactivate the Hierarchical Orthologous Group inference, which is not needed here, by setting `DoHierarchicalGroups := false;`
- Likewise, deactivate the gene function prediction by setting `DoGroupFunctionPrediction := false;`
- *Tip:* do not omit a semicolon at the end of each uncommented statement.

Running OMA standalone. To run OMA standalone, one needs to be aware that the OMA pipeline can be split into two parts: all-against-all alignments for homology inference and orthology calling. Because OMA can compute Smith-Waterman alignments in parallel for all species which were not exported from OMA (see Export the all-against-all from OMA), it is beneficial to perform the computations on a computer cluster. However, if the dataset is small (e.g. 2–3 additional genomes), the computations can be run locally on a standard computer.

To run OMA standalone on a small dataset locally:

1. Within the extracted tarball folder you can start the computation with the command line:

   ```
   $: bin/oma -n NR_PROCESSES
   ```

   `NR_PROCESSES` should not be higher than the number of CPUs you have available on your machine.

For a larger dataset, we recommend the use of an HPC cluster. We recommend to break up the computations into two parts: first the all-against-all part, then the orthology inference part:

1. Create a submission script for your cluster. Examples of submission scripts are provided at https://omabrowser.org/standalone/#schedulers and 18.
2. Make sure that the submission script enters the folder into which the tarball was extracted, by either running the script from inside that directory or using the `cd` command appropriately.
3. The line to start the OMA all-against-all computation in the submission scripts is:

   ```
   $: bin/oma -s
   ```

   The -s option means stop after the all-against-all phase. Since this part can be parallelized, we recommend using job-arrays. For this you need to set the number of parallel jobs as an environment variable (`export NR_PROCESSES=100`) and use the job-array syntax in the submission script (e.g. in LSF: `bsub -J oma[1-$NR_PROCESSES] bin/oma -s`). OMA Standalone automatically partitions the work chunks in a static and deterministic way among the specified number of workers. Progress of the entire computation can be checked with the OMA Status command (see below). For environments with limited runtimes/walltimes see https://omabrowser.org/standalone/#advanced%20optimisations.
4. Check whether the all-against-all computation is finished using:

   ```
   $: bin/oma-status -i
   ```

   This command will output a file formatted as:

   ```
   Summary of OMA standalone All-vs-All computations:
   --------------------------------------------------
   Nr chunks started: A (D%)
   Nr chunks finished: B (E%)
   Nr chunks finished w/o exported genomes: C (F%)
   ```

   Where the letters A, B, C, D, E and F represent numbers. Once the computations are completed, D should be equal to 0.0%, and both E and F to 100.0%
5. In the case where the jobs are finished but the all-against-all computation is still not complete, use the `oma-cleanup` and `oma-compact` commands before re-submitting.

   ```
   $:bin/oma-cleanup
   $: bin/oma-compact
   ```

These commands remove partially finished output files in the Cache/AllAll folder and zip all partial computations that are finished to one file, respectively.

6. Once the all-against-all computation has finished, the final step is the orthology calling. This step is more memory intensive, requires a single process, and can be called with:

```
$: bin/oma
```

Once the computation finishes, all results will be stored in the newly-created “Output” folder. In this folder there will be an “EstimatedSpeciesTree.nwk” file that contains a phylogenetic tree that can be visualized using a tree visualization tool such as Phylo.io[19]. This is a distance tree based on the weighted average of the pairwise distances between sequences within the most complete OMA groups. **This species tree is a rough estimate that is computed on the fly, and is not the final tree.** It can be used as control to identify problems in the dataset but will not be as reliable as the tree inferred using the generated OGs later in this protocol. **Therefore, it is recommended to use the OGs to compute your own tree with external software.** The OGs (OMA Groups)[20] can be found in the “OrthologousGroupsFasta” folder, with each OG containing at least two species.

Usually for the construction of phylogenetic trees, one would select only OGs that contain at least X% of species, as described above with the parameter *Minimum Species Coverage*. The python script filter_groups.py from the git repository associated to this publication (https://doi.org/10.5281/zenodo.6037516[21]) can be used to filter the OMA groups that contain at least X `MIN_NR_SPECIES` (replace `<MIN_NR_SPECIES>` and `<destination/directory>` with your own values):

```
$: python filter_groups.py --min-nr-species <MIN_NR_SPECIES> --input Output/
OrthologousGroupFasta/ --output <destination/directory>
```

For example, we performed an analysis adding two yeast proteomes hypothetically not available in OMA and 18 available yeast proteomes. As a first step, we downloaded the precomputed data for the 18 proteomes from the OMA Browser and launched the computation after adding two separate proteomes. Once the computation finished, we selected 880 OGs that included at least 90% of the 20 species — 18 — as a dataset to construct a tree. The data used in this example is available at FigShare[17] in the Protocol_2 folder.

Protocol 3: Downstream processing and tree inference

Once all selected OGs are obtained from either of the first two protocols, the next step is to align all sequences within each OG. This can be done with any Multiple Sequence Alignment (MSA) tools, in this example we use MAFFT[22]. To run it, navigate to the folder containing the selected OGs and execute the following command, which runs mafft on each fasta file:

```
$: for i in $(ls -1 *.fa); do  mafft --maxiterate 1000 --localpair $i > $i.aln;
done
```

This command sequentially generates an MSA file (.aln) for each OG. Depending on the number of OGs and species in your dataset, executing it may take a prohibitive amount of time. If it is the case, we recommend using job-arrays to execute the alignments in parallel. In order to infer the phylogeny of the species from these alignments, they have to be concatenated in a single alignment commonly referred to as supermatrix. We provide a python script to automate this, concat_alignment.py, available on https://doi.org/10.5281/zenodo.6037516[21]. The `--format-output` option allows for choosing the output format of this concatenation, either fasta or phylip format (some phylogenetic software requires a specific format as input). Once the python script is downloaded or cloned, ensure that all alignments are in the same folder, and launch using the following command:

```
$: python concat_alignments.py <path>/<to>/<alignments>/*aln --format-output
[fasta/phylip] > output
```

After computing the supermatrix, the phylogenetic tree can be inferred using any number of available software. We recommend choosing from the tools in Table 2, sorted by computing time and increasing precision.

Tree visualization. Most of the current phylogenetic inference tools provide trees in Newick format as output. In order to visualize such a tree, one can use the web-based viewer phylo.io (http://phylo.io) or other tree visualization tools (e.g. FigTree, phylogeny.io, etc). Displaying bootstrap values for internal nodes is recommended to evaluate the confidence of the inferred tree topology.

In our examples, we inferred trees by aligning the sequences with MAFFT, concatenated the alignments using the aforementioned concat_alignments.py, and ran both IQ-TREE and RAxML (Figure 3 and Figure 4). The data used for and the results from the computations can be found on FigShare[17] (alignments in the "data/Alignments" folder, and trees in the "tree" folder). The exact code used for these examples is on[21].

Discussion and conclusion

With the wealth of genomic data available in an era of high-throughput sequencing, there is much to gain by making phylogenies from concatenations of multiple genes rather than from one single gene. This can better represent the evolutionary history of a clade, because the evolutionary history of a single gene can be

Table 2. Recommended software and example commands for computing a phylogenetic tree. Parameters, such as memory or threads, may vary based on size of dataset.

Software for making phylogenetic trees	Example command
IQ-Tree	`iqtree -s alignment.phy -m LG -T 20 --mem 20G -seed 12345 -bb 1000`
RaML	`raxml-ng --threads 20 --model LG+G8+F --seed 15826 --msa alignment.phy --all --bs-trees 100`
PhyloBayes	`pb_mpi -dc -gtr -cat -dgam 4 -x 10 1000 -d alignment.phy alignment.chain1`

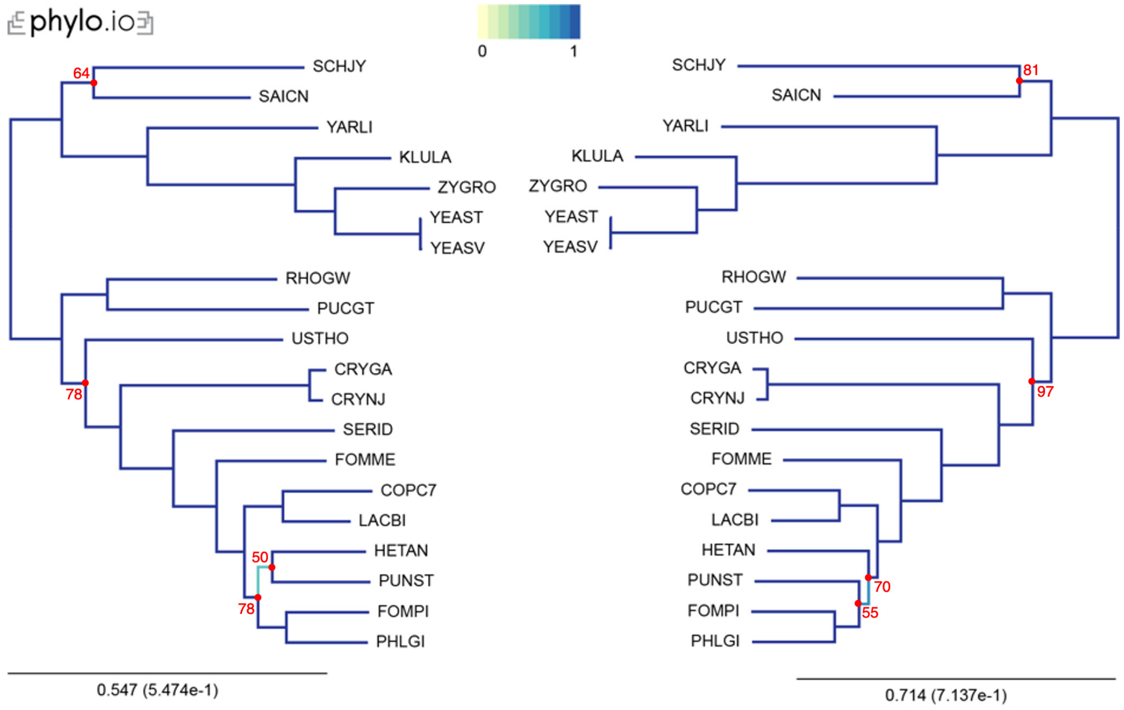


Figure 3. Comparison of phylogenetic trees computed by IQ-TREE, using an LG substitution model (left), and RAxML, using an LG substitution model, a discrete Gamma model of rate heterogeneity with 8 categories, and empirical amino-acid frequencies (right). Trees were computed with 20 yeast species present in OMA. The leaves of the trees are the UniProt 5-letter species codes. The following export options were used: Minimum species coverage: 1, Maximum nr of markers: -1 (uncapped). 168 marker genes were exported. Visualization was done with phylo.io; different shades of blue show variations in topology. Bootstrap values are reported in red for each bipartition with a bootstrap <100.

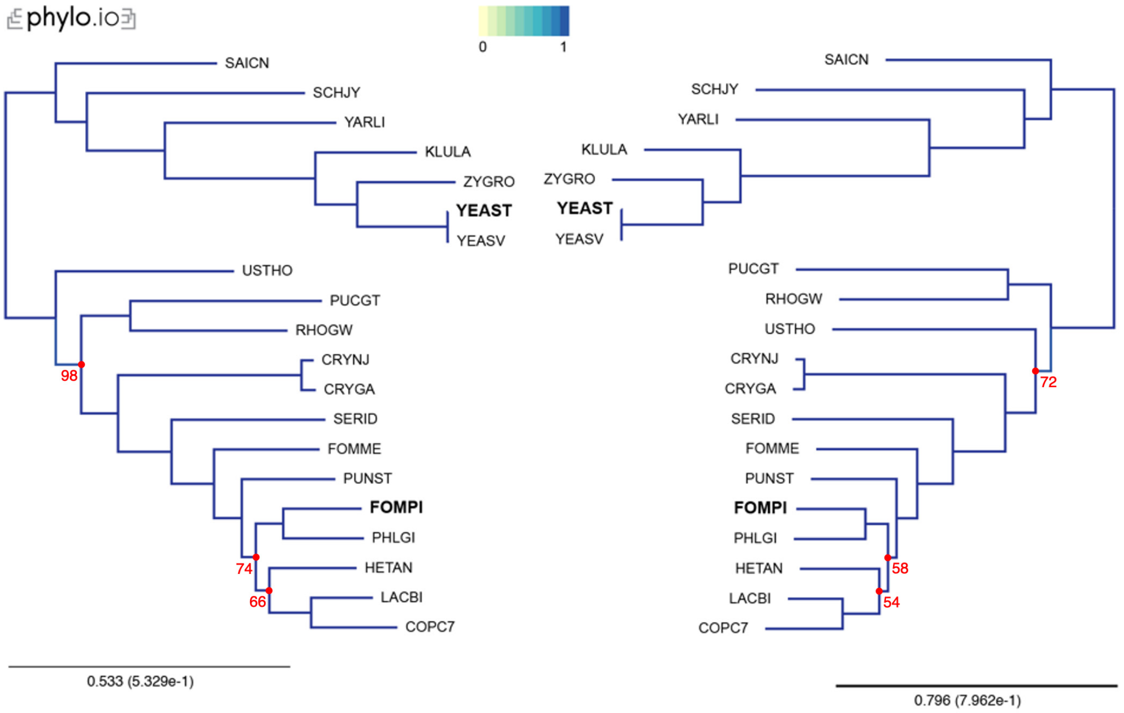


Figure 4. Comparison of phylogenetic trees, using additional species, computed by IQ-TREE, under a LG substitution model (left), and RAxML, under a LG substitution model, a discrete Gamma model of rate heterogeneity with 8 categories and empirical amino-acid frequencies (right). Trees were computed with 18 yeast species present in OMA, plus two additional proteomes (YEAST and FOMPI). The leaves of the trees are the UniProt 5-letter species codes. Genes used to compute the tree had to be shared by at least 90% of the species (minimum species coverage: 0.9, maximum number markers: -1). This represents 880 OGs. Visualization was done with phylo.io; different shades of blue show variations in topology (in this case both trees have identical topology). Bootstrap values are reported in red for each bipartition with a bootstrap <100.

misrepresentative of a species evolutionary history. The principle of a supermatrix approach is that by combining multiple genes in one single phylogeny, one can combine the phylogenetic signal of multiple genes. One has to be careful however, to not combine “phylogenetic noise”. Orthologs selection is particularly important in this regard[4,23], because errors in orthology inference could add genes that are not true orthologs, but rather paralogs descending from the ancestral genes by a duplication event. Thus, they would have a different evolutionary history than the sought species phylogeny.

OMA Groups (or Orthologous Groups) are a well-suited set of orthologs for this kind of analysis, as the criteria used to compute these orthologs are stringent. They require that all genes are reciprocally closest genes in their respective species to all the other genes of the group and do not allow more than one gene in a species, thus excluding paralogs. In the *Quest for Orthologs* Benchmark[14], the community benchmark for orthology inference, OMA Groups are consistently the most specific inference, although lacking in recall. As potentially missing genes are less detrimental to phylogenetic determination than false predictions are[24], this is an appropriate choice of orthology inference method for this tutorial. Several phylogenies have already been published using OMA standalone or data from the OMA Browser, including those for archaea, sharks, spiders, worms, and insects, among others[25–29].

This tutorial demonstrated how to carry out these different steps to infer a phylogenetic tree: orthology determination, sequence alignments, supermatrix construction, and phylogeny inference. It is designed to allow users to leverage the state of the art orthology inference provided by OMA Groups while reducing the necessary computation from their side, namely by relying on precomputed all-against-all alignments provided by the OMA Browser. We include code snippets and scripts that automate the whole process, and ensure reproducibility of all phylogenetic analyses following this protocol. The tutorial is accompanied by practical examples with all data available on GitHub and Figshare.

This tutorial is designed to help users generate a species tree phylogeny on reliable data, by relying on the least amount of computation. Nevertheless, we advise care in interpretation of the obtained species tree. In particular, even with accurate selection of orthologs, non-phylogenetic noise may persist in the data, making some branches hard to resolve. It is exemplified in this tutorial by a few differences between the species tree produced by the two different protocols, likely due to difference in the number of genes used. To avoid misinterpretation of the data, it is wise to compute and report measures of bipartition consistency, like bootstrap support values[30], while generating a species tree. A low bootstrap value will flag bipartitions that are subject to phylogenetic noise and that cannot be asserted with confidence. In our examples, bipartitions that differ between protocols have relatively low bootstrap values in the species tree.

For more information about the theory behind phylogenomics and the different methods, we refer the reader to recent reviews[31–33]. In the context of this tutorial, we used well-established MSA and phylogenetic tree inference tools. For the more difficult cases however, it advised to carefully choose which tool to use, including some tools which are not mentioned here. For more information about existing tools the readers are invited to turn to the relevant literature[32,34]. The protocols described here can be adapted to suit any other software compatible with standard data formats.

Data availability

The imported OG data and the OMA standalone software can be obtained from the OMA Browser (https://omabrowser.org), following instructions in this tutorial.

Figshare: Phylogenetic Tree Tutorial Example Data, https://doi.org/10.6084/m9.figshare.10780820.v6[17]

Data are available under the terms of the Creative Commons Zero “No rights reserved” data waiver (CC0 1.0 Public domain dedication).

Additional python scripts (filter_groups.py and concat_alignments.py) are publicly available: https://github.com/DessimozLab/f1000_PhylogeneticTree

Archived scripts as at time of publication: https://zenodo.org/record/6037516#.YgVAju6ZP0s[21]

License for scripts: MIT license

Software availability

OMA Browser available at: https://omabrowser.org/.

Source code for OMA Standalone available from: https://github.com/DessimozLab/OmaStandalone/tree/v2.4.0

Archived source code of OMA StandAlone at time of publication: https://doi.org/10.5281/zenodo.3555595[13].

OMA Browser license: Mozilla Public License version 2.

Acknowledgments

We would like to thank Natalia Zajac, Marion Brechet, Katharina Pfaller, and the reviewers Jianbo Xie and Denis Baurain for their useful feedback on the tutorial.

References

1. Hinchliff CE, Smith SA, Allman JF, *et al.*: **Synthesis of Phylogeny and Taxonomy into a Comprehensive Tree of Life.** *Proc Natl Acad Sci U S A.* 2015; **112**(41): 12764–9.
PubMed Abstract | Publisher Full Text | Free Full Text

2. Lane DJ, Pace B, Olsen GJ, *et al.*: **Rapid Determination of 16S ribosomal RNA Sequences for Phylogenetic Analyses.** *Proc Natl Acad Sci U S A.* 1985; **82**(20): 6955–9.
PubMed Abstract | Publisher Full Text | Free Full Text

3. Maddison WP: **Gene Trees in Species Trees.** *Syst Biol.* 1997; **46**(3): 523–36.
Publisher Full Text

4. Philippe H, Brinkmann H, Lavrov DV, *et al.*: **Resolving Difficult**

Phylogenetic Questions: Why More Sequences Are Not Enough. *PLoS Biol.* 2011; **9**(3): e1000602.
PubMed Abstract | Publisher Full Text | Free Full Text

5. Philippe H, de Vienne DM, Ranwez V, *et al.*: **Pitfalls in supermatrix phylogenomics.** *EJT.* 2017; (283).
Publisher Full Text

6. Gadagkar SR, Rosenberg MS, Kumar S: **Inferring Species Phylogenies From Multiple Genes: Concatenated Sequence Tree Versus Consensus Gene Tree.** *J Exp Zool B Mol Dev Evol.* 2005; **304**(1): 64–74.
PubMed Abstract | Publisher Full Text

7. Hug LA, Baker BJ, Anantharaman K, *et al.*: **A New View of the Tree of Life.** *Nat Microbiol.* 2016; **1**: 16048.
PubMed Abstract | Publisher Full Text

8. Fitch WM: **Distinguishing homologous from analogous proteins.** *Syst Zool.* 1970; **19**(2): 99–113.
PubMed Abstract | Publisher Full Text

9. Altenhoff AM, Glover NM, Dessimoz C: **Inferring Orthology and Paralogy.** *Methods Mol Biol.* In Anisimova M, editor. *Evolutionary Genomics: Statistical and Computational Methods.* New York, NY: Springer New York; 2019; **1910**: 149–75.
PubMed Abstract | Publisher Full Text

10. Zahn-Zabal M, Dessimoz C, Glover NM: **Identifying orthologs with OMA: A primer [version 1; peer review: 2 approved].** *F1000Res.* 2020; **9**(27): 27.
PubMed Abstract | Publisher Full Text | Free Full Text

11. Simão FA, Waterhouse RM, Ioannidis P, *et al.*: **BUSCO: Assessing Genome Assembly and Annotation Completeness With Single-Copy Orthologs.** *Bioinformatics.* 2015; **31**(19): 3210–2.
PubMed Abstract | Publisher Full Text

12. Altenhoff AM, Levy J, Zarowiecki M, *et al.*: **OMA Standalone: Orthology Inference Among Public and Custom Genomes and Transcriptomes.** *Genome Res.* 2019; **29**(7): 1152–63.
PubMed Abstract | Publisher Full Text | Free Full Text

13. Altenhoff AM, Glover NM, Train CM, *et al.*: **The OMA orthology database in 2018: retrieving evolutionary relationships among all domains of life through richer web and programmatic interfaces.** *Nucleic Acids Res.* 2018; **46**(D1): D477–85.
PubMed Abstract | Publisher Full Text | Free Full Text

14. Altenhoff AM, Boeckmann B, Capella-Gutierrez S, *et al.*: **Standardized benchmarking in the quest for orthologs.** *Nat Methods.* 2016; **13**(5): 425–30.
PubMed Abstract | Publisher Full Text | Free Full Text

15. Zhu Q, Mai U, Pfeiffer W, *et al.*: **Phylogenomics of 10,575 genomes reveals evolutionary proximity between domains Bacteria and Archaea.** *Nat Commun.* 2019; **10**(1): 5477.
PubMed Abstract | Publisher Full Text | Free Full Text

16. Altenhoff AM, Train CM, Gilbert KJ, *et al.*: **OMA orthology in 2021: website overhaul, conserved isoforms, ancestral gene order and more.** *Nucleic Acids Res.* 2021; **49**(D1): D373–9.
PubMed Abstract | Publisher Full Text | Free Full Text

17. Nevers Y: **Phylogenetic Tree Tutorial Example Data.** 2022.
https://figshare.com/articles/dataset/Example_Data/10780820/6

18. Glover N: **OMA standalone cheat sheet.** 2021; [cited 2022 Feb 8].
Publisher Full Text

19. Robinson O, Dylus D, Dessimoz C: ***Phylo.io*: Interactive Viewing and Comparison of Large Phylogenetic Trees on the Web.** *Mol Biol Evol.* 2016; **33**(8): 2163–6.
PubMed Abstract | Publisher Full Text | Free Full Text

20. Roth AC, Gonnet GH, Dessimoz C: **Algorithm of OMA for Large-Scale Orthology Inference.** *BMC Bioinformatics.* 2008; **9**: 518.
PubMed Abstract | Publisher Full Text | Free Full Text

21. Altenhoff A, Nevers Y, Glover N: **DessimozLab/f1000_PhylogeneticTree: v1.1.** 2022.
https://zenodo.org/record/6037516#.YgU9uO6ZP0s

22. Katoh K, Standley DM: **MAFFT multiple sequence alignment software version 7: improvements in performance and usability.** *Mol Biol Evol.* 2013; **30**(4): 772–80.
PubMed Abstract | Publisher Full Text | Free Full Text

23. Fernández R, Gabaldon T, Dessimoz C: **Orthology: Definitions, prediction, and impact on species phylogeny inference.** *Phylogenetics in the Genomic Era.* 2020; 2–4.
Reference Source

24. Baurain D, Philippe H: **Current Approaches to Phylogenomic Reconstruction.** In: Caetano-Anollés G, editor. *Evolutionary Genomics and Systems Biology.* Hoboken, NJ, USA: John Wiley & Sons, Inc.; 2010; 17–41.
Publisher Full Text

25. Williams TA, Szöllősi GJ, Spang A, *et al.*: **Integrative modeling of gene and genome evolution roots the archaeal tree of life.** *Proc Natl Acad Sci U S A.* 2017; **114**(23): E4602–11.
PubMed Abstract | Publisher Full Text | Free Full Text

26. Hara Y, Yamaguchi K, Onimaru K, *et al.*: **Shark genomes provide insights into elasmobranch evolution and the origin of vertebrates.** *Nat Ecol Evol.* 2018; **2**(11): 1761–71.
PubMed Abstract | Publisher Full Text

27. Wood HM, González VL, Lloyd M, *et al.*: **Next-generation museum genomics: Phylogenetic relationships among palpimanoid spiders using sequence capture techniques (Araneae: Palpimanoidea).** *Mol Phylogenet Evol.* 2018; **127**: 907–18.
PubMed Abstract | Publisher Full Text

28. Philippe H, Poustka AJ, Chiodin M, *et al.*: **Mitigating Anticipated Effects of Systematic Errors Supports Sister-Group Relationship between Xenacoelomorpha and Ambulacraria.** *Curr Biol.* 2019; **29**(11): 1818–26.e6.
PubMed Abstract | Publisher Full Text

29. Dikow RB, Frandsen PB, Turcatel M, *et al.*: **Genomic and transcriptomic resources for assassin flies including the complete genome sequence of *Proctacanthus coquilletti* (Insecta: Diptera: Asilidae) and 16 representative transcriptomes.** *PeerJ.* 2017; **5**: e2951.
PubMed Abstract | Publisher Full Text | Free Full Text

30. Felsenstein J: **CONFIDENCE LIMITS ON PHYLOGENIES: AN APPROACH USING THE BOOTSTRAP.** *Evolution.* 1985; **39**(4): 783–91.
PubMed Abstract | Publisher Full Text

31. Yang Z, Rannala B: **Molecular phylogenetics: principles and practice.** *Nat Rev Genet.* 2012; **13**(5): 303–14.
PubMed Abstract | Publisher Full Text

32. Patané JSL, Martins J Jr, Setubal JC: **Phylogenomics.** *Methods Mol Biol.* 2018; **1704**: 103–87.
PubMed Abstract | Publisher Full Text

33. Simion P, Delsuc F, Philippe H: **To What Extent Current Limits of Phylogenomics Can Be Overcome?** No commercial publisher| Authors open access book; 2020.
Reference Source

34. Scornavacca C, Delsuc F, Galtier N: **Phylogenetics in the Genomic Era**. No commercial publisher| Authors open access book; 2020.
Reference Source

Open Peer Review

Current Peer Review Status: ? ?

Version 1

Reviewer Report 02 June 2021

https://doi.org/10.5256/f1000research.26251.r84387

? **Denis Baurain**
InBioS - PhytoSYSTEMS, Eukaryotic Phylogenomics, University of Liège, Liège, Belgium

This tutorial illustrates how to use the online OMA Browser and OMA Standalone package to assemble datasets of single-copy orthologous genes for phylogenomic inference. It covers two use cases: 1) all species of interest are present in the OMA Browser and 2) some species are not available and must thus be added by the user using OMA Standalone.

Generally speaking, the structure and writing are clear, the figures and tables are appropriate, and the data and code underlying the examples are well described and made available in two public repositories (Figshare and Zenodo). Therefore, I do not have major comments. However, I think that it is possible to easily improve a number of minor points in the manuscript. I provide a list below, simply sorted by order of occurrence. There are no line numbers, so I try to provide some context when needed.

- "16 rRNA ribosomal genes" should rather be "small subunit ribosomal RNA (SSU rRNA 16S/18S) gene" (no plural) to encompass the eukaryotic gene as well.

- "see 3 for common pitfalls in phylogenomics": Don't you mean Philippe *et al.* (2017) "Pitfalls in supermatrix phylogenomics[1]" instead?

- "OGs can have at most one representative gene per species": You should probably say a word about recently diverged in-paralogs (which do not harm species phylogenies) because exactly single-copy genes barely exist.

- "among more than 2300 genomes across the tree of life": You might provide a few numbers here: bacteria, archaea, non-animals, non-plants - or at least recall the rationale for genome inclusion in OMA Browser because it is quite incomplete in terms of species diversity.

- "using species in OMA in addition to other genomes not available in the database": It is not clear if OMA Standalone requires conceptual translations of the sequences (i.e., proteins) or if it can translate DNA sequences on the fly; if it cannot, please specify that one needs protein sequences and not genomes to make use of OMA.

- "phylogenetic tree inference softwares": I am not a native English speaker, but "software" is uncountable; maybe you mean "software packages"?

- IQtree, RaxML, Phylobayes, Figtree etc: Please try to use the correct mixture of uppercase and lowercase letters in each name across the manuscript: IQ-TREE, RAxML, PhyloBayes, FigTree.

- A better download address for PhyloBayes would be: http://www.atgc-montpellier.fr/phylobayes/.

- The interface of OMA Browser has slightly changed since the manuscript has been written. Two details that would need updating: "in the upper left of the home screen" is incorrect [for "Explore"] and "Compute" has now become "Download".

- "To speed up the tree inference, set this value to below 1000 genes": One wonders how these <1000 genes would be prioritized and selected; this should be mentioned here.

- "The name of the fasta file is either the species name or the 5-letter UniProt identifier...": This is not specific enough; what about whitespace, dots in strain names, etc? Are they allowed? Similarly, you should better define "other special characters" just below.

- "...and not keeping other unrelated orthology inferences": This is a bit mysterious. I am aware that OMA can do more than what you demonstrate here, but this sentence should probably be slightly expanded with some examples.

- "...use the job-array syntax in the submission script": This is the part where reproducibility is not straightforward. In particular, is there a master OMA instance controlling the parallelization or is this left to the user? From the code snippet provided, it is not clear, albeit the output of the oma-status command suggests that there is some orchestrator freeing the user from this chore. Please explain a bit more.

- "we selected 880 OGs with at least 18 species as a dataset": Considering Figure 4, I understand your point: 90% of 20 species is 18. However, since you start with 18 genomes and add two "new" ones, the re-use of the number "18" is confusing to the user. As the analyses are done, I would not suggest to change this number, but rather to explain directly the 90% rationale in the text.

- "This command generates a MSA file (.aln) for each OG.": This "for" loop is a very fine candidate for parallelization. Please remind the user that they could use a job array for this. Otherwise, they may wait a lot of time staring at the shell.

- "there is much to gain by making phylogenies from multiple gene families": Here, I am strongly opposed to the words "gene families". The whole point of OMA is to identify single-copy (orthologous) genes, not gene families (i.e., including paralogues). Please use instead "from concatenations of multiple genes, rather than from one single gene" (or something similar).

- "because the evolutionary history of a single gene can be misrepresentative of a species evolutionary history": This is said twice in the manuscript (and this is true) but not backed up by specific reasons. Please enumerate a few of them between parentheses.

- In the legend of Figure 3 and 4, please provide the evolutionary model used. I know it is in Table 2, but somewhat hidden in the command line arguments of the software packages.

- "in this case both trees have identical topology" [Figure 4]: This is correct, but this topology is widely different from the two topologies of Figure 3. This shows that phylogenetic artefacts are at play (which is well known for yeasts...) and this might worth mentioning in the sentences about the caveats of phylogenomics. In other words, it is a pity that this use case does not provide a clear phylogenetic solution for the selected genomes.

References

1. Philippe H, Vienne D, Ranwez V, Roure B, et al.: Pitfalls in supermatrix phylogenomics. *European Journal of Taxonomy*. 2017. Publisher Full Text

Is the rationale for developing the new method (or application) clearly explained?

Yes

Is the description of the method technically sound?

Yes

Are sufficient details provided to allow replication of the method development and its use by others?

Partly

If any results are presented, are all the source data underlying the results available to ensure full reproducibility?

Yes

Are the conclusions about the method and its performance adequately supported by the findings presented in the article?

Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Phylogenomics, comparative genomics, software engineering

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 10 Feb 2022

Natasha Glover, Swiss Institute of Bioinformatics, Lausanne, Switzerland

We thank the reviewer for their very helpful, clear, and pertinent remarks. Each original comment is in italics and our responses are in bold.

1. "16 rRNA ribosomal genes" should rather be "small subunit ribosomal RNA (SSU rRNA 16S/18S) gene" (no plural) to encompass the eukaryotic gene as well.

Done.

2. "see 3 for common pitfalls in phylogenomics": Don't you mean Philippe et al. (2017) "Pitfalls in supermatrix phylogenomics[1]" instead?

The previous reference addressed most of the same issues as the one the reviewer suggests, but the latter is more recent and comprehensive. We now cite both.

3. "OGs can have at most one representative gene per species": You should probably say a word about recently diverged in-paralogs (which do not harm species phylogenies) because exactly single-copy genes barely exist.

We have now updated the text to describe how in-paralogs are handled: "If recently diverged in-paralogs are inferred (i.e., co-orthologs), only one of the copies will be selected for the OG. Thus, all members of the OG are still orthologous to each other."

4. "among more than 2300 genomes across the tree of life": You might provide a few numbers here: bacteria, archaea, non-animals, non-plants - or at least recall the rationale for genome inclusion in OMA Browser because it is quite incomplete in terms of species diversity.

We added more details on species selection and species breakdown on major clades. We also updated the numbers to be in line with the latest release of OMA: "...provides orthology information among more than 2400 genomes across the tree of life, selected to maximize taxon coverage and users' needs (Altenhoff *et al.* 2021); to date, there are 1710 Bacteria, 153 Archaea, and 561 Eukaryotes."

5. "using species in OMA in addition to other genomes not available in the database": It is not clear if OMA Standalone requires conceptual translations of the sequences (i.e., proteins) or if it can translate DNA sequences on the fly; if it cannot, please specify that one needs protein sequences and not genomes to make use of OMA.

We added precision on the input format specification for Protocol 2. Since it needs to be compatible with precomputed OMA data, only protein sequence may be used for this Protocol. We updated the text at the end of the Introduction to say "using species in OMA in addition to other proteomes not available in the database, e.g. a proteome obtained from sequencing a new species. By proteome we mean all protein or nucleic acid sequences of protein-coding genes annotated in a genome."

We also changed the later text in "Combining the added genomes with exported OMA

data" to read: "For this procedure, the added genomes data must fulfil certain conditions: Each additional dataset is in the form of a fasta file, containing protein sequences of all coding genes in the corresponding genome. Please note that OMA Standalone can work on nucleic coding sequences when starting from scratch, however for compatibility with the pre-computed OMA data, only protein sequences may be used when combining new and exported data. "

6. "phylogenetic tree inference softwares": I am not a native English speaker, but "software" is uncountable; maybe you mean "software packages"?

We corrected this error, following the reviewer's suggestion.

7. IQtree, RaxML, Phylobayes, Figtree etc: Please try to use the correct mixture of uppercase and lowercase letters in each name across the manuscript: IQ-TREE, RAxML, PhyloBayes, FigTree.

We thank the reviewer for pointing out this oversight. Now all the mentioned software names have been corrected.

8. A better download address for PhyloBayes would be: http://www.atgc-montpellier.fr/phylobayes/.

Thanks, corrected.

9. The interface of OMA Browser has slightly changed since the manuscript has been written. Two details that would need updating: "in the upper left of the home screen" is incorrect [for "Explore"] and "Compute" has now become "Download".

We changed the text to be adapted to the latest version of the browser. We updated Figure 1 as well to reflect the new menus and design.

10. "To speed up the tree inference, set this value to below 1000 genes": One wonders how these <1000 genes would be prioritized and selected; this should be mentioned here.

When the number of markers is limited, Orthologous Groups are selected in decreasing order of species representation. This ensures the genes with representatives in most of the selected species will be prioritized. We adapted the text to reflect this.

11. "The name of the fasta file is either the species name or the 5-letter UniProt identifier...": This is not specific enough; what about whitespace, dots in strain names, etc? Are they allowed? Similarly, you should better define "other special characters" just below.

We agree, this could be clarified. We now changed the text to say:

"The added genomes data must fulfill certain conditions:

-The name of the fasta file should identify the species clearly and uniquely. The

exported genomes from OMA use for example UniProt's mnemonic five-letter species codes. The filename must end with a ".fa" suffix and must not contain any whitespace characters. The filename without the ".fa" suffix is used as the species name throughout the process and result files.

-Each sequence in the fasta file has a clear and unique identifier. We suggest not to use special characters such as brackets, dots, or a pipe character. The reason is that many programs use them for special purposes, e.g. brackets are used in the newick format for tree representation, and the pipe character is often used to separate ids and annotations."

12. "...and not keeping other unrelated orthology inferences": This is a bit mysterious. I am aware that OMA can do more than what you demonstrate here, but this sentence should probably be slightly expanded with some examples.

We added more details concerning Hierarchical Orthologous Group, which is the main thing we referred to here. We changed the text to say: "If the goal is to only generate a dataset for species phylogeny inference (and not to keep other unrelated orthology inferences, such as Hierarchical Orthologous Groups (Zahn-Zabal *et al.* 2020), which are better representation of individual genes' evolutionary histories but take time to compute), one can avoid doing computations and generating output files that are not needed by the following:..."

13. "...use the job-array syntax in the submission script": This is the part where reproducibility is not straightforward. In particular, is there a master OMA instance controlling the parallelization or is this left to the user? From the code snippet provided, it is not clear, albeit the output of the oma-status command suggests that there is some orchestrator freeing the user from this chore. Please explain a bit more.

We try to make this more clear by now saying: "OMA Standalone automatically partitions the work chunks in a static and deterministic way among the specified number of workers. Progress of the entire computation can be checked with the OMA Status command (see below)."

14. "we selected 880 OGs with at least 18 species as a dataset": Considering Figure 4, I understand your point: 90% of 20 species is 18. However, since you start with 18 genomes and add two "new" ones, the re-use of the number "18" is confusing to the user. As the analyses are done, I would not suggest to change this number, but rather to explain directly the 90% rationale in the text.

We reformulated a bit to make the reasoning more explicit. We now say: "Once the computation finished, we selected 880 OGs that included at least 90% of the 20 species --18 -- as a dataset to construct a tree."

15. "This command generates a MSA file (.aln) for each OG.": This "for" loop is a very fine candidate for parallelization. Please remind the user that they could use a job array for this. Otherwise, they may wait a lot of time staring at the shell.

We now add a reminder that using job arrays is a recommended solution with big datasets. We changed the text to say: “This command sequentially generates a MSA file (.aln) for each OG. Depending on the number of OG and species in your dataset, executing it may take a prohibitive amount of time. If it is the case, we recommend using job-arrays to execute the alignments in parallel.”

16. "there is much to gain by making phylogenies from multiple gene families": Here, I am strongly opposed to the words "gene families". The whole point of OMA is to identify single-copy (orthologous) genes, not gene families (i.e., including paralogues). Please use instead "from concatenations of multiple genes, rather than from one single gene" (or something similar).

We agree this phrase is misleading because even if OMA also provides orthology at the level of whole gene families (HOGs) we are using OGs here - single-copy orthologs. We changed the text as suggested by the reviewer.

17. "because the evolutionary history of a single gene can be misrepresentative of a species evolutionary history": This is said twice in the manuscript (and this is true) but not backed up by specific reasons. Please enumerate a few of them between parentheses.

We now enumerate some reasons and link to a reference on the first time this is mentioned, in the introduction.

18. In the legend of Figure 3 and 4, please provide the evolutionary model used. I know it is in Table 2, but somewhat hidden in the command line arguments of the software packages.

We now provide the evolutionary models used in both figures. The legends now say “Comparison of phylogenetic trees computed by IQ-TREE, using an LG substitution model (left), and RAxML, using an LG substitution model, a discrete Gamma model of rate heterogeneity with 8 categories, and empirical amino-acid frequencies (right)...”

19. "in this case both trees have identical topology" [Figure 4]: This is correct, but this topology is widely different from the two topologies of Figure 3. This shows that phylogenetic artefacts are at play (which is well known for yeasts...) and this might worth mentioning in the sentences about the caveats of phylogenomics. In other words, it is a pity that this use case does not provide a clear phylogenetic solution for the selected genomes.

We thank the reviewer for pointing this out. It is true that there are some differences between the species tree obtained from the two protocols and that it must be addressed. The difference corresponds to bipartitions that are not well supported in either species tree but this was not easily apparent in the first version of the manuscript.

We made two major change to address this comment:

- ○ **We changed Figures 3 and 4 to show the least statistically bipartitions in the species tree.**

- **We added a paragraph in the discussion pointing out the difference between the species tree and discussing the importance of accounting for statistical support when constructing a species phylogeny.**

Competing Interests: No competing interests were disclosed.

Reviewer Report 25 May 2021

https://doi.org/10.5256/f1000research.26251.r84386

? **Jianbo Xie**
National Engineering Laboratory for Tree Breeding, College of Biological Sciences and Technology, Beijing Forestry University, Beijing, China

The manuscript provides useful tools to make use of OMA Orthologous Groups to infer a phylogenetic species tree. This manuscript is an interesting study, but I found some shortcomings that the authors should be improved before the manuscript can be considered for indexing. Some marker gene databases, such as Qiyun Zhu *et al.* (2019)[1], have been published - could you integrate the database into your tool? Or describe the advantage of your tool?

Minor:

1. The introduction section should include some similar tools such as PGAP, Orthomcl.

2. Please discuss the advantage of OMA compared with other tools.

3. Could you integrate the steps of analyses to one steps with multiple parameters?

References

1. Zhu Q, Mai U, Pfeiffer W, Janssen S, et al.: Phylogenomics of 10,575 genomes reveals evolutionary proximity between domains Bacteria and Archaea. *Nature Communications*. 2019; **10** (1). Publisher Full Text

Is the rationale for developing the new method (or application) clearly explained?

Yes

Is the description of the method technically sound?

Yes

Are sufficient details provided to allow replication of the method development and its use

by others?

Yes

If any results are presented, are all the source data underlying the results available to ensure full reproducibility?

Yes

Are the conclusions about the method and its performance adequately supported by the findings presented in the article?

Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Genetics; evolution

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 10 Feb 2022

Natasha Glover, Swiss Institute of Bioinformatics, Lausanne, Switzerland

We thank the reviewer for their comments, which are listed below (in italics) with our responses (in bold):

The manuscript provides useful tools to make use of OMA Orthologous Groups to infer a phylogenetic species tree. This manuscript is an interesting study, but I found some shortcomings that the authors should be improved before the manuscript can be considered for indexing. Some marker gene databases, such as Qiyun Zhu et al. (2019)[1], have been published - could you integrate the database into your tool? Or describe the advantage of your tool?

We now mention the marker gene database as the reviewer suggests. This tutorial's aim is mainly to indicate how to use OMA for inferring marker genes from orthology relationships and then construct a species tree. We do not think it is desirable to integrate other tools into our protocol.

One of the clear advantages in using OMA is a clear procedure which can work for any subset of species.

We now make it more clear in the text, and state: "It is sometimes possible to rely on existing marker genes used in large-scale studies (for example (Zhu *et al*. 2019)), but they are generally available only for a subset of species and do not include newly sequenced species."

Minor:
1. The introduction section should include some similar tools such as PGAP, Orthomcl.

In the introduction, we chose to mention only OMA and BUSCO, because they are to our knowledge the only maintained software that provides single-copy orthologs (does not include any paralogs). This is a prerequisite to use the supermatrix approach for species tree inference.

2. Please discuss the advantage of OMA compared with other tools.

We added a line indicating the main advantage of OMA compared to other methods: it is possible to use a precomputed dataset, and also to mix it with local genomic data, which makes it usable in numerous contexts. We now state:

“This type of Orthologous Group is provided by only a few orthology databases such as BUSCO (8) and Orthologous MAtrix (OMA) (Altenhoff *et al*. 2019; Altenhoff *et al*. 2018) and, to our knowledge, only OMA allows for use of both precomputed and user-computed OGs.”

3. Could you integrate the steps of analyses to one steps with multiple parameters?

As the scope of the paper is to provide a tutorial, we do not plan on integrating the steps into only one command line script.

One of the main reasons to not integrate it into a single command line is to leave more choice to the user regarding the non-OMA tools they may use, as well as their parameters. Given the high number of possible parameters at any step, it would make designing and using such a script highly complicated.

A second reason is that such an implementation will need to make technical choices (e.g regarding parallelization) which would make it incompatible with some computing infrastructures.

Competing Interests: No competing interests were disclosed.