ImOSM: Imbedding One Step Mutations into Phylogenetic Data

Program Manual.

Printed: June 08, 2011.

Minh Anh Thi Nguyen

email: `minh.anh.nguyen(at)univie.ac.at`

Tanja Gesell

email: `tanja.gesell(at)univie.ac.at`

Arndt von Haeseler

email: `arndt.von.haeseler(at)univie.ac.at`

Center for Integrative Bioinformatics Vienna,
Max F. Perutz Laboratories,
Dr. Bohr-Gasse 9/6, A-1030 Vienna, AUSTRIA.

Contents

1 License Agreement

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

2 Introduction

ImOSM is a program to imbed *intermittent evolution* into a given alignment, where intermittent evolution refers to extra substitution(s) which are thrown on branch(es) of the tree to convert a site pattern of the alignment into another site pattern. Extra substitutions are modeled by the one step mutation (OSM) matrix. Thus, ImOSM actually *Im*beds *O*ne *S*tep *M*utations into the alignment.

3 Method

The method is described in the following article (section ImOSM method):

- Minh Anh Thi Nguyen, Tanja Gesell, and Arndt von Haeseler. ImOSM: Intermittent evolution and robustness of phylogenetic methods. *In preparation.*

4 Availability

The program is written in C++ and available free of charge. The executable file currently works under Linux platform. Binary files for MacOS and Windows systems can be compiled from the source code provided.

5 Command-line options and input files

Run `imosm -h` to print a short description of available options.
Usage: `imosm -a <file> -t <file> OPTIONS`

Required input arguments:

`-a <file>`: file contains an alignment in Phylip format.
`-t <file>`: file contains a tree in Newick format.
OPTIONS: must contain at least either `-usebranch` or `-n` (see below). ImOSM first draws the given tree on the screen. Therefore, you may prepare and submit these OPTIONS after having a look at the tree drawn!

General options:

`-h`: Print this help message.

`-ni:` Do NOT prompt interactive interface. Just start the program with provided arguments.

`-seed <number>:` Seed for random number generator.

`-r <num>:` Number of repetitions (i.e. number of output alignments).

`-prefix:` Prefix for the output alignments.

`-sd:` To print the resulting alignments in standard interleave PHYLIP format. By default, the alignments are printed in sequential PHYLIP format.

`-add:` To print additional information.

Tree options:

`-o <file>:` File contains outgroup taxa to root the tree if the input tree is unrooted. Each taxon is on one line. By default, the tree is rooted at the first leaf as appeared in the tree file.

Imbedding options: Either -n or -usebranch must be given!

`-n <number>:` Number of extra substitutions to be imbedded into the alignment. Note that EXACT `<number>` extra substitutions will be calculated! This option can be used together with `-b, -m, -p, -bm, -bp, -mp,-bmp, -bprob -mrates -prates`.

`-usebranch:` Use branch length in the input tree as the EXPECTED number of extra substitutions per site to occur on the branch. This option can be used together with `-m, -p, -mrates, -prates.`

`-b <file>:` File contains a list of branches (brID) on which extra substitutions should be put. You are recommended to root the tree by providing `-o <outgroupFile>`. You may refer to tree drawn afterward and then choose the branches (brIDs).

`-m <file>:` File contains a list of integers (each on one line) indicating the types of the substitutions that should happen. The types of substitutions are encoded by 1 for the transitions $A \Leftrightarrow G$ and $C \Leftrightarrow T$, 2 for the transversions $A \Leftrightarrow C$ and $G \Leftrightarrow T$, and 3 for the other transversions $A \Leftrightarrow T$ and $G \Leftrightarrow C$.

`-p <file>:` File contains a list of non negative integers (STARTED BY 0), each on one line, indicating alignment site positions should be changed by the extra-substitutions.

`-bm <file>:` File contains pairs of branch and substitution type (subs.type) for embedding. Each pair on one line, use space-bar or tab to seperate the two numbers. This file should contain exactly `-n` lines, otherwise the pair (branch, subs.type) may not be paired! If given, `-b` and `-m` will be ignored.

`-bp <file>:` File contains pairs of branch and alignment site position for embedding the extra substitutions. Each pair on one line, use space-bar or tab to seperate the two numbers. This file should contain exactly `-n` lines, otherwise the pair (branch, position) may not be paired! If given, `-b` and `-p` will be ignored.

`-mp <file>:` File contains pairs of substitution type and alignment site position for embedding. Each pair on one line, use space-bar or tab to seperate the two numbers. This file should contain exactly `-n` lines, otherwise the pair (subs.type, position) may not be paired! If given, `-m` and `-p` will be ignored.

`-bmp <file>:` File contains series of branch, mutation type and alignment site position for embedding. Each series on one line, use space-bar or tab to seperate the numbers. This file should contain exactly `-n` lines, otherwise the series (branch, subs.type, position) may not be paired together! If given, all in {`-b, -m, -p, -bm, -bp, -mp`} will be ignored.

`-bprob` If none in -b, -bm, -bp, -bmp, -usebranch is given then this option indicates that the extra substitutions will happen on the branches proportionally to their branch length. This also means the input tree must have branch length. A branch is excluded from the extra substitutions by setting its branch length to 0. If none in {`-b, -bm, -bp, -bmp, -bprob, -usebranch`} is given then branches are selected uniformly.

`-mrates <#1 #2 #3>:` If none in {`-m, -bm, -mp, -bmp`} is given then this option indicates that the classes of the extra substitutions will be selected proportionally to the given rates (probabilities): `#1` for the transition (A-G,C-T), `#2` for the transverstion (A-C,G-T) and `#3` for the transversion (A-T,G-C). `#1` + `#2` + `#3` need NOT to be 1. The program itself will normalize them. If none in {`-m, -bm, -mp, -bmp, -mprob`} is given then substitution types are selected uniformly.

`-prates <file>:` File contains rates for sites in the alignment. Each line contains the site position (STARTED BY 1) and the corresponding rate. This format follows the output from seq-gen. If none in {`-p, -bp, -mp, -bmp`} is given, then alignment sites are selected proportionally to these rates. If none in {`-p, -bp, -mp, -bmp, -prates`} is given, then alignment sites are selected uniformly.

6 Output files

Replace the following `*` by `[prefix]`.

`*.id.mut` store `r` (option `-r`) resulting or disturbed alignments, where `id`= 0 · · · `r-1`.

`*.id.suppl` (when `-add` is indicated) store the additional information, a list containing series of (branch, substitution types, site position) that specify the extra-substitutions actually happen.

7 Installation

To compile the source code, you just need to unpack, configure and make. From a terminal (console) type the following commands:

```
tar xvfz imosm.tar.gz
cd imosm
./configure
make
```

The binary file of ImOSM is produced and located in the sub-directory `src/`.

To run the program from any directory, please make sure that your operating system should be able to locate the program. To do that, copy the binary file of ImOSM into a directory and specify this directory in the global vairable `PATH`. For UNIX system, you have to add `export PATH="directory_path:$PATH"` to the `.bashrc` located in your home directory (`directory_path` is the path to the directory contains the binary files).

8 Example

An example to try the program is given in `example.tar.gz`. The tree `rooted-ori.tree` is a rooted tree with branch lengths: only two external branches have positive branch lengths, the other branches have zero branch lengths. The below command, for instance,

```
imosm -a example.phy -t rooted-ori.tree -usebranch -r 2 -ni
```

will produces two *disturbed* alignments (`-r 2` option): `example.phy.0.mut` and `example.phy.1.mut`.

9 Version History

- June 2011: The first version was launched

10 Credits

Several utilities to manipulate alignment and tree are kindly provided by Bui Quang Minh.

11 Acknowledgment

Support from the Wiener Wissenschafts-, Forschungs- and Technologiefonds (WWTF) is greatly appreciated. A.v.H. also acknowledges the funding from the DFG Deep Metazoan Phylogeny project, SPP (HA1628/9). T.G. and A.v.H. appreciate support from the Austrian GEN-AU project Bioinformatics Integration Network III.