PhyML – Manual

Version 3.0
September 17, 2008

http://www.atgc-montpellier.fr/phyml

Contents

1 Citation

- "A simple, fast and accurate algorithm to estimate large phylogenies by maximum likelihood" Guindon S., Gascuel O. *Systematic Biology* **52**(5):696-704

2 Authors

- Stéphane Guindon and Olivier Gascuel conceived the original PhyML algorithm.
- Stéphane Guindon, Wim Hordjik and Olivier Gascuel conceived the SPR-based tree search algorithm.
- Maria Anisimova and Olivier Gascuel conceived the aLRT method for branch support.
- Stéphane Guindon, Franck Lethiec, Jean-Francois Dufayard and Vincent Lefort implemented PhyML.
- Jean-Francois Dufayard created the benchmark and implemented the tools that are used to check PhyML accuracy and performances.
- Vincent Lefort, Stéphane Guindon, Patrice Duroux and Olivier Gascuel conceived and implemented PhyML web server.
- Stéphane Guindon wrote this document.

3 Overview

PhyML [1] is a software that estimates maximum likelihood phylogenies from alignments of nucleotide or amino acid sequences. It provides a wide range of options that were designed to facilitate standard phylogenetic analyses. The main strengths of PhyML lies in the large number of substitution models coupled to various options to search the space of phylogenetic tree topologies, going from very fast and efficient methods to slower but generally more accurate approaches. It also implements two methods to evaluate branch supports in a sound statistical framework (the non-parametric bootstrap and the approximate likelihood ratio test,)

PhyML was designed to process moderate to large data sets. In theory, alignments with up to 4,000 sequences 2,000,000 character-long can analyzed. In practice however, the amount of memory required to process a data set is proportional of the product of the number of sequences by their length. Hence, a large number of sequences can only be processed provided that they are short. Also, PhyML can handle long sequences provided that they are not numerous. With most standard personal computers, the "comfort zone" for PhyML generally lies around 100-200 sequences less than 2,000 character long. For larger data sets, we recommend using other software's such as RAxML [2] or GARLI [3] or Treefinder (http://www.treefinder.de).

4 Installing PhyML

4.1 Sources and compilation

The sources of the program are available free of charge by sending an e-mail to Stéphane Guindon at `guindon@lirmm.fr` or `guindon@stat.auckland.ac.nz`.

The compilation on UNIX-like systems is fairly standard. It is described in the 'INSTALL' file that comes with the sources. In a command-line window, go to the directory that contains the sources and type:

```
> aclocal;
> autoconf -f;
> automake -f;
> ./configure;
> make;
```

Note – when PhyML is going to be used mostly of exclusively in batch mode, it is preferable to turn on the batch mode option in the Makefile. In order to do so, the file `Makefile.am` needs to be modified: add `-DBATCH` to the line with `DEFS=-DUNIX -D$(PROG) -DDEBUG`.

4.2 Installing PhyML on UNIX-like systems (including Mac OS)

Copy PhyML binary file in the directory you like. For the operating system to be able to locate the program, this directory must be specified in the global variable `PATH`. In order to achieve this, you will have to add `export PATH="/your_path/:$PATH"` to the `.bashrc` or the `.bash_profile` located in your home directory (`your_path` is the path to the directory that contains PhyML binary).

4.3 Installing PhyML on Microsoft Windows

Copy the files `phyml.exe` and `phyml.bat` is the same directory. To launch PhyML, click on the icon corresponding to `phyml.bat`. Clicking on the icon for `phyml.exe` works too but the dimensions of the window will not fit PhyML interface.

4.4 Installing the parallel version of PhyML

Bootstrap analysis can run on multiple processors. Each processor analyses one bootstraped dataset. Therefore, the computing time needed to perform R bootstrap replicates is divided by the number of processors available.

This feature of PhyML relies on the MPI (Message Passing Interface) library. To use it, your computer must have MPI installed on it. In case MPI is not installed, you can dowload it from http://www.mcs.anl.gov/research/projects/mpich2/. Once MPI is installed, a few modification of the file '`Makefile.am`' must be applied. The relevant section of this file and the instruction to add or remove the MPI option to PhyML are printed below:

```
# Uncomment (i.e. remove the '#' character at the begining of)
# the two lines below if you want to use MPI.
# Comment the two lines below if you don't want to use MPI.

# CC=mpicc
# DEFS=-DUNIX -D$(PROG) -DDEBUG -DMPI

# Comment the line below if you want to use MPI.
# Uncomment the line below if you don't want to use MPI.

DEFS=-DUNIX -D$(PROG) -DDEBUG
```

5 Program usage.

PhyML has two distinct user-interfaces. The first interface is probably the most popular. It corresponds to a PHYLIP-like text interface that makes the choice of the options self-explanatory (see Figure 1). The command-line interface is well-suited for people that are familiar with PhyML options or for running PhyML in batch mode.

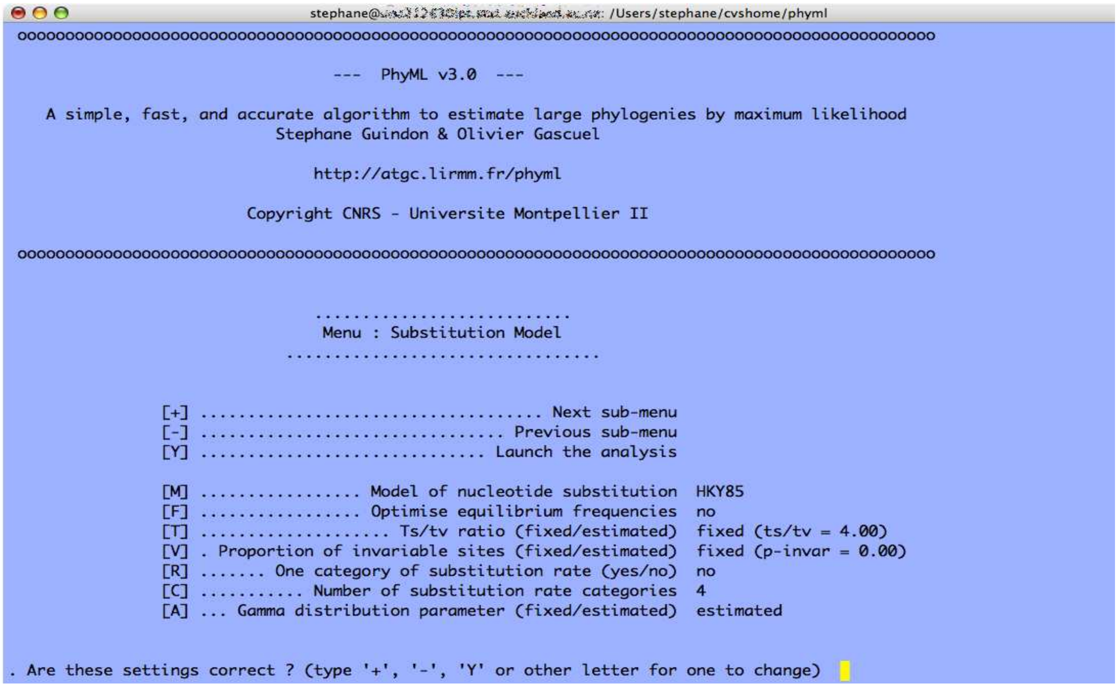
5.1 PHYLIP-like interface

The default is to use the PHYLIP-like text interface (Figure 1) by simply typing '`phyml`' in a command-line window or by clicking on the PhyML icon (see Section 4.3). After entering the name of the input sequence file, a list of sub-menus helps the users to set up the analysis. There are currently four distinct sub-menus:

1. *Input Data*: specify whether the input file contains amino-acid or nucleotide sequences. What the sequence format is (see Section 6) and how many data sets should be analysed.

2. *Substitution Model*: selection of the Markov model of substitution.

3. *Tree Searching*: selection of the tree topology searching algorithm.



```
stephane@[redacted]: /Users/stephane/cvshome/phyml
oooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooo

                                --- PhyML v3.0 ---

   A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood
                         Stephane Guindon & Olivier Gascuel

                              http://atgc.lirmm.fr/phyml

                     Copyright CNRS - Universite Montpellier II

oooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooo


                              ..........................
                               Menu : Substitution Model
                           .................................


              [+] .................................... Next sub-menu
              [-] ................................ Previous sub-menu
              [Y] .............................. Launch the analysis

              [M] ................. Model of nucleotide substitution  HKY85
              [F] ................. Optimise equilibrium frequencies  no
              [T] .................... Ts/tv ratio (fixed/estimated)  fixed (ts/tv = 4.00)
              [V] . Proportion of invariable sites (fixed/estimated)  fixed (p-invar = 0.00)
              [R] ....... One category of substitution rate (yes/no)  no
              [C] ........... Number of substitution rate categories  4
              [A] ... Gamma distribution parameter (fixed/estimated)  estimated


. Are these settings correct ? (type '+', '-', 'Y' or other letter for one to change)
```

Figure 1. PHYLIP-like interface to PhyML.

4. *Branch Support*: selection of the method that is used to measure branch support.

'+' and '-' keys are used to move forward and backward in the sub-menu list. Once the model parameters have been defined, typing 'Y' (or 'y') launches the calculations. The meaning of some options may not be obvious to users that are not familiar with phylogenetics. In such situation, we strongly recommend to use the default options. As long as the format of the input sequence file is correctly specified (sub-menu *Input data*), the safest option for non-expert users is to use the default settings.

The different options provided within each sub-menu are described in what follows.

5.1.1 Input Data sub-menu

```
[D] ................................ Data type (DNA/AA)
```

Type of data in the input file. It can be either DNA or amino-acid sequences in PHYLIP format (see Section 6). Type `D` to change settings.

```
[I] ...... Input sequences interleaved (or sequential)
```

PHYLIP format comes in two flavours: interleaved or sequential (see Section 6). Type `I` to selected among the two formats.

```
[M] ....................... Analyze multiple data sets
```

If the input sequence file contains more than one data sets, PhyML can analyse each of them in a single run of the program. Type `M` to change settings.

5.1.2 Substitution model sub-menu

```
[M] ................. Model of nucleotide substitution
```

```
[M] ................ Model of amino-acids substitution
```

PhyML implements a wide range of substitution models: JC69 [4], K80 [5], F81 [6], F84 [7], HKY85 [8], TN93 [9] GTR [10,11] and custom for nucleotides; LG [12], WAG [13], Dayhoff [14], JTT [15], Blosum62 [16], mtREV [17], rtREV [18], cpREV [19], DCMut [20], VT [21] and mtMAM [22] anf custom for amino acids. Cycle through the list of nucleotide or amino-acids substitution models by typing `M`. Both nucleotide and amino-acid lists include a 'custom' model. The custom option provides the most flexible way to specify the nucleotide substitution model. The model is defined by a string made of six digits. The default string is '`000000`', which means that the six relative rates of nucleotide changes: $A \leftrightarrow C$, $A \leftrightarrow G$, $A \leftrightarrow T$, $C \leftrightarrow G$, $C \leftrightarrow T$ and $G \leftrightarrow T$, are equal. The string '`010010`' indicates that the rates $A \leftrightarrow G$ and $C \leftrightarrow T$ are equal and distinct from $A \leftrightarrow C = A \leftrightarrow T = C \leftrightarrow G = G \leftrightarrow T$. This model corresponds to HKY85 (default) or K80 if the nucleotide frequencies are all set to 0.25. '`010020`' and '`012345`' correspond to TN93 and GTR models respectively. The digit string therefore defines groups of relative substitution rates. The initial rate within each group is set to 1.0, which corresponds to F81 (JC69 if the base frequencies are equal). Users also have the opportunity to define their own initial rate values. These rates are then optimised afterwards (option '`O`') or fixed to their initial values. The custom option can be used to implement all substitution models that are special cases of GTR.

The custom model also exists for protein sequences. It is useful when one wants to use an amino-acid substitution model that is not hard-coded in PhyML. The symmetric part of the rate matrix, as well as the equilibrium amino-acid frequencies, are given in a file which name is given as input of the program. The format of this file is described in the section 6.4.

```
[F] ................. Optimise equilibrium frequencies
```

```
[E] ......... Equilibrium frequencies (empirical/user)
```

```
[F] . Amino acid frequencies (empirical/model defined)
```

For nucleotide sequences, optimising nucleotide frequencies means that the values of these parameters are estimated in the maximum likelihood framework. When the custom model option is selected, it is also possible to give the program a user-defined nucleotide frequency distribution at equilibrium (option `E`). For protein sequences, the stationary amino-acid frequencies are either those defined by the substitution model or those estimated by counting the number of different amino-acids observed in the data. Hence, users should be well aware that the meaning of the `F` option depends on the type of the data to be processed.

```
[T] .................... Ts/tv ratio (fixed/estimated)
```

Fix or estimate the transition/transversion ratio in the maximum likelihood framework. This option is only available when DNA sequences are to be analysed under K80, HKY85 or TN93 models. The definition given to this parameter by PhyML is the same as PAML's one. Therefore, the value of this parameter does *not* correspond to the ratio between the expected number of transitions and the expected number of transversions during a unit of time. This last definition is the one used in PHYLIP. PAML's manual gives more detail about the distinction between the two definitions.

```
[V] . Proportion of invariable sites (fixed/estimated)
```

The proportion of invariable sites, i.e., the expected frequency of sites that do not evolve, can be fixed or estimated. The default is to fix this proportion to 0.0. By doing so, we consider that each site in the sequence may accumulate substitutions at some point during its evolution, even if no differences across sequences are actually observed at that site. Users can also fix this parameter to any value in the $[0.0, 1.0]$ range or estimate it from the data in the maximum-likelihood framework.

```
[R] .......  One category of substitution rate (yes/no)
```

```
[C] ...........  Number of substitution rate categories
```

```
[A] ...  Gamma distribution parameter (fixed/estimated)
```

```
[G] .........'Middle' of each rate class (mean/median)
```

Rates of evolution often vary from site to site. This heterogeneity can be modelled using a discrete gamma distribution. Type `R` to switch this option on or off.

The different categories of this discrete distribution correspond to different (relative) rates of evolution. The number of categories of this distribution is set to 4 by default. It is probably not wise to go below this number. Larger values are generally preferred. However, the computational burden involved is proportional to the number of categories (i.e., an analysis with 8 categories will generally take twice the time of the same analysis with only 4 categories). Note that the likelihood will not necessarily increase as the number of categories increases. Hence, the number of categories should be kept below a "reasonable" number, say 20. The default number of categories can be changed by typing `C`.

The middle of each discretized substitution rate class can be determined using the mean or the median. PAML, MrBayes and RAxML use the mean. However, the median is generally associated with greater likelihoods than the median. This conclusion is based on our analysis of several real-world data sets extracted from TreeBase. Despite this, the default option in PhyML is to use the mean in order to make PhyML likelihoods comparable to those of other phylogenetic software. One must bare in mind that likelihoods calculated with the mean approximation are not directly comparable to the likelihoods calculated using the median approximation.

The shape of the gamma distribution determines the range of rate variation across sites. Small values, typically in the $[0.1, 1.0]$ range, correspond to large variability. Larger values correspond to moderate to low heterogeneity. The gamma shape parameter can be fixed by the user or estimated via maximum-likelihood. Type `A` to select one or the other option.

5.1.3 Tree searching sub-menu

```
[O] .......................... Optimise tree topology
```

By default the tree topology is optimised in order to maximise the likelihood. However, it is also possible to avoid any topological alteration. This option is useful when one wants to compute the likelihood of a tree given as input (see below). Type O to select among these two options.

```
[S] .................. Tree topology search operations
```

PhyML proposes three different methods to estimate tree topologies. The default approach is to use simultaneous NNI. This option corresponds to the original PhyML algorithm [1]. The second approach relies on subtree pruning and regrafting (SPR). It generally finds better tree topologies compared to NNI but is also significantly slower. The third approach, termed BEST, simply estimates the phylogeny using both methods and returns the best solution among the two. Type S to choose among these three choices.

```
[R] ......................... Use random starting tree
```

```
[N] .................. Number of random starting trees
```

When the SPR or the BEST options are selected, is is possible to use random trees rather than BioNJ or a user-defined tree, as starting tree. If this option is turned on (type R to change), five trees, corresponding to five random starts, will be estimated. The output tree file will contain the best tree found among those five. The number of random starts can be modified by typing N.

```
[U] ..................... Input tree (BioNJ/user tree)
```

When the tree topology optimisation option is turned on, PhyML proceeds by refining an input tree. By default, this input tree is estimated using BioNJ [23]. It is also possible for the user to user her/his own tree. This tree should be in Newick format (see Section 6). This option is useful when one wants to evaluate the likelihood of a given tree with a fixed topology, using PhyML. Type U to choose among these two options.

5.1.4 Branch support sub-menu

```
[B] ................ Non parametric bootstrap analysis
```

The support of the data for each internal branch of the phylogeny can be estimated using non-parametric bootstrap. By default, this option is switched off. Typing B switches on the bootstrap analysis. The user is then prompted for a number of bootstrap replicates. The largest this number the more precisely the bootstrap support are. However, for each bootstrap replicate a phylogeny is estimated. Hence, the time needed to analyse N bootstrap replicates corresponds to N-times the time spent on the analysis of the original data set. $N = 100$ is generally considered as a reasonable number of replicates.

```
[A] ................ Approximate likelihood ratio test
```

When the bootstrap option is switched off (see above), approximate likelihood branch supports are estimated. This approach is considerably faster than the bootstrap one. However, both methods intend to estimate different quantities and conducting a fair comparison between both criteria is not straightforward. The estimation of approximate likelihood branch support comes in two flavours: the measured statistics is compared to a χ^2 distribution or a non-parametric distribution estimated using a RELL approximation.

5.2 Command-line interface

The alternative to the PHYLIP-like interface is the command line. Users that do not need to modify the default parameters can launch the program with the '`phyml -i seq_file_name`' command. The list of all command line arguments and how to use them is given in the 'Help' section which is displayed after entering the '`phyml help`' command. The options are also described in what follows.

- `-i` (or `--input`) `seq_file_name`
  `seq_file_name` is the name of the nucleotide or amino-acid sequence file in PHYLIP format.

- `-d` (or `--datatype`) `data_type`
  `data_type` is `nt` for nucleotide (default) and `aa` for amino-acid sequences.

- `-q` (or `--sequential`)
  Changes interleaved format (default) to sequential format.

- `-n` (or `--multiple`) `nb_data_sets`
  `nb_data_sets` is an integer giving the number of data sets to analyse.

- `-b` (or `--bootstrap`) `int`

  - `int` > 0: `int` is the number of bootstrap replicates.
  - `int` = 0: neither approximate likelihood ratio test nor bootstrap values are computed.
  - `int` = -1: approximate likelihood ratio test returning aLRT statistics.
  - `int` = -2: approximate likelihood ratio test returning Chi2-based parametric branch supports.
  - `int` = -4: SH-like branch supports alone.

- `-m` (or `--model`) `model_name`
  `model_name` : substitution model name.

  - *Nucleotide-based models*: `HKY85` (default) `| JC69 | K80 | F81 | F84 | TN93 | GTR | custom`
    The `custom` option can be used to define a new substitution model. A string of six digits identifies the model. For instance, 000000 corresponds to F81 (or JC69 provided the distribution of nucleotide frequencies is uniform). 012345 corresponds to GTR. This option can be used for encoding any model that is a nested within GTR. See Section 5.1.2. *NOTE:* the substitution parameters

of the custom model will be optimised so as to maximise the likelihood. It is possible to specify and fix (i.e., avoid optimisation) the values of the substitution rates only through the PHYLIP-like interface.

  - *Amino-acid based models*: `LG` (default) `WAG | JTT | MtREV | Dayhoff | DCMut | RtREV | CpREV | VT | Blosum62 | MtMam | MtArt | HIVw | HIVb | custom` The `custom` option is useful when one wants to use an amino-acid substitution model that is not available by default in PhyML. The symmetric part of the rate matrix, as well as the equilibrium amino-acid frequencies, are given in a file which name is asked for by the program. The format of this file is described in section 6.4.

- `-f e`, `m`, or "`fA fC fG fT`"
  Nucleotide or amino-acid frequencies.

  - `e` : the character frequencies are determined as follows :
    - *Nucleotide sequences*: (Empirical) the equilibrium base frequencies are estimated by counting the occurence of the different bases in the alignment.
    - *Amino-acid sequences*: (Empirical) the equilibrium amino-acid frequencies are estimated by counting the occurence of the different amino-acids in the alignment.
  - `m` : the character frequencies are determined as follows :
    - *Nucleotide sequences*: (ML) the equilibrium base frequencies are estimated using maximum likelihood.
    - *Amino-acid sequences*: (Model) the equilibrium amino-acid frequencies are estimated using the frequencies defined by the substitution model.
  - "`fA fC fG fT`" : only valid for nucleotide-based models. `fA`, `fC`, `fG` and `fT` are floating numbers that correspond to the frequencies of A, C, G and T respectively.

- `-t` (or `--ts/tv`) `ts/tv_ratio`
  `ts/tv_ratio`: transition/transversion ratio. DNA sequences only. Can be a fixed positive value (e.g., 4.0) or type `e` to get the maximum likelihood estimate.

- `-v` (or `--pinv`) `prop_invar`
  `prop_invar`: proportion of invariable sites. Can be a fixed value in the [0,1] range or type `e` to get the maximum likelihood estimate.

- `-c` (or `--nclasses`) `nb_subst_cat`
  `nb_subst_cat`: number of relative substitution rate categories. Default: `nb_subst_cat=4`. Must be a positive integer.

- `-a` (or `--alpha`) `gamma`
  `gamma`: value of the gamma shape parameter. Can be a fixed positive value or e to get the maximum likelihood estimate. The value of this parameter is estimated in the maximum likelihood framework by default.

- `--use_median`
  The middle of each substitution rate class in the discrete gamma distribution is taken as the median. The mean is used by default.

- `-s` (or `--search`) `move`
  Tree topology search operation option. Can be either `NNI` (default, fast) or `SPR` (a bit slower than `NNI`) or `BEST` (best of NNI and SPR search).

- `-u` (or `--inputtree`) `user_tree_file`
  `user_tree_file`: starting tree filename. The tree must be in Newick format.

- `-o params`
  This option focuses on specific parameter optimisation.

  - `params=tlr`: tree topology (`t`), branch length (`l`) and substitution rate parameters (`r`) are optimised.
  - `params=tl`: tree topology and branch lengths are optimised.
  - `params=lr`: branch lengths and substitution rate parameters are optimised.
  - `params=l`: branch lengths are optimised.
  - `params=r`: substitution rate parameters are optimised.
  - `params=n`: no parameter is optimised.

- `--rand_start`
  This option sets the initial tree to random. It is only valid if SPR searches are to be performed.

- `--n_rand_starts num`
  `num` is the number of initial random trees to be used. It is only valid if SPR searches are to be performed.

- `--r_seed num`
  `num` is the seed used to initiate the random number generator. Must be an integer.

- `--print_site_lnl`
  Print the likelihood for each site in file *_phyml_lk.txt.

- `--print_trace`
  Print each phylogeny explored during the tree search process in file *_phyml_trace.txt.

5.3 Parallel bootstrap

Bootstrapping is a highly parallelizable task. Indeed, bootstrap replicates are independent from each other. Hence, each bootstrap sample can be analysed separately. Modern computers often have more than one CPU. Each CPU can therefore be used to process a bootstrap sample. Using this parallel strategy, performing R bootstrap replicates on C CPUs 'costs' the same amount of computation time as processing $R \times C$ bootstrap replicates on a single CPU. In other words, for a given number of replicates, the computation time is divided by R compared to the non-parallel approach.

PhyML sources must be compiled with specific options to turn on the parallel option (see Section 4.4). Once the binary file (`phyml`) has been generated, running a bootstrap analysis with, say 100 replicates on 2 CPUs, can be done by typing the following command-line:

```
mpirun -np 2 ./phyml -i seqfile -b 50
```

The output files are similar to the ones generated using the standard, non-parallel, analysis (see Section 6).

a)

```
5 80
seq1  CCATCTCACGGTCGGTACGATACACCKGCTTTTGGCAGGAAATGGTCAATATTACAAGGT
seq2  CCATCTCACGGTCAG---GATACACCKGCTTTTGGCGGGAAATGGTCAACATTAAAAGAT
seq3  RCATCTCCCGCTCAG---GATACCCCKGCTGTTG????????????????ATTAAAAGGT
seq4  RCATCTCATGGTCAA---GATACTCCTGCTTTTGGCGGGAAATGGTCAATCTTAAAAGGT
seq5  RCATCTCACGGTCGGTAAGATACACCTGCTTTTGGCGGGAAATGGTCAAT????????GT

ATCKGCTTTTGGCAGGAAAT
ATCKGCTTTTGGCGGGAAAT
AGCKGCTGTTG?????????
ATCTGCTTTTGGCGGGAAAT
ATCTGCTTTTGGCGGGAAAT

5 40
seq1  CCATCTCANNNNNNNNNACGATACACCKGCTTTTGGCAGG
seq2  CCATCTCANNNNNNNNNNGGATACACCKGCTTTTGGCGGG
seq3  RCATCTCCCGCTCAGTGAGATACCCCKGCTGTTGXXXXXX
seq4  RCATCTCATGGTCAATG-AATACTCCTGCTTTTGXXXXXX
seq5  RCATCTCACGGTCGGTAAGATACACCTGCTTTTGxxxxxx
```

Figure 2. **PHYLIP interleaved (a) and sequential (b) formats.**

6 Inputs / outputs

PhyML reads data from standard text files, without the need for any particular file name extension.

6.1 Sequence formats

Alignments of DNA or protein sequences must be in PHYLIP sequential or interleaved format (Figure 6.1). The first line of the input file contains the number of species and the number of characters, in free format, separated by blanks. One slight difference with PHYLIP format deals with sequence name lengths. While PHYLIP format limits this length to ten characters, PhyML can read up to hundred character long sequence names. Blanks and the symbols "(),:" are not allowed within sequence names because the Newick tree format makes special use of these symbols. Another slight difference with PHYLIP format is that actual sequences must be separated from their names by at least one blank character.

An input sequence file may also display more than a single data set. Each of these data sets must be in PHYLIP format and two successive alignments must be separated by an empty line. Processing multiple data sets requires to toggle the '`M`' option in the *Input Data* sub-menu or use the '`-n`' command line option and enter the number of data sets to analyse. The multiple data set option can be used to process re-sampled data that were generated using a non-parametric procedure such as cross-validation or jackknife (a bootstrap option is already included in PhyML). This option is also useful in multiple gene studies, even if fitting the same substitution model to all data sets may not be suitable.

Character	Nucleotide	Character	Nucleotide
A	Adenosine	Y	C or T
G	Guanine	K	G or T
C	Cytosine	B	C or G or T
T	Thymine	D	A or G or T
U	Uracil (=T)	H	A or C or T
M	A or C	V	A or C or G
R	A or G	$-$ or N or X or ?	unknown (=A or C or G or T)
W	A or T		
S	C or G		

Table 1. **List of valid characters in DNA sequences and the corresponding nucleotides.**

Character	Amino-Acid	Character	Amino-Acid
A	Alanine	L	Leucine
R	Arginine	K	Lysine
N or B	Asparagine	M	Methionine
D	Aspartic acid	F	Phenylalanine
C	Cysteine	P	Proline
Q or Z	Glutamine	S	Serine
E	Glutamic acid	T	Threonine
G	Glycine	W	Tryptophan
H	Histidine	Y	Tyrosine
I	Isoleucine	V	Valine
L	Leucine	$-$ or X or ?	unknown (can be any amino acid)
K	Lysine		

Table 2. **List of valid characters in protein sequences and the corresponding amino acids.**

6.1.1 Gaps and ambiguous characters

Gaps correspond to the '`-`' symbol. They are systematically treated as unknown characters "on the grounds that we don't know what would be there if something were there" (J. Felsenstein, PHYLIP main documentation). The likelihood at these sites is summed over all the possible states (i.e., nucleotides or amino acids) that could actually be observed at these particular positions. Note however that columns of the alignment that display only gaps or unknown characters are simply discarded because they do not carry any phylogenetic information (they are equally well explained by any model). PhyML also handles ambiguous characters such as R for A or G (purines) and Y for C or T (pyrimidines). Tables 1 and 2 give the list of valid characters/symbols and the corresponding nucleotides or amino acids.

6.2 Tree format

PhyML can read one or several phylogenetic trees from an input file. This option is accessible through the *Tree Searching* sub menu or the '`-u`' argument from the command line. Input trees are generally used as initial maximum likelihood estimates to be subsequently adjusted by the tree searching algorithm. Trees can be either rooted or unrooted and multifurcations are allowed. Taxa names must, of course, match the corresponding sequence names.

```
((seq1:0.03,seq2:0.01):0.04,(seq3:0.01,(seq4:0.2,seq5:0.05):0.2):0.01);
((seq3,seq2),seq1,(seq4,seq5));
```

Figure 3. **Input trees**. The first tree (top) is rooted and has branch lengths. The second tree (bottom) is unrooted and does not have branch lengths.

6.3 Multiple alignments and trees

Single or multiple sequence data sets may be used in combination with single or multiple input trees. When the number of data sets is one ($n_D = 1$) and there is only one input tree ($n_T = 1$), then this tree is simply used as input for the single data set analysis. When $n_D = 1$ and $n_T > 1$, each input tree is used successively for the analysis of the single alignment. If $n_D > 1$ and $n_T = 1$, the same input tree is used for the analysis of each data set. The last combination is $n_D > 1$ and $n_T > 1$. In this situation, the i-th tree in the input tree file is used to analyse the i-th data set. Hence, n_D and n_T must be equal here.

6.4 Custom amino-acid rate model

The custom amino-acid model of substitutions can be used to implement a model that is not hard-coded in PhyML. This model must be time-reversible. Hence, the matrix of substitution rates is symmetrical. The format of the rate matrix with the associated stationary frequencies is identical to the one used in PAML. An example is given below:

```
0.55
0.51  0.64
0.74  0.15  5.43
1.03  0.53  0.27  0.03
0.91  3.04  1.54  0.62  0.10
1.58  0.44  0.95  6.17  0.02  5.47
1.42  0.58  1.13  0.87  0.31  0.33  0.57
0.32  2.14  3.96  0.93  0.25  4.29  0.57  0.25
0.19  0.19  0.55  0.04  0.17  0.11  0.13  0.03  0.14
0.40  0.50  0.13  0.08  0.38  0.87  0.15  0.06  0.50  3.17
0.91  5.35  3.01  0.48  0.07  3.89  2.58  0.37  0.89  0.32  0.26
0.89  0.68  0.20  0.10  0.39  1.55  0.32  0.17  0.40  4.26  4.85  0.93
0.21  0.10  0.10  0.05  0.40  0.10  0.08  0.05  0.68  1.06  2.12  0.09  1.19
1.44  0.68  0.20  0.42  0.11  0.93  0.68  0.24  0.70  0.10  0.42  0.56  0.17  0.16
3.37  1.22  3.97  1.07  1.41  1.03  0.70  1.34  0.74  0.32  0.34  0.97  0.49  0.55  1.61
2.12  0.55  2.03  0.37  0.51  0.86  0.82  0.23  0.47  1.46  0.33  1.39  1.52  0.17  0.80  4.38
0.11  1.16  0.07  0.13  0.72  0.22  0.16  0.34  0.26  0.21  0.67  0.14  0.52  1.53  0.14  0.52  0.11
0.24  0.38  1.09  0.33  0.54  0.23  0.20  0.10  3.87  0.42  0.40  0.13  0.43  6.45  0.22  0.79  0.29  2.49
2.01  0.25  0.20  0.15  1.00  0.30  0.59  0.19  0.12  7.82  1.80  0.31  2.06  0.65  0.31  0.23  1.39  0.37  0.31

8.66  4.40  3.91  5.70  1.93  3.67  5.81  8.33  2.44  4.85  8.62  6.20  1.95  3.84  4.58  6.95  6.10  1.44  3.53  7.09
```

The entry on the i-th row and j-th column of this matrix corresponds to the rate of substitutions between amino-acids i and j. The last line in the file gives the stationary frequencies and must be separated from the rate matrix by one line. The ordering of the amino-acids is alphabetical, i.e, Ala, Arg, Asn, Asp, Cys, Gln, Glu, Gly, His, Ile, Leu, Lys, Met, Phe, Pro, Ser, Thr, Trp, Tyr and Val.

6.5 Output files

Table 3 presents the list of files resulting from an analysis. Basically, each output file name can be divided into three parts. The first part is the sequence file name, the second part

Sequence file name : 'seq'

Output file name	Content
seq_phyml_tree.txt	ML tree
seq_phyml_stats.txt	ML model parameters
seq_phyml_boot_trees.txt	ML trees – bootstrap replicates
seq_phyml_boot_stats.txt	ML model parameters – bootstrap replicates
seq_phyml_trees.txt	ML trees – multiple random starts

Table 3. **Standard output files**

corresponds to the extension '_phyml_' and the third part is related to the file content. When launched with the default options, PhyML only generates two files: the tree file and the model parameter file. The estimated maximum likelihood tree is in standard Newick format (see Figure 3). The model parameters file, or statistics file, displays the maximum likelihood estimates of the substitution model parameters, the likelihood of the maximum likelihood phylogenetic model, and other important information concerning the settings of the analysis (e.g., type of data, name of the substitution model, starting tree, etc.). Two additional output files are created if bootstrap supports were evaluated. These files simply contain the maximum likelihood trees and the substitution model parameters estimated from each bootstrap replicate. Such information can be used to estimate sampling errors around each parameter of the phylogenetic model. When the random tree option is turned on, the maximum likelihood trees estimated from each random starting trees are printed in a separate tree file (see last row of Table 3).

7 Recommendations on program usage.

The choice of the tree searching algorithm among those provided by PhyML is generally a tough one. The fastest option relies on local and simultaneous modifications of the phylogeny using NNI moves. More thorough explorations of the space of topologies are also available through the SPR options. As these two classes of tree topology moves involve different computational burdens, it is important to determine which option is the most suitable for the type of data set or analysis one wants to perform. Below is a list of recommendations for typical phylogenetic analyses.

1. *Single data set, unlimited computing time.* The best option here is probably to use a SPR search (i.e., straight SPR of best of SPR and NNI). If the focus is on estimating the relationships between species, it is a good idea to use more than one starting tree to decrease the chance of getting stuck in a local maximum of the likelihood function. Using NNIs is appropriate if the analysis does not mainly focus on estimating the evolutionary relationships between species (e.g. a tree is needed to estimate the parameters of codon-based models later on). Branch supports can be estimated using bootstrap and approximate likelihood ratios.

2. *Single data set, restricted computing time.* The three tree searching options can be used depending on the computing time available and the size of the data set. For small data sets (i.e., < 50 sequences), NNI will generally perform well provided that the phylogenetic signal is strong. It is relevant to estimate a first tree using NNI moves and examine the reconstructed phylogeny in order to have a rough idea of the

strength of the phylogenetic signal (the presence of small internal branch lengths is generally considered as a sign of a weak phylogenetic signal, specially when sequences are short). For larger data sets (> 50 sequences), a SPR search is recommended if there are good evidence of a lack of phylogenetic signal. Bootstrap analysis will generally involve large computational burdens. Estimating branch supports using approximate likelihood ratios therefore provides an interesting alternative here.

3. *Multiple data sets, unlimited computing time.* Comparative genomic analyses sometimes rely on building phylogenies from the analysis of a large number of gene families. Here again, the NNI option is the most relevant if the focus is not on recovering the most accurate picture of the evolutionary relationships between species. Slower SPR-based heuristics should be used when the topology of the tree is an important parameter of the analysis (e.g., identification of horizontally transferred genes using phylogenetic tree comparisons). Internal branch support is generally not a crucial parameter of the multiple data set analyses. Using approximate likelihood ratio is probably the best choice here.

4. *Multiple data sets, limited computing time.* The large amount of data to be processed in a limited time generally requires the use of the fastest tree searching and branch support estimation methods Hence, NNI and approximate likelihood ratios rather than SPR and non-parametric bootstrap are generally the most appropriate here.

Another important point is the choice of the substitution model. While default options generally provide acceptable results, it is often warranted to perform a pre-analysis in order to identify the best-fit substitution model. This pre-analysis can be done using popular software such as Modeltest [24] or ProtTest [25] for instance. These programs generally recommend the use of a discrete gamma distribution to model the substitution process as variability of rates among sites is a common feature of molecular evolution. The choice of the number of rate classes to use for this distribution is also an important one. While the default is set to four categories in PhyML, it is recommended to use larger number of classes if possible in order to best approximate the patterns of rate variation across sites [26]. Note however that run times are directly proportional to the number of classes of the discrete gamma distribution. Here again, a pre-analysis with the simplest model should help the user to determine the number of rate classes that represents the best trade-off between computing time and fit of the model to the data.

8 Frequently asked questions

1. *The program crashes before reading the sequences. What's wrong ?*

   - The format of your sequence file is not recognized by PhyML. See Section 6
   - The carriage return characters in your sequence files are not recognized by PhyML. You must make sure that your sequence file is a plain text file, with standard carriage return characters (i.e., corresponding to "`\n`", or "`\r`")

2. *The program crashes after reading the sequences. What's wrong ?*

   - You analyse protein sequences and did not enter the `-d aa` option in the command-line.

- The format of your sequence file is not recognized by PhyML. See Section 6

3. *Does PhyML handle outgroup sequences ?*

   - No, PhyML does not make any difference between outgroup and ingroup sequences. The best solution to take into account outgroup sequences is to run two separate analysis. The first analysis should be conducted on the set of aligned sequences *excluding* the outgroup sequences. This data set is used to estimate the ingroup phylogeny. The second analysis includes the whole set of sequences. The tree corresponds to the ingroup+outgroup phylogeny. The third step is to position the root on the ingroup phylogeny using the ingroup+outgroup phylogeny. The advantage of this technique is to avoid long-branch attraction in the phylogeny estimation due to distantly related outgroup sequences.

4. *Does PhyML estimate clokc-constrained trees ?*

   - No, PhyML cannot do that at the moment. However, future releases of the program will include this feature.

5. *Can PhyML analyse partitioned data, such as multiple gene sequences ?*

   - We are currently working on this topic. Future releases of the program will provide options to estimate trees from phylogenomic data sets, with the opportunity to use different substitution models on the different data partitions (e.g., different genes). PhyML will also include specific algorithms to search the space of tree topologies for this type of data.

9 Acknowledgements

The development of PhyML since 2000 has been supported by the Centre National de la Recherche Scientifique (CNRS) and the Ministère de l'Éducation Nationale.

References

[1] Guindon, S. & Gascuel, O. A simple, fast and accurate algorithm to estimate large phylogenies by maximum likelihood. *Systematic Biology* **52**, 696–704 (2003).

[2] Stamatakis, A. RAxML-VI-HPC: Maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. *Bioinformatics* **22**, 2688–2690 (2006).

[3] Zwickl, D. *Genetic algorithm approaches for the phylogenetic analysis of large biological sequence datasets under the maximum likelihood criterion.* Ph.D. thesis, The University of Texas at Austin (2006).

[4] Jukes, T. & Cantor, C. Evolution of protein molecules. In Munro, H. (ed.) *Mammalian Protein Metabolism*, vol. III, chap. 24, 21–132 (Academic Press, New York, 1969).

[5] Kimura, M. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *Journal of Molecular Evolution* **16**, 111–120 (1980).

[6] Felsenstein, J. Evolutionary trees from DNA sequences: a maximum likelihood approach. *Journal of Molecular Evolution* **17**, 368–376 (1981).

[7] Felsenstein, J. *PHYLIP (PHYLogeny Inference Package) version 3.6a2* (Distributed by the author, Department of Genetics, University of Washington, Seattle, 1993).

[8] Hasegawa, M., Kishino, H. & Yano, T. Dating of the Human-Ape splitting by a molecular clock of mitochondrial-DNA. *Journal of Molecular Evolution* **22**, 160–174 (1985).

[9] Tamura, K. & Nei, M. Estimation of the number of nucleotide substitutions in the control region of mitochondrial DNA in humans and chimpanzees. *Molecular Biology and Evolution* **10**, 512–526 (1993).

[10] Lanave, C., Preparata, G., Saccone, C. & Serio, G. A new method for calculating evolutionary substitution rates. *Journal of Molecular Evolution* **20**, 86–93 (1984).

[11] Tavaré, S. Some probabilistic and statistical problems on the analysis of DNA sequences. *Lectures on Mathematics in the Life Sciences* **17**, 57–86 (1986).

[12] Le, S. & Gascuel, O. An improved general amino-acid replacement matrix. *Mol. Biol. Evol.* (2008).

[13] Whelan, S. & Goldman, N. A general empirical model of protein evolution derived from multiple protein families using a maximum-likelihood approach. *Molecular Biology and Evolution* **18**, 691–699 (2001).

[14] Dayhoff, M., Schwartz, R. & Orcutt, B. A model of evolutionary change in proteins. In Dayhoff, M. (ed.) *Atlas of Protein Sequence and Structure*, vol. 5, 345–352 (National Biomedical Research Foundation, Washington, D. C., 1978).

[15] Jones, D., Taylor, W. & Thornton, J. The rapid generation of mutation data matrices from protein sequences. *Computer Applications in the Biosciences (CABIOS)* **8**, 275–282 (1992).

[16] Henikoff, S. & Henikoff, J. Amino acid substitution matrices from protein blocks. *Proceedings of the National Academy of Sciences of the United States of America (PNAS)* **89**, 10915–10919 (1992).

[17] Adachi, J. & Hasegawa, M. MOLPHY version 2.3. programs for molecular phylogenetics based on maximum likelihood. In Ishiguro, M. *et al.* (eds.) *Computer Science Monographs*, 28 (The Institute of Statistical Mathematics, Tokyo, 1996).

[18] Dimmic, M., Rest, J., Mindell, D. & Goldstein, D. rtREV: an amino acid substitution matrix for inference of retrovirus and reverse transcriptase phylogeny. *Journal of Molecular Evolution* **55**, 65–73 (2002).

[19] Adachi, J., P., W., Martin, W. & Hasegawa, M. Plastid genome phylogeny and a model of amino acid substitution for proteins encoded by chloroplast DNA. *Journal of Molecular Evolution* **50**, 348–358 (2000).

[20] Kosiol, C. & Goldman, N. Different versions of the Dayhoff rate matrix. *Molecular Biology and Evolution* **22**, 193–199 (2004).

[21] Muller, T. & Vingron, M. Modeling amino acid replacement. *Journal of Computational Biology* **7**, 761–776 (2000).

[22] Cao, Y. *et al.* Conflict among individual mitochondrial proteins in resolving the phylogeny of eutherian orders. *Journal of Molecular Evolution* **47**, 307–322 (1998).

[23] Gascuel, O. BIONJ: an improved version of the NJ algorithm based on a simple model of sequence data. *Molecular Biology and Evolution* **14**, 685–695 (1997).

[24] Posada, D. & Crandall, K. Modeltest: testing the model of DNA substitution. *Bioinformatics* **14**, 817–918 (1998).

[25] Abascal, F., Zardoya, R. & Posada, D. Prottest: selection of best-fit models of protein evolution. *Bioinformatics* **21**, 2104–2105 (2005).

[26] Galtier, N. & Jean-Marie, A. Markov-modulated Markov chains and the covarion process of molecular evolution. *Journal of Computational Biology* **11**, 727–733 (2004).

Index