

MultiSim.py

Aim

MultiSim.py is a python2 script designed to help MaBoSS users perform multiple simulations by automatically generating configuration files. The inputs of MultiSim.py are a generic configuration file, a profile file (optional) and a mutation file (optional). Note that this script calls FormatMaBoSS.pl.

MultiSim.py is mainly useful when MaBoSS users need to perform multiple simulations of a Boolean model with various parameters (such as initial conditions, rates, active and inactive inputs, etc.). It can also be used to easily perform simulations of multiple mutation profiles. This script also allows the user to perform multiple simulations in parallel on a cluster.

How to run the script

MaBoSS, FormatMaBoSS.pl, the bnd and cfg files need to be in the working directory. Note that the bnd and cfg files must have the same base name.

A profile file (in simple text format) needs to be constructed. Note that it can also be generated on the fly by launching MultiSim.py without providing the profile file. This file is written as blocks of 6 lines, each block representing a simulation.

The profile file

The syntax of a 6 line block is as follows:

- First line: name of the simulation.
- Second line: inactive inputs, separated by commas. Initial states of inactive inputs are set to 0 for the simulation.
- Third line: active inputs, separated by commas. Initial states of active inputs are set to 1 for the simulation.
- Fourth line: transition rates that are modified. In the bnd, for each node, two transitions rates are associated: “\$u_nodename” “\$d_nodename”, like it is done with GINsim export (note that the transition rates you wish to vary need to be have this format for the script to work). In the profile file, the modifications associated to each transition rate correspond to fields separated by commas. Each of these fields is composed of 3 attributes separated by “:”: the first attribute is “u” or “d” for up or down; the second attribute is the name of the node; and the third attribute is the value of the rate.
- Fifth line: initial states that are modified. Each of these initial states correspond to fields separated by commas. Each of these fields is composed of two attributes separated by “:”: the first attribute is the name of the

node; and the second attribute is the probability of the node of being active at the initial state step of the simulation.

- Sixth line: external nodes. Indicate the name of the nodes that we wish to set as external separated by commas.

Example

```
simulation_name
inactive_input1,inactive_input2,...,inactive_inputn
active_inputs1,active_input2,...,active_inputn
u:node1:0.1,d:node1:0.1,u:node2:0.03,u:node4:0.23
node2:0.4,node3:0.9,node1:0,node5:0.4
external_node1,external_node2,...,external_noden
```

An optional mutation file (also in text format) could be constructed. Its syntax is as follows:

```
u:node1,d:node2,...,u:noden
d:node1
d:node1,dnode2
```

Each line represents a mutation profile. Each mutation is separated by commas. Each mutation is composed of two attributes separated by ":" the first attribute is "u" or "d" for knock in or knock out; and the second attribute is a list of nodes. Each mutation profile will be simulated with each simulation selected from the profile file.

Command line

MultiSim.py is using the version 2 of python. FormatMaBoSS.pl needs to be in the working directory. MultiSim.py can be run using the following command line (using version 2 of python):

```
python2 MultiSim.py config_file.cfg profile_file.txt (optional) mutation_file.txt
```

Then simply follow the on-screen instructions. The following libraries need to be installed:

```
string
re
sys
os
```

NOTE FOR CLUSTER USE: The script allows launching simulation on a cluster, but uses by default a qsub command. You may need to modify the submission command to adapt to your own cluster system. To do so: open MultiSim.py with a text editor go to line 230 (232), in quote you can find the content of the bash script used to submit the job to the cluster. Modify the command to adapt

to your own system (check the documentation of your own cluster system for more details) save and quit.

Optionally, in MultiSim.py file, from line 120 to 131, you can directly modify the simulation parameters such as number of runs per simulation, time interval, etc. To do so, just change the second parameter of the re.sub functions to the desired value.

Outputs

The output files will be placed in the working directory.