Copycat
The Cophylogenetic Analysis Tool

User Manual

March 12, 2012

Contents

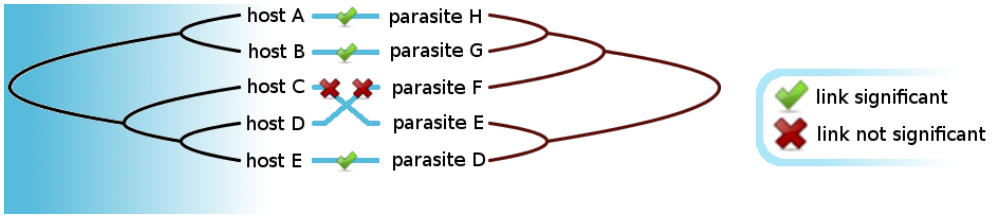

Figure 1: A tanglegram showing host-parasite associations between both five parasite and five host taxa.

`Copycat` is a software tool written in Java which provides an easy and fast access to cophylogenetic analyses. It incorporates a wrapper for the program `Parafit`, which conducts statistical tests for the presence of global congruence between host and parasite phylogenies and for the significance of individual host-parasite associations [1]. A tanglegram is the visualization of host-parasite associations and can be further enhanced by adding `Copycat`'s information on (non-)significant links (see Fig. 1).

The software offers various features, such as the creation of customized host-parasite association data, the reconstruction of host or parasite trees from the NCBI taxonomy, and the computation of several tree statistics. As of April 2007, `Copycat` supports Alexandros Stamatakis' programs `AxParafit` and `AxPcoords`, which are highly optimized versions of `Parafit` and `DistPCoA`, respectively (see section "`Copycat` – Available menu bar options"). This manual describes the features specific to `Copycat`; regarding the principles of the statistical tests implemented in `Parafit`, users are strongly advised to consult [5]. The literature cited in the references section is also suggested for further reading. If you use `Copycat`, you should cite the accompanying paper:

Jan P. Meier-Kolthoff, Alexander F. Auch, Daniel H. Huson, Markus Göker. `Copycat`: Co-phylogenetic Analysis Tool. Bioinformatics, 23(7):898-900, 2007. PDF

[1] Of course, any kind of associations with hosts can be examined in that way, including mutualists. We refer to parasites only just for convenience.

1 Checklist

In prior to installing `Copycat`, please make sure that you have verified the following items:

- ☐ You have at least one of the following operating systems: Mac OS X (tested on 10.6.8), a Win32-compatible or a Linux-based operating system (currently tested under Windows XP, Windows 7 and Linux x86 32bit with GTK 2.0)
- ☐ Your machine is equipped with at least 512 MB of memory. Even though it is not recommended, hardware with less memory can sometimes be used as long as the NCBI taxonomy facilities are not applied.
- ☐ The Java 1.5 (or higher) runtime environment is installed and the Java Binary must be included in the PATH environment variable (this is done automatically by the Java Installer for Windows). Java 1.5 (a.k.a. JDK5) is available here: `http://java.sun.com/javase/downloads/index.jsp`

  Hint: by entering the command "java -version" in a command console the currently installed version is reported.
- ☐ The Mac version additionally requires the GNU scientific library (GSL) installed. A binary installer can be found on `http://ascend4.org/Binary_installer_for_GSL-1.13_on_Mac_OS_X`.
- ☐ Optional: You have downloaded the current NCBI taxonomy file ("taxdmp.zip") from the following URL: `ftp://ftp.ncbi.nih.gov/pub/taxonomy/taxdmp.zip` and placed this file in the "input-data" subfolder. It is located in your `Copycat` installation folder (e.g., /home/john/`Copycat`/input-data/). **As the NCBI taxonomy is updated on a regular basis it is advisable to get the latest version of "taxdmp.zip" from time to time.** The `Copycat` Download already ships with an older version of the NCBI taxonomy data.

2 The first start

Once you have started `Copycat` for the first time, a configuration dialogue will appear[2] - if this dialog won't show up you can access it by choosing the "Setup" option from the top menu (see Fig. 2).

Here, you might want to choose another working directory ("WORKING_DIR") or another directory for your custom data ("USERDATA"). If you are experiencing problems while alternative settings, you might want to default to the well-tested standard settings. The latter will be relevant if `Copycat` asks for a file to open - then this directory will be prompted first. To proceed, please click "apply" followed by "save and proceed". By pressing "exit `Copycat`" the whole application will close. If

[2] The following screen shots were made using the Mac version of `Copycat` - the "look & feel" of the Windows and Linux versions differs slightly.

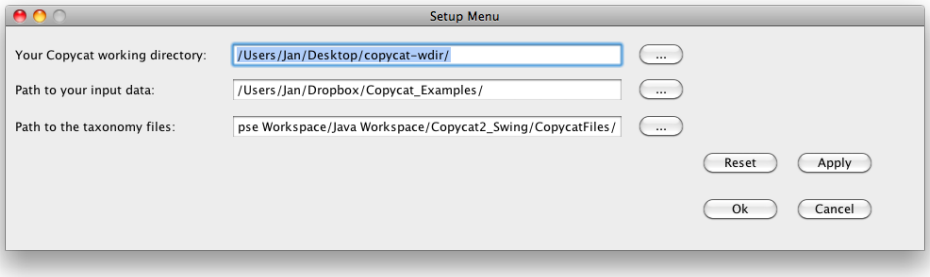

Figure 2: `Copycat`'s configuration dialog

the configuration is not complete the dialogue will be displayed again until everything is well-configured. Now, the `Copycat` application launches. It generates a random session ID (here: ID 9511) and creates a subdirectory for this session within your working directory (e.g., D:\`Copycat`\myworkingdir\ID_9511\). All files and results are now going to be stored in that directory. The contents of that folder and thus the results of this session can be examined by selecting the "View" menu item.

3 Copycat - The first tab

The first tab (see Fig. 3) deals with the creation and pre-processing of an association table containing parasite or host associations.

3.1 Finding NCBI taxonomy IDs

Given an (unresolved) association list containing parasite or host names, the "Resolve Association File" option tries to assign a NCBI taxonomy ID for each organism name. The set of IDs is necessary for inferring the NCBI host or parasite tree. The user specifies a file (unresolved association table) containing one parasite and one host name per line (tab-separated). Then `Copycat` tries to retrieve an NCBI taxonomy ID equivalent to each entry (e.g., the taxonomy ID "9606" for the host name "Homo sapiens").

These results will be automatically displayed in an extra window (see Figure 4).

If that window has accidentally been closed, it will immediately show up again once the user has specified the unresolved association file/table again und "Select Association File". If for both parasite and host name a respective taxon ID is retrieved, this parasite-host-association appears in the final (resolved) association table as a green-colored entry. The set of all host or parasite taxa contained in that table is used for the reconstruction of the NCBI host or parasite tree, respectively. If a certain parasite or host name can't be resolved (e.g., because of a misspelling), the user has the possibility to manually enter a proper NCBI taxonomy ID. The format for this is

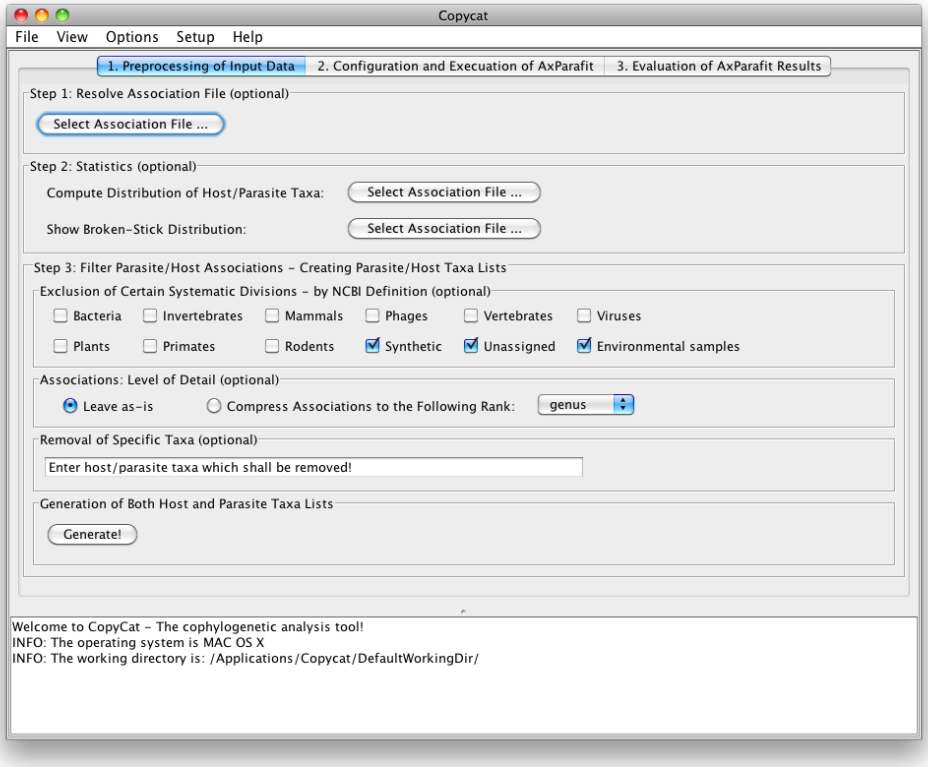

Figure 3: The first tab of `Copycat`

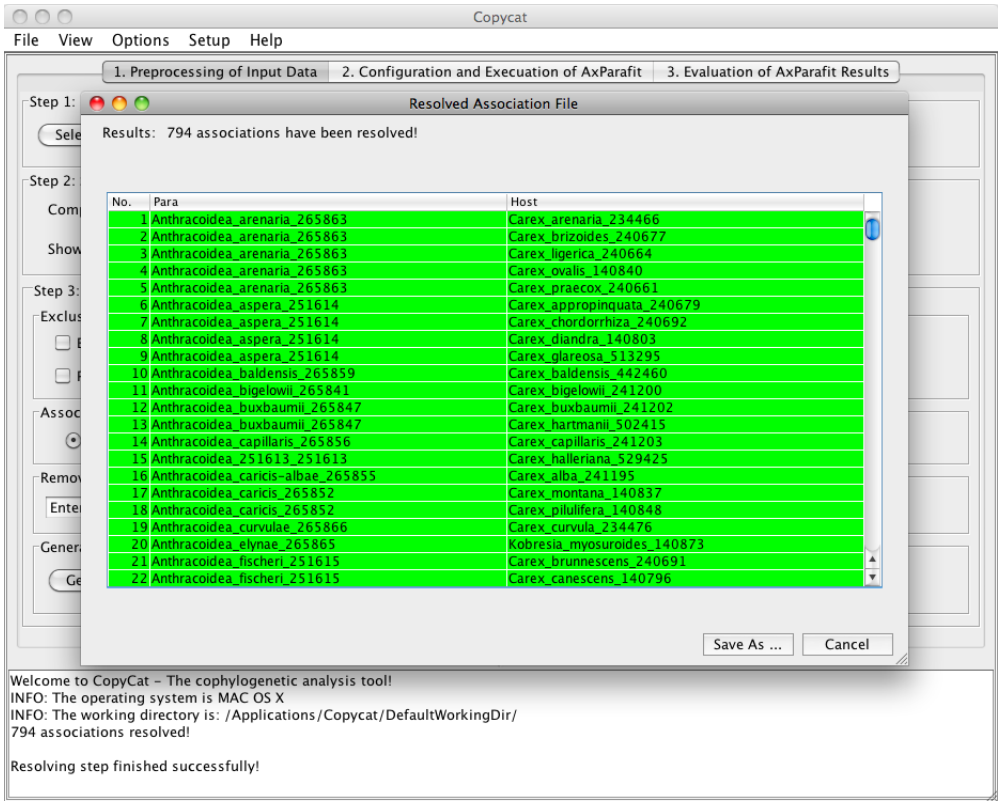

Figure 4: An association table as resolved by `Copycat`. The green entries have a valid ID assigned for both parasite and host as provided by the NCBI Taxonomy Database. Unresolved associations are marked red. The message window at the bottom informs the user once the resolving process has finished by displaying "Resolving step finished successfully!".

the organism name followed by the taxonomy ID, e.g., "Homo sapiens 9606". The user should ascertain that this taxonomy ID is definitely contained in the NCBI taxonomy database. The better and most consistent way of retrieving a correct ID is as follows: the user corrects the organism name and then applies the changes by pressing the "apply" button. By entering a name available in the NCBI taxonomy, the respective organism field in the association table should then yield the corresponding ID. Finally, all resolvable associations are written to a file after having selected the "dump results to working directory" button/option. Along the new association file, two files containing only the parasite and host names are written to the same directory.

3.1.1 Characteristics of the association table

The user hits the "select association file"-button and then specifies an association table (AT; refer to first step). The AT is read and the following features are reported to the message window:

- The distribution of the taxonomical ranks within the set of hosts and the set of parasites respectively (in accordance to the NCBI taxonomy)
- The affiliation of each parasite and host taxon, respectively, to one of the following divisions:

  division 0: Archaea

  division 1: Bacteria

  division 2: Eukaryota

  division 3: Viroids

  division 4: Viruses

  division 5: Other

  division 6: Unclassified

  division 7: taxon ID not found in NCBI taxonomy
- The number of associations contained in the AT
- The number of different parasite taxa in AT
- The number of different host taxa in AT
- The estimated size of the association matrix drawn from the AT

N.B.: As a rule, a phylogenetic tree derived from character data (e.g., molecular sequences) contains more information than a taxonomy of the same taxa since the topology of the latter is usually much less resolved. However, if a cophylogenetic study is based on specific marker sequences such as 16SrRNA or ITS, it is limited to species for which there is a common marker gene available. Even though the number of single-locus or even genome sequences is steadily increasing, we presume that NCBI taxonomic data are available for many more taxa than are such orthologous genetic data. (We guess this rule also holds if we consider non-homologous loci for use in supertree reconstruction, let alone the current debate about whether and how to infer supertrees.) However, this may not be true for all taxonomic groups. Since there most likely is a trade-off between the number of taxa and the topological resolution available as input for `Parafit`, the user has to decide whether a certain `Parafit` analysis based on taxonomic data is worth conducting or not. It is therefore necessary to closely examine the number of resolved associations (compared to those of a study of the same taxonomic group but based on character data) as well as the resolution of the host or parasite taxonomy trees, as described below.

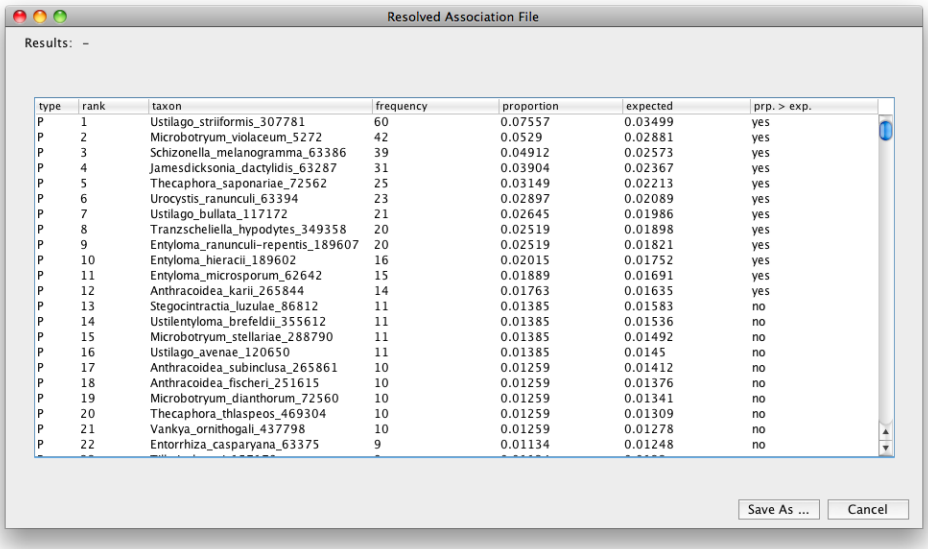

Figure 5: `Copycat`'s representation of the Broken-Stick Distribution.

3.2 Computation of the broken-stick distribution (BSD) for the set of resolved associations

The user selects a resolved AT, which results in a new window displaying the BSD (see Fig. 5).

It basically consists of two parts: the first part shows the BSD for the parasites (indicated by a "P" in the first column of each line), the second part the BSD for the hosts (a "H" as identifier). The further columns show the rank of the taxon, its name, its absolute and relative frequency of occurrence within the associations, and its expected frequency according to the BSD. The last column shows whether the real relative frequency is larger than the expected one. By holding the CTRL button (Mac OS X: Command button) and using the left mouse button, the user can highlight multiple entries within the list. Each entry represents a parasite or host, which is then marked for removal from the association table. Often the entries in the list are highlighted by means of an alternating pattern of dark and light-grey. Several lines sharing the same colour-scheme correspond to the same tied rank and therefore have to be treated equally.

N.B.: The broken stick distribution [6, p. 244] is a standard null model of community structure in ecology. It can be used to predict species' relative abundances but may also be used with other kind of data such as, e.g., eigenvectors [6, p. 410]. Species the relative frequency of which is larger than the corresponding broken stick value occur more frequently than expected by chance. We have included the BSD here since it

may be used to detect host or parasite species which are represented in significantly more associations than others. This is not to say that `Parafit` is unable to deal with widespread parasites; on the contrary, these are treated more consistently in `Parafit` than in other cophylogeny programs [5].

However, a list of associations derived from literature data may, for instance, include many more associations from host species which are medically or economically important and, thus, have been studied more intensively than their less important relatives. If the BSD detects species which are represented in a particularly large number of associations, the user may wish to conduct `Parafit` runs both before and after exclusion of these taxa. In case such taxa display a cophylogenetic behaviour strongly deviating from that of other taxa (i.e., significant vs. insignificant associations, or vice versa), presence or absence of these highly frequent taxa may considerably influence global significance. Even though `Parafit`'s results will (according to Legendre et al. 2002) always be the correct ones given the correctness of the associations, the user may be interested in the impact of such extremely widespread parasites. The brokenstick method provides an objective means to distinguish these from "normal" species.

3.3 Filtering of an association table

This step allows the selection of divisions (as defined within the NCBI taxonomy), whose corresponding taxa and, hence, associations should be completely removed from the table. The "additional filter option" provides a so-called "rank mapping" feature. Each taxon is mapped to its respective parent taxon until the specified rank (e.g. genus or family) is obtained. Redundant entries are removed automatically. The field "remove parasite/host associations containing specific parasite/host taxon-IDs" allows the removal of associations, whose taxa are listed in this box. Valid input is a list of space-separated taxa IDs – each of them with the leading letter "P" or "H" – indicating the taxon's membership in the group of parasite or host taxa (the taxa being selected via the BSD option are listed in this field). For example, "H9606" would result in a removal of those associations having a host associated with the taxonomy ID "9606". "Homo_sapiens_9606" would also be a valid input, although only the ID is of interest.

4 `Copycat` - The second tab

This tab (see Fig. 6) deals with the selection of the input files and parameters for `Parafit` [5] and with the `Parafit` run itself. Here, the most important feature is a wrapper for the program `Parafit` preparing and finally providing the properly formatted input data. This tab is divided into 4 steps.

4.1 Step 1

The number of permutations per row of the association matrix to be conducted by `Parafit` can be specified. `Parafit` requires principal coordinates inferred from host and parasite distance matrices as computed by, e.g., the program `DistPCoA` [4].

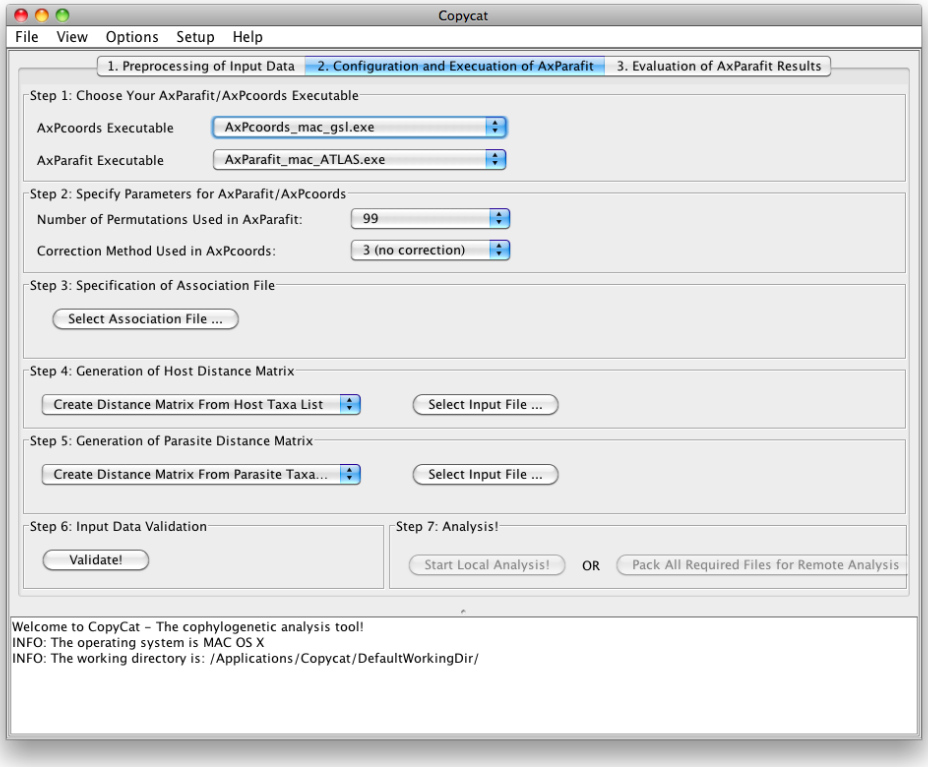

Figure 6: The second tab of `Copycat`.

`DistPCoA` supports two correction methods for negative eigenvalues: Lingoes method and Cailliez method. The user might choose one of these or simply select the "no correction method" option. The advantages and disadvantages of the several corrections methods are explained in, e.g., [3] and [6, pp. 432ff.].

4.2 Step 2

The user selects an association file.

4.3 Step 3 (create host distance matrix)

The user has three possibilities:

- The user selects a pre-existing host distance matrix.
- The user specifies a list of host taxa, which is used to reconstruct the NCBI host tree. The tree is automatically converted to a patristic distance matrix [2].

- The user specifies a host tree, which is used to create the distance matrix.

Once a tree is constructed, several of its features are reported to the message window. "Balance" is a measure of tree balance as described by [1]. Note that "balance" can only be computed for rooted binary trees. "Cherry" is the measure of tree balance suggested by [7] divided by the maximum possible number of cherries (n/2, provided that n is the number of taxa). "Resolution" (described as "Colless' consensus fork index" in [8]) as well as "information content" [9]) are measures of topological resolution. "Resolution" is just the number of internal nodes divided by the maximum possible number of internal nodes (i.e., n-2) and, thus, bound between 0 and 1. A value of 1 indicates full resolution. The cladistic information content has some theoretical advantages, but in case of not fully resolved topologies this measure may rapidly converge towards 0 if the number of taxa increases and, thus, may not be applicable when dealing with large datasets.

N.B.: Due to the presumed trade-off between the number of taxa and the topological resolution available as input for `Parafit`, the user has to base her decision whether to conduct a `Parafit` analysis based on taxonomic data not only on the number of resolved associations, but also on the amount of topological resolution. Even though one of the advantages of `Parafit` is that it does not require fully resolved (binary) trees [5], trees may well display not enough topological resolution (too many polytomies) to be of value in conducting cophylogenetic analyses. As an extreme example, consider a totally unresolved taxonomic tree. Since in that case the eigenvectors of all taxa as output by `DistPCoA` will be identical, such a tree will lead to all associations being insignificant just for trivial reasons.

4.4 Step 4 (create the parasite distance matrix)

This is the same as above.

4.5 Step 5 (validation)

If all input files have been specified and all parameters been set, the user should then hit the "validate the specified data" button. During this validation, the program reports whether the taxon names in the association matrix are consistent with the taxon names in the host and parasite distance matrix or not. If taxon names are present in the association table, which are not contained in the respective distance matrix, the program returns an error. In the opposite case (taxon names from the distance matrix can't be found in the association table) the program offers a "shrink distance matrix"-option, which allows the removal of the respective columns and rows from the distance matrix.

N.B.: In case host or parasite trees are derived by pruning from larger phylogenies, it is much more convenient to change just the association table than to manipulate the trees themselves. This feature of `Copycat` results in a great gain in user flexibility with respect to running `Parafit` with slightly different sets of taxa. If the validation returns no errors, the following two options are enabled:

- "start local analysis": The `Parafit` wrapper is started with the parameters specified above.
- "prepare data for remote analysis": As an alternative, all input files, the `Parafit` wrapper (including `Parafit` and `DistPCoA`) and a setup file are put into a ZIP archive. The archive can be transferred to a highperformance machine. After archive extraction the wrapper can be invoked by the command "java -Xmx512M -jar `Parafit`Wrapper.jar". The "-Xmx" switch denotes the maximum amount of memory the wrapper has for its own disposal (here: 512 MB).

5 `Copycat` - The third tab

This tab (see Figure 7) deals with the evaluation of the `Parafit` results. After the analysis has ended, an output file – called `Parafit`.out – should have been created. In this step, this output file is specified via the "open"-dialogue, together with the host and parasite distance matrix used in that `Parafit` run. The distance matrix files are needed to display the correct organism names, instead of the non-interpretable labels like "Parasite 4" or "Host 17". A sample `Parafit` output as resolved by `Copycat` is shown in Figure 8.

6 `Copycat` – Available menu bar options

The menu bar offers the following options:

6.1 File

6.1.1 "Transfer content of working directory to a place of your choice"

Once your work with `Copycat` is done, you might want to transfer data from the working directory to a directory of your choice. This can be done by selecting this option.

6.1.2 "Download NCBI taxonomy file(s)"

As mentioned above, it is advisable to get the latest NCBI taxonomy file from time to time. The taxonomy itself is steadily improved due to the incorporation of more recent phylogenetic insights, and the total number of both terminal taxa and taxa of higher rank included in the taxonomy dump files is steadily increasing, potentially increasing both the number of resolved associations and the topological resolution in the NCBI-based cophylogenetic analyses. By checking this option, `Copycat` downloads the latest NCBI taxonomy dump file and places it in the appropriate directory.

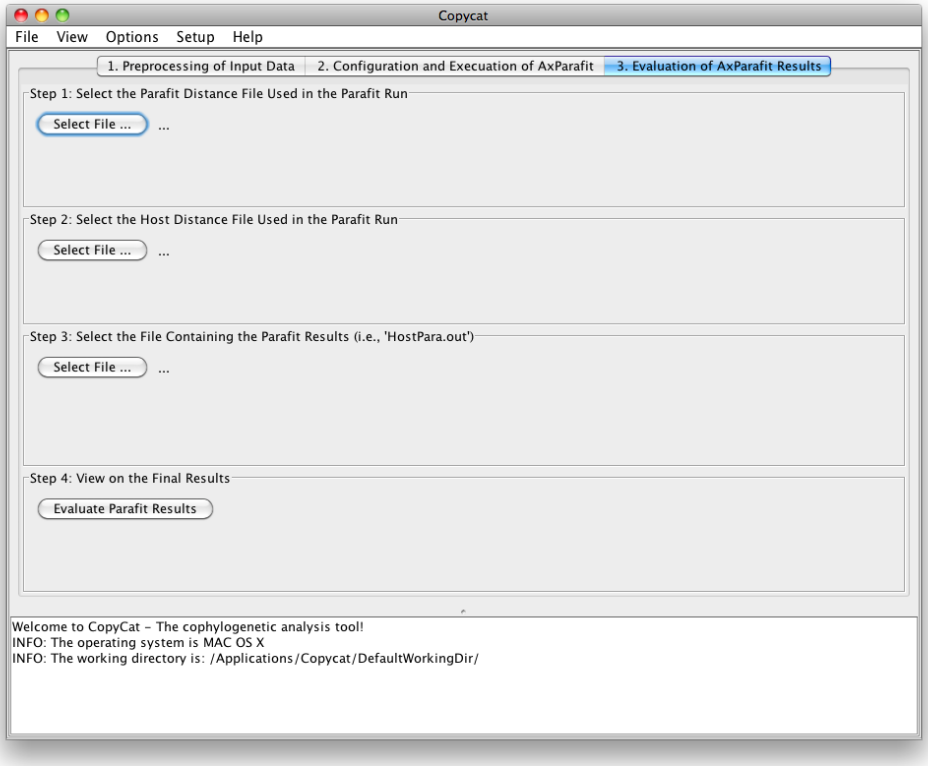

Figure 7: The third tab of `Copycat`.

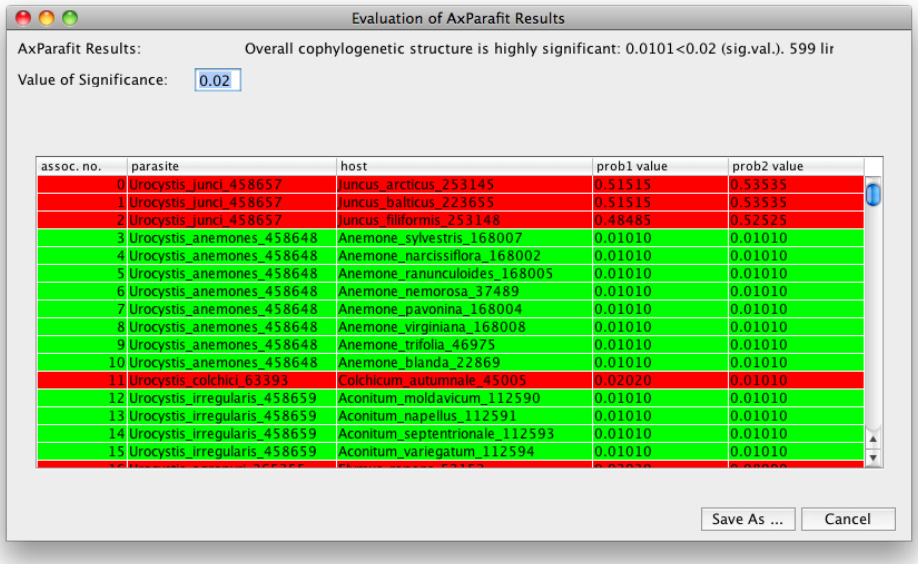

Figure 8: `Copycat`'s representation of the `Parafit` output. By selecting "dump information to working directory" the results are stored in a simple text file (ASCII format). Lines contained in that file marked by a "+" character represent significant links, while lines starting with a "-" character are considered as "non-significant".

6.2 View

6.2.1 "View content of working directory"

This shows a view of the current working directory and its content.

6.3 Options

6.3.1 "Enable Strict Filtering of Association Table"

By default the filter process scans an association table and removes all associations (=lines), which do not fulfill one of the following criteria. Both, parasite and host label have to exist in the NCBI taxonomy and should not be blacklisted. If the user has provided an association table containing custom taxon labels (such as "Patient234" instead of "Homo sapiens"), the program would remove this line from the new associations table due to the first condition ("... have to exist in the NCBI taxonomy ..."). Even though, this condition can be relaxed by unchecking the option "Enable Strict Filtering of Association Table" in the menu bar.

6.3.2 "Use Equal Branch Length (=1) for tree2dist Conversion"

If you have specified a tree file in `Copycat`'s second tab, you might want to have topological distances in the distance matrix resulting from that tree. By checking this option branch lengths are set to 1. By default this option is not checked. In that case, a patristic distance matrix results in which each pairwise distance between two taxa A and B represents the sum of the number of taxa in which A is contained (including A), but not B, and the number of taxa in which B is contained (including B), but not A. Accordingly, by default branch lengths in the NCBI taxonomy tree may be larger than one, representing more information extracted from the taxonomy than just the topology.

6.3.3 "Use `AxPcoords` and `AxParafit` instead of `DistPCoA` and `Parafit`"

By selecting this option, `AxPcoords` and `AxParafit` are used instead of Legendre's programs [5]. `Copycat` supports Alexandros Stamatakis' programs `AxParafit` and `AxPcoords`, which are highly optimized versions of `Parafit` and `DistPCoA`, respectively. `Copycat` searches within the "code" subdirectory, which contains all executables for the analyses. By default, this folder contains the following operating system-dependant executables:

- `Parafit`_win.exe
- `DistPCoA`_win.exe
- `AxParafit`_win.exe
- `AxPcoords`_win.exe

The attribute "win" denotes the executables for the Windows platform (accordingly, "mac" for Macintosh and "linux" for Linux systems are used). In case the user has selected the above option, `Copycat` chooses "`AxParafit`_win.exe" and "`AxPcoords`_win.exe" (operating system: Windows). If you wish to use externally compiled executables (e.g., using the ACML[3] or MKL[4] libraries), they should obey to the naming convention [PROGRAM]_[OS]_[LIBRARY] (e.g., `AxParafit`_win_MKL.exe or `AxPcoords`_win_MKL.exe) and moved into the "code" subdirectory. `Copycat` will detect and use them for the next computation. Such an optimized executable will always be preferred to the default one. Once the user has selected the `AxPcoords`/`AxParafit` option, the option "correction method used in `DistPCoA`" (the second item on the second tab) will be disabled because correction methods for negative eigenvalues are not supported by `AxPcoords`.

6.4 Setup

6.4.1 "Show setup menu at next program start"

If you want to change the working directory, you just have to check this option. On the next start of `Copycat` the configuration dialogue will appear again.

7 Tutorial - A step-by-step example run of `Copycat`

This tutorial focuses on the kind of input needed for certain steps in `Copycat` and shows the output produced by that input. The underlying data set for this example run is the list of European smut fungi and their hosts from Vánky [10, 11]. This data set does not contain parasite or host trees or distance matrices, so we have to construct them using the NCBI taxonomy.

1. The input: An association table of smut fungi (parasites) and their respective hosts. Here is an excerpt of the input file 'smut_fungi_association_table.txt':

2. Resolving the association table: Each parasite/host contained in the NCBI Taxonomy Database has a unique taxonomy ID. This step tries to gather these IDs. Each association containing both, a parasite and a host with a valid ID, is used in the resulting so-called "resolved association table". A representation of an unresolved association table is shown below.

   [!ht]

3. Generation of the parasite/host taxa lists. For the `Parafit` analysis [5] of this association data, we first need to draw two lists from the resolved association table: one containing all parasite taxa and another one containing all host taxa. This is achieved by selecting 'apply settings to association list'. Here, the specified association list can be filtered in regard to certain criteria

[3] AMD © Core Math Library
[4] Intel © Math Kernel Library

parasites	hosts
Anthracoidea altera	Carex fuliginosa
Anthracoidea angulata	Carex hirta
Anthracoidea arenaria	Carex arenaria
Anthracoidea arenaria	Carex brizoides
Anthracoidea arenaria	Carex ligerica
Anthracoidea arenaria	Carex ovalis
Anthracoidea arenaria	Carex praecox
Anthracoidea aspera	Carex appropinquata
Anthracoidea aspera	Carex chordorrhiza
Anthracoidea aspera	Carex diandra
Anthracoidea aspera	Carex glareosa
Anthracoidea baldensis	Carex baldensis
...	...

Table 1: The unresolved association table. Parasites are in the left column - hosts in the right column. Each parasite is separated from its respective host by a tab character. The file contains 1853 host-parasite associations.

and as a side-effect the parasite/host taxa lists are written to the working directory. Naturally, the user is not obliged to select certain filter criteria but can simply choose the 'leave associations in their current state' option. Consequently, the specified association table stays untouched. In this tutorial we make use of the latter and issue the association table gained in the previous section. This operation will take a moment. Finally, the following two files appear in the working directory: 'hosts_filtered_using_option_0.txt' and 'parasites_filtered_using_option_0.txt'. The value '0' indicates that we selected the 'leave associations in their current state' option.

4. Creation of a host distance matrix and a parasite distance matrix. The taxa lists from the previous step are now being used for the creation of the respective NCBI host tree and NCBI parasite tree. Once this is done, the respective distance matrices are generated. We switch to `Copycat`'s second tab and select the 'distance matrix from host taxa list' option together with the 'hosts_filtered_using_option_0.txt' file. This results in the call of the `ParafitWrapper`. The wrapper will now try to create the denoted host distance matrix. You might want to follow the process of the wrapper by reading the lines in the message window that are marked by the purple "WRAPPER" tag. The wrapper is finished once the message window contains the following three information lines at the very end (here: example values).

   INFO on host tree: Resolution: 0,252525
   INFO on host tree: Balance: input is no rooted binary tree
   INFO on host tree: Information content: Infinity

   The working directory should now contain the files 'hosts.dist' and 'host.out.tree'.

We repeat the procedure for the 'parasites_filtered_using_option_0.txt' file by selecting the 'distance matrix from host taxa list' option in the box below. The distance matrix is finished once the following three lines appear at the end of the message window (here: example values).

> INFO on parasite tree: Resolution: 0,252525
> INFO on parasite tree: Balance: input is no rooted binary tree
> INFO on parasite tree: Information content: Infinity

The working directory should now contain the files 'parasites.dist' and 'parasites.out.tree'. These info messages show the resolution and the phylogenetic information content (which cannot be computed for large numbers of taxa) of the host/parasite tree created. If the resolution is not satisfying enough you have the option to cancel the further cophylogenetic analysis at this stage. If you want to proceed, you have to choose the 'validate button'. The validation step ensures that the taxa contained in the following three files are consistent with respect to each other. This means that a parasite's name contained in the association table should exist in the parasite distance matrix and so forth. If the validation is successful, you will have the option to immediately start the `Parafit` run on this machine or, as an alternative, to pack all relevant files to a zip archive. This archive can be transferred to another, probably more powerful machine. Once the `ParafitWrapper` has started (either on this or another machine) it generates a random session ID (similar to `Copycat`) and creates a respective subdirectory within your working directory. This directory is named after that session ID. The time the wrapper is running any interaction with `Copycat` is blocked. Depending on the size of the input data, the `Parafit` run can be a timeconsuming issue.

5. Analysis of the `Parafit` results. `Parafit` performs tests for both the overall phylogenetic congruence as well as the significance of individual associations.

   These results are listed in the file 'HostPara.out'. `Copycat` reports the location of 'HostPara.out' by printing:

   > WRAPPER: Please, check file [D:\`Copycat`\defaultwdir\ID_3738\ID_8313\ Hostpara.out] for results!

   'HostPara.out' holds the results on the individual links in the following format:

   > Parasite 17 Host154 F1 =******* Prob1 = 0.94600 F2 = -0.00016 Prob2 = 0.99200

   As 'Parasite 17' and 'Host 154' provide no information on the actual organisms, `Copycat` needs to know the location of the parasite and the host distance file used in that specific `Parafit` run. These files should reside in your working directory. Once the three files ('Hostpara.out', parasite distance matrix and host distance matrix) have been specified, `Copycat` opens a new window showing the results. An example is shown in Figure 8.

References

[1] D H Colless. Review of "Phylogenetics: The Theory and Practice of Phylogenetic Systematics" by E. O. Wiley. *Systematic Zoology*, 31(1):100–104, 1982.

[2] J.S. Farris. The meaning of relationship and taxonomic procedure. *Systematic Zoology*, pages 44–51, 1967.

[3] P. Legendre and M. J Anderson. Distance-based redundancy analysis: testing multi-species responses in multi-factorial ecological experiments. *Ecological Monographs*, 69:1–24, 1998.

[4] P. Legendre and M. J. Anderson. Program distpcoa, 1998.

[5] Pierre Legendre, Yves Desdevises, and Eric Bazin. A statistical test for host-parasite coevolution. *Syst Biol*, 51(2):217–234, April 2002.

[6] P Lengendre. *Numerical Ecology, 2nd English edition*. Elsevier, Amsterdam, Amsterdam, 2nd englis edition, 1998.

[7] A. McKenzie and M. Steel. Distributions of cherries for two models of trees. *Mathematical biosciences*, 164(1):81–92, 2000.

[8] DL Swofford. When are phylogeny estimates from molecular and morphological data incongruent. *Phylogenetic analysis of DNA sequences. Oxford Univ. Press, New York.*, pages 295–331, 1991.

[9] J.L. Thorley and R.D.M. Page. RadCon: phylogenetic tree comparison and consensus. *Bioinformatics*, 16(5):486, 2000.

[10] K. Vánky. *European smut fungi*. G. Fischer, Stuttgart, Jena, New York, 1994.

[11] K. Vanky. European smut fungi (Ustilaginomycetes pp and Microbotryales) according to recent nomenclature. *Mycologia Balcanica*, 2:169–177, 2005.