# BioNetGen: Parallel Implementation

Dipak Barua

July 26, 2014

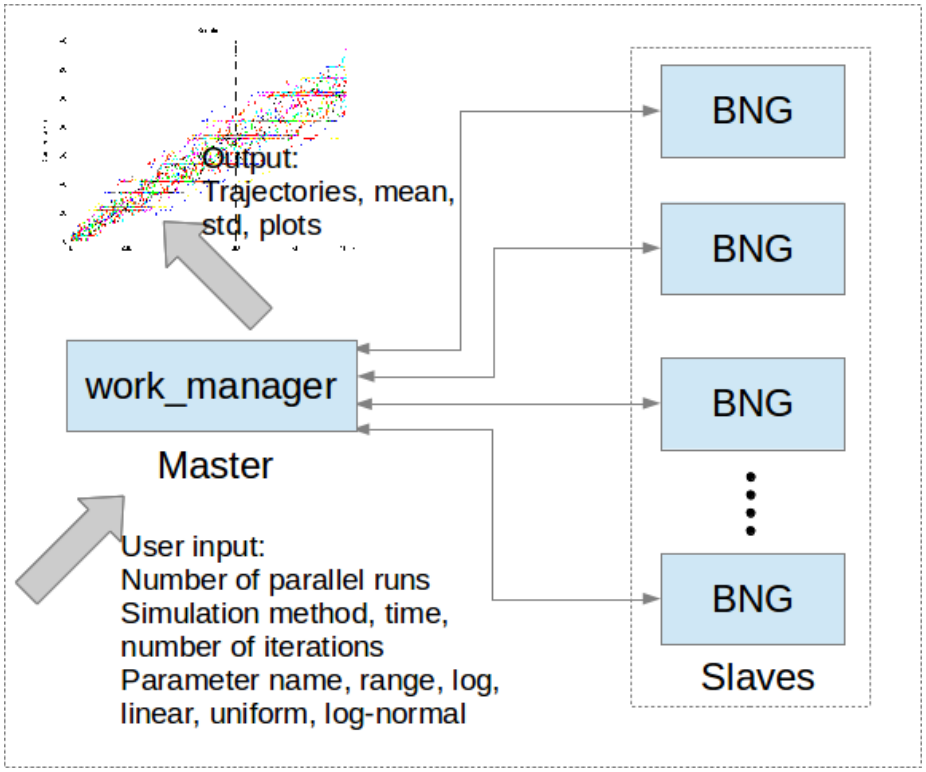


Figure 1: Scheme for parallel BioNetGen simulation.The master program (work_manager) creates parallel child (BioNetGen) processes. An interprocesses MPI communication mediates data exchange between the master program and each child process. After each simulation step or iteration, the master program collects results (time-dependent concentration of species) from each child processe. The master process displays the results after all parallel runs are completed.

# 1 Introduction

The parallel feature enables:

1. parallel stochastic (SSA) simulation.
2. parameter scan (both SSA and ODE).

# 2 Installation

Step 1: Open a terminal and hange directory to work_manager:

```
cd /PathToBioNetGenHomeDir/work_manager
```

Step 2: Execute the following:

```
make clean
make
```

(Note: This will locally install Open MPI, which may take some time. Root access is not necessary for installation.)

# 3 Special requirement

The machine running simulations should be connected to internet.

# 4 Parallel SSA

Step 1: Change directory to `/PathToBNGHomeDir/work_manager/example`. Create a .net file from a .bngl model file by executing BioNetGen (example: `../../BNG2.pl fceri_ji.bngl`). An example .net file, `fceri_ji.net`, is provided in the directory.

Step 2: Execute the following:

```
../bin/run -m fceri_ji.net -t 1000 -n 100 -p ssa -q 100
```

This will launch 100 parallel (`-q 100`) SSA simulations (`-p ssa`). Each run will continue for 1,000 simulation sec (`-t 1000`). Results will be dumped in .gdat files in 100 steps or every 10 sec (`-n 100`).

When finished, uniquely named gdat files will store results for each parallel run. The gdat files will be named after the process ids (PID) of corresponding parallel runs. The files will be stored in the current (example) directory. Two more files, `avg.gdat` and `std.gdat` will store mean and standard deviation of values

from all parallel runs.

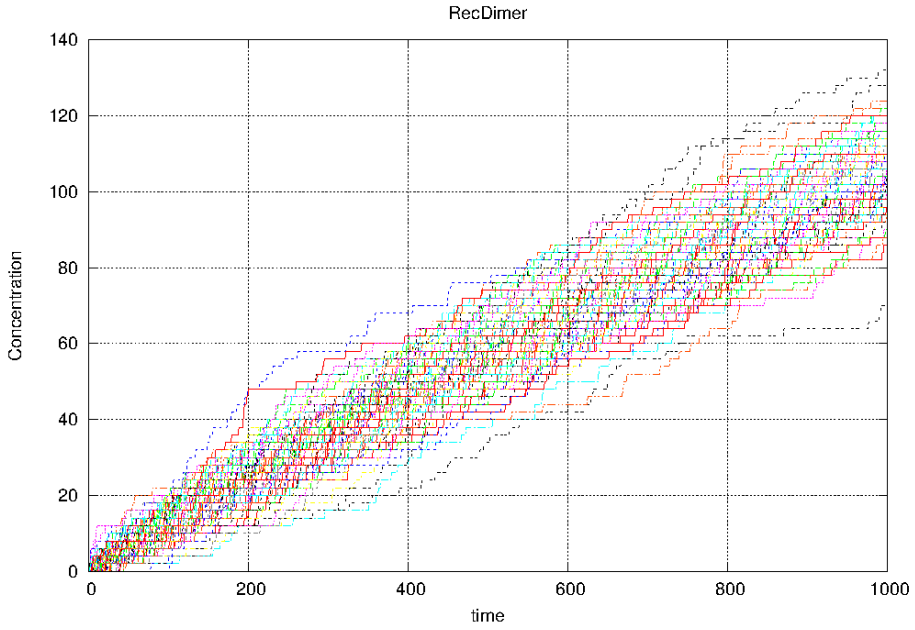


Figure 2: Trajectories from 100 parallel SSA simulations. The results represent time-dependence of an observable species RecDimer in (`fceri_ji.bngl`).

# 5 Parameter Scan (SSA and ODE)

For parameter scan on parameter `kp1` of the model in `fceri_ji.bngl`, execute the following:

For log scale:

```
../bin/run -m fceri_ji.net -t 1000 -n 100 -p ssa -q 100 -log kp1 1e-8
1e-6
```

For linear scale:

```
../bin/run -m fceri_ji.net -t 1000 -n 100 -p ssa -q 100 -lin kp1 1e-8
1e-6
```

For random sampling from log-normal distribution:

```
../bin/run -m fceri_ji.net -t 1000 -n 100 -p ssa -q 100 -lgauss kp1
-7 3
```

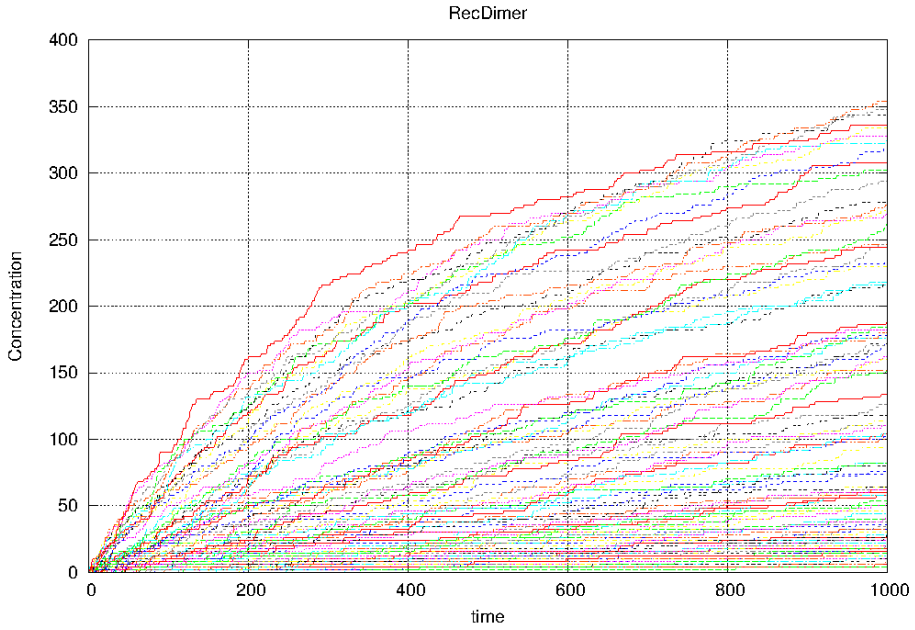


Figure 3: Parameter scan from parallel SSA on `fceri_ji` model. The trajectories represent time-dependent concentrations of an observable species (the same species concentrations plotted in Fig 2). Each trajectory corresponds to a distinct value of a model parameter `kp1`.The values are uniformly distributed in log space between -8 and -6 (actual values are between 1e-8 and 1e-6.)

(Note: In log normal sampling, values immediately after `-lgauss` represent the mean and standard deviation of a parameter in log scale. In the above example, -7 represents log of the mean value of parameter $kp1$ (actual mean 1e-7) and 3 represents standard deviation of the log normal distribution.)

(Note: These commands are applicable for SSA (`-p ssa`). For ode simulation, replace `-p ssa` with `-p cvode`.)

# 6 Command-line Interface

A command-line interface will be displayed while simulations are running in the background. The interface will allow a user to check the progress of simulations using the following commands:

```
conc
```

The above command will display simulation time and concentrations of all ob-

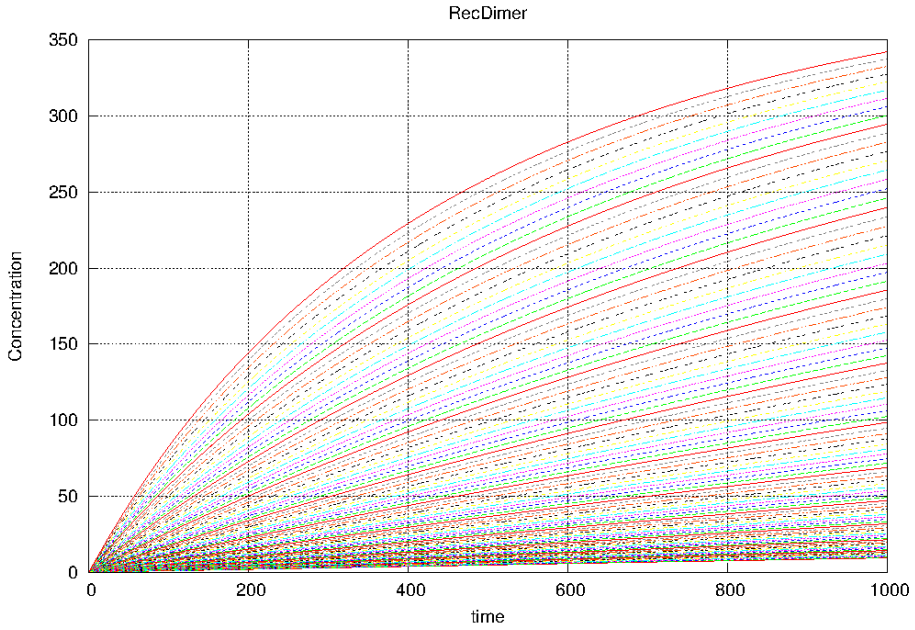


Figure 4: Parameter scan results from parallel ODE simulations of `fceri_ji` model. Trajectories represent the same results as in in Fig 3.

servables in all running processes.

`group--3`

The above command will display concentration of group 3 (if such a group exists in the BioNetgen model) and corresponding simulation time in all parallel runs.

`rank--4`

The above command will display simulation time and concentrations of all observables in a particular run (in this example run 4).

`rank--5 group--2`

The above command will display concentration of group–2 and corresponding simulation time in run 5.

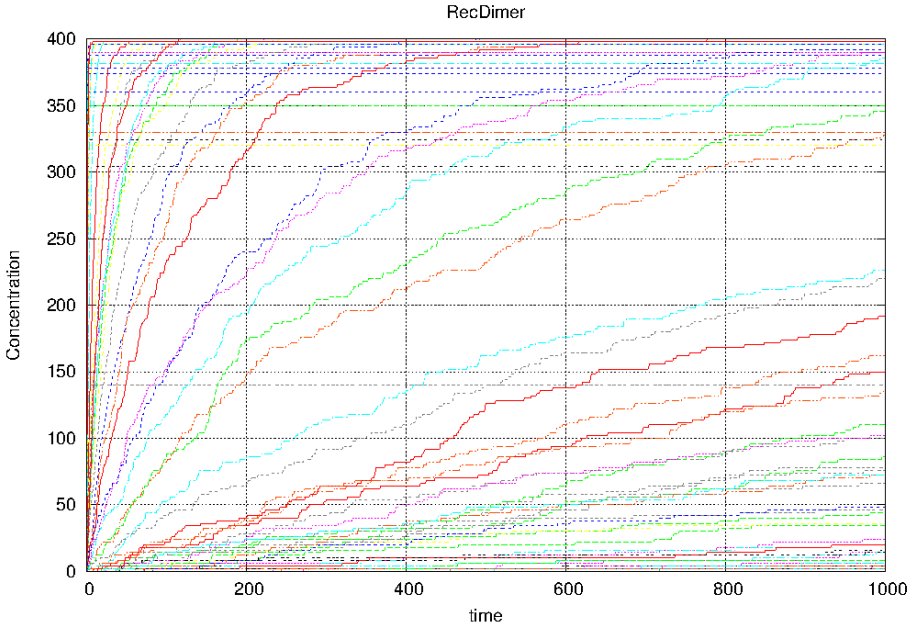


Figure 5: Results from random sampling of parameter `kp1` in `fceri_ji` model. The parameter is sampled from a log-normal distribution (`-lgauss`) with log mean value -7 (actual value 1e-7) and standard deviation 2 (in log space). Each trajectory represents a randomly sampled value of parameter `kp1`.