SonicParanoid

User Guide

Salvatore Cosentino & Wataru Iwasaki

http://iwasakilab.bs.s.u-tokyo.ac.jp/sonicparanoid

January 17, 2018

Contents

1 Summary

SonicParanoid is a stand-alone software for the identification of orthologous relationships among multiple species. SonicParanoid is an open source software released under the Apache-2.0 license, implemented in Python3, Cython, and C++, and working on Linux and Mac OSX. The software is designed to run using multiple processors and proved to be up to 1245X faster then InParanoid, 166X faster than Proteinortho, and 172X faster than OrthoFinder 2.0 with an accuracy comparable to that of well-established orthology inference tools. Thanks to its speed, accuracy, and usability SonicParanoid substantially relieves the difficulties of orthology inference for biologists who need to construct and maintain their own genomic datasets.

SonicParanoid was tested on a benchmark proteome dataset provided by the Quest for Orthologs (QfO) consortium [`https://questfororthologs.org`], and its accuracy was assessed, and compared to that of other 13 methods, using a publicly available orthology benchmarking service (Altenhoff et al. 2016).
SonicParanoid is available at:
`http://iwasakilab.bs.s.u-tokyo.ac.jp/sonicparanoid`.

2 Installation

In this section the installation of SonicParanoid on different operative systems will be explained step-by-step, including the installation of the third-party software used by SonicParanoid.

2.1 Hardware requirements

SonicParanoid requires a system with a 64-bit multi-core CPU and 8 GB of memory.

2.2 Supported operative systems

SonicParanoid should work on any Unix-based system provided the required software is installed and the hardware requirements are met. SonicParanoid

was tested to work on the following systems:

- Apple MacOS High Sierra (10.13)
- Ubuntu (ver. 17.10)
- OpenSUSE Leap (ver. 42.3)

2.3 Software requirements

Before downloading SonicParanoid make sure that the following software is installed in your system:

- Python 3.6 or above, with numpy, pandas, cython, sh, and biopython modules installed
- Git version control system (version 2.0 or above)
- GNU GCC compiler (version 5.0 or above)

2.4 Installation, setup, and test on the supported operative systems

Because MMseqs2 is in active development we suggest that you build the version that comes with SonicParanoid, following the instructions in this chapter.

On Linux systems you will require root privileges, please ask your system administrator if you do not have root privileges.

2.4.1 Ubuntu (ver. 17.10 or above)

Ubuntu 17.10 comes with Python3 (ver. 3.6.3) already installed.

Install required software

```
$ sudo apt install git python3-pip python3-pandas python3-biopython build-essential cmake --assume-yes
$ sudo -H pip3 install -U pip setuptools sh cython
```

Obtain SonicParanoid

```
$ git clone https://bitbucket.org/salvocos/sonicparanoid
$ cd sonicparanoid
```

Compile MMseqs2 (from within the sonicparanoid directory)

```
$ cd src
$ tar -zxf mmseqs.tar.gz
$ mkdir build
$ cd build
$ cmake -DCMAKE_BUILD_TYPE=RELEASE DCMAKE_INSTALL_PREFIX=../..
..
$ make
$ make install
$ cd ../../
```

Setup and test (from within the sonicparanoid directory)

```
$ python3 setup_sonicparanoid.py
$ python3 sonicparanoid.py -i test_input -o test_output -t 4 -m fast
```

2.4.2 OpenSUSE Leap (ver. 42.3 or above)

OpenSUSE Leap 42.3 comes with Python3 (ver. 3.4.6) already installed.

Install required development tools

```
$ sudo zypper install -y git python3-devel
$ sudo zypper install -y gcc-c++ gcc7
$ sudo zypper install -y cmake
```

Obtain SonicParanoid

```
$ git clone https://bitbucket.org/salvocos/sonicparanoid
$ cd sonicparanoid
```

Compile MMseqs2 (from within the sonicparanoid directory)

```
$ cd src
$ tar -zxf mmseqs.tar.gz
$ mkdir build
$ cd build
$ cmake -DCMAKE_BUILD_TYPE=RELEASE DCMAKE_INSTALL_PREFIX=../..
..
$ make
$ make install
$ cd ../../
```

Setup and test (from within the sonicparanoid directory)

```
$ python3 setup_sonicparanoid.py
$ python3 sonicparanoid.py -i test_input -o test_output -t 4 -m fast
```

2.4.3 Apple MacOSX High Sierra

Because there are no distributable binaries of MMseqs2 for OSX, it must be compiled locally following the steps below and using the same Git commit suggested below.

Install required development tools

```
Download and install the latest stable version of Xcode from
https://developer.apple.com/download
$ Xcode install and setup
```

Install Homebrew, Python3, and Git

```
$ ruby -e ”$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)”
$ export PATH=/usr/local/bin:/usr/local/sbin:$PATH
$ brew install gcc
```

```
$ brew install cmake
$ brew install git python3
```

Obtain SonicParanoid

```
$ git clone https://bitbucket.org/salvocos/sonicparanoid
$ cd sonicparanoid
```

Compile MMseqs2 (from within the sonicparanoid directory)

```
$ cd src
$ tar -zxf mmseqs.tar.gz
$ mkdir build
$ cd build
$ CXX="$(brew –prefix)/bin/g++-7" cmake -DCMAKE_BUILD_TYPE=RELEASE
-DCMAKE_INSTALL_PREFIX=. ..
$ make
$ make install
$ cd ../../
```

Setup and test (from within the sonicparanoid directory)

```
$ python3 setup_sonicparanoid.py
$ python3 sonicparanoid.py -i test_input -o test_output -t 4 -m fast
```

3 Known issues

This section contains possible solutions to some of the known installation issues in SonicParanoid.

Issue 1

SonicParanoid crashed with the error 'No module named inpyranoid_c', or 'No module named mmseqs_parser_c'

Possible solution

Make sure that Cython has been properly installed before proceeding.

Execute the following two commands from the directory in which SonicParanoid has been installed:

```
$ python3 compile_inpyranoid_c.py build_ext --inplace
$ python3 compile_mmseqs_parser_c.py build_ext --inplace
```

Issue 2

The file with the multi-species ortholog groups contains only the headers.

Possible solution

Make sure that the GCC compiler has been properly installed before proceeding.

From the directory in which SonicParanoid is installed type the following commands:

```
$ cd quick_multi_paranoid
$ make clean
$ make qa
$ cd ..
```

4 Usage

SonicParanoid can be executed through the command line by running the script sonicparanoid.py which is located inside the directory where the program has been installed.
The command:

```
python3 sonicparanoid.py --help
```

provides extra information on the command line parameters.

4.1 Input format

SonicParanoid input files must be valid FASTA formatted files containing protein sequences.

- The file names must not contain dash - symbols nor extensions.
- SonicParanoid will automatically replace blank, tabulation, and additional greater-then > symbols in the headers with pipe | symbols.

4.2 Execution example

The main directory contains a directory (test_input) containing 4 bacterial proteomes that can be used to test that SonicParanoid has been successfully installed.
You can perform the test run using the following command:

```
python3 sonicparanoid.py -i test_input -o test_output -m fast -t 4
```

The above command infers the orthologous relationships among the species which proteomes in FASTA format are stored in test_input, using 4 CPUs, and store the output in the directory test_output.

4.3 Output

Given a run with N input proteomes, the main output directory will have the following structure and content:

- *N * (N - 1) / 2* each containing the ortholog table for one of the possible proteome-proteome combinations for the *N* input proteomes.
- The multi_species directory contains the file with the multi-species ortholog groups (multispecies_clusters_all.tsv)

- species.txt contains all the *N* input file names.
- species_pairs.txt contains a list of all the *N * (N - 1) / 2* combinations for the *N* input proteomes.

4.4 Command line parameters

You can list all the available parameters by typing

```
python3 sonicparanoid.py --help
```

Following is a list of SonicParanoid's parameters and their use:

-i INPUT_DIRECTORY, --input-directory INPUT_DIRECTORY
Directory containing the proteomes (in FASTA format) of the species to be compared. NOTE: the file names MUST NOT contain the '-' nor '.' characters

-o OUTPUT_DIRECTORY, --output-directory OUTPUT_DIRECTORY
The directory in which the results will be stored.

-t THREADS, --threads THREADS
Number of parallel 1-CPU threads to be used. Default=4.

-u UPDATE_NAME, --update UPDATE_NAME
Update the ortholog tables database by adding or removing input proteomes. Performs only required alignments (if any) for new species pairs if required and re-compute the ortholog groups. NOTE: an ID (UPDATE_NAME) for the update must be provided.

-m {ultra-fast, fast, default}, --mode {ultra-fast, fast, default}
SonicParanoid execution mode. The fast mode is suggested for most type of studies. Use default when comparing evolutionary distant species. Default = default

-se, --sensitivity

Sensitivity for MMseqs2 [1, 7.5]. This will overwrite the --mode parameter

-ml, --max-len-diff
Maximum allowed length difference between main orthologs and canditate inparalogs.
Example: 0.5 means one of the sequences could be 2 times longer than the other;
0 means no length difference allowed;
1 means no restriction is applied on lenght difference.
Default = 0.5

-ot, --overwrite-tables
This will force the re-computation of the ortholog tables. Only missing alignment files will be re-computed.

-ow, --overwrite
Overwrite previous runs and execute it again. This can be useful to update a subset of the computaed tables.

-sm, --skip-multi-species
Skip the creation of multi-species ortholog groups.

5 Becnhmarks

SonicParanoid was benchmarked using its three execution modes (*ultra-fast*, *fast*, and *default*), using the Orthology Benchmarking service from the QfO consortium.

6 Test data

SonicParanoid was tested using a benchmark proteome dataset from the Quest for Orthologs consortium (QfO), composed of 66 proteomes, 40 of which from eukaryotes, 5 archaea and 21 bacteria.

The dataset is avaliable at the following link:
ftp://ftp.ebi.ac.uk/pub/databases/reference_proteomes/previous_releases/qfo_release-2011_04

7 Obtaining help

The fastest way to get scientific, and technical support is through the following Gitter chat-room:
https://gitter.im/SonicParanoid/Lobby
which is also avaliable at SonicPanoid's web-page.

8 License

9 Contact

Salvatore Cosentino
salvocos@bs.s.u-tokyo.ac.jp
salvo981@gmail.com

Wataru Iwasaki
iwasaki@bs.s.u-tokyo.ac.jp