

Graphical User Interface for Stochastic Simulation of Biochemical Networks

Ravishankar Rao Vallabhajosyula
Postdoctoral Fellow, Keck Graduate Institute
Claremont CA 91786
Document updated: 15 Dec 2006

Introduction

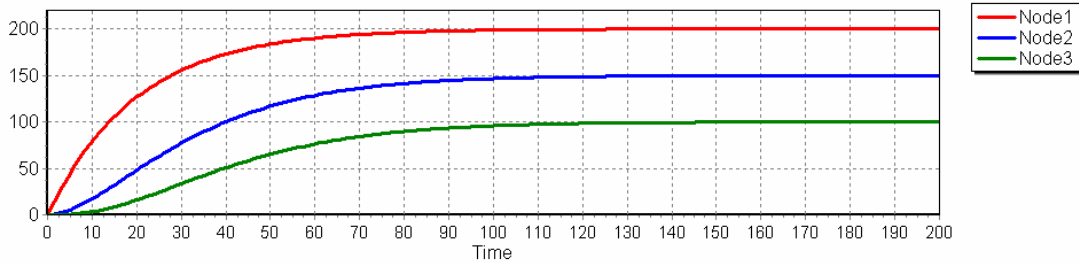
We have developed a user interface for stochastic simulation of biochemical networks. This tool allows users to load a model in SBML format, generate data using a stochastic simulator (currently Gillespie simulator based on SSA) and carry out statistical analysis on the results. These include computation of probability density histograms, ensemble means and averages, power spectral densities and transfer functions.

User Interface Features

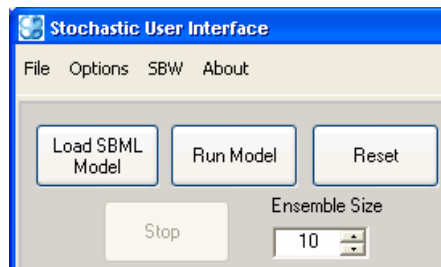
This document describes the features available in the application using a simple example. The network used here has three internal species and two boundary species. This model uses simple mass-action kinetics, although the simulator can support more complex rate-laws including that involve regulatory kinetics. The network used is shown below.



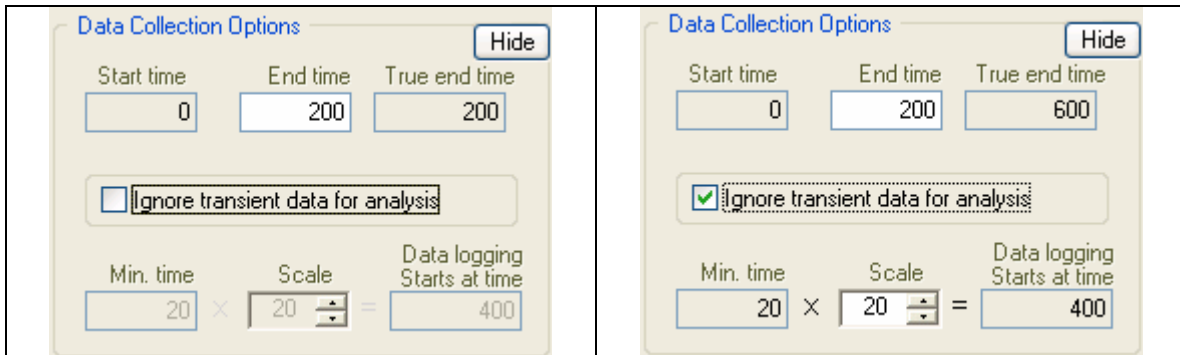
The initial concentration of Node0 is set 100. Other Nodes have initial concentration of 0. The rate constants of reactions from left to right have been set to 0.1, 0.05, 0.066667 and 0.1 respectively. The steady state values are Node1 (200), Node2 (150) and Node3 (100). The deterministic simulation results for this model are shown below.



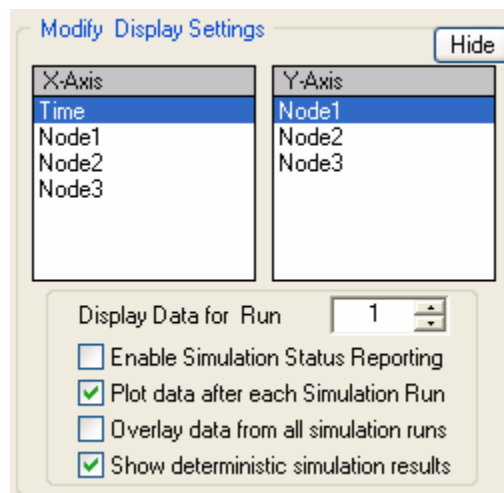
The stochastic simulation user interface allows models in SBML to be analyzed either by selecting File > Load or by clicking on the LoadSBML button, as shown below.



This panel also has buttons to start and stop the simulation. The users can also set the number of runs for the stochastic simulation. The start and end time for the simulation can be set in the data Data collection Options group box, which also includes an option to include or exclude transients from subsequent statistical analyses. If this option is disabled, the true end time, that is, the time when the simulation ends is equal to the data end time (in this case 200). On the other hand, if this option is enabled, the fastest reaction rate in the network is computed and the time for transients (Min. Time) is computed as its inverse. It is usually common to ignore a data length of 50 times this smallest time, but in this case, the default value is set to 20x, and can be edited if needed. For the model under consideration, the Min. Time is 20, and a scale of 20 is used, yielding a transient data time of length 400. The user specified end-time (200) is then added, increasing the true end time to 600 (200 + 400). In essence, the simulation runs from a start time of 0 to end time of 600, but the data points corresponding till 400 are ignored when carrying out the subsequent statistical analysis.

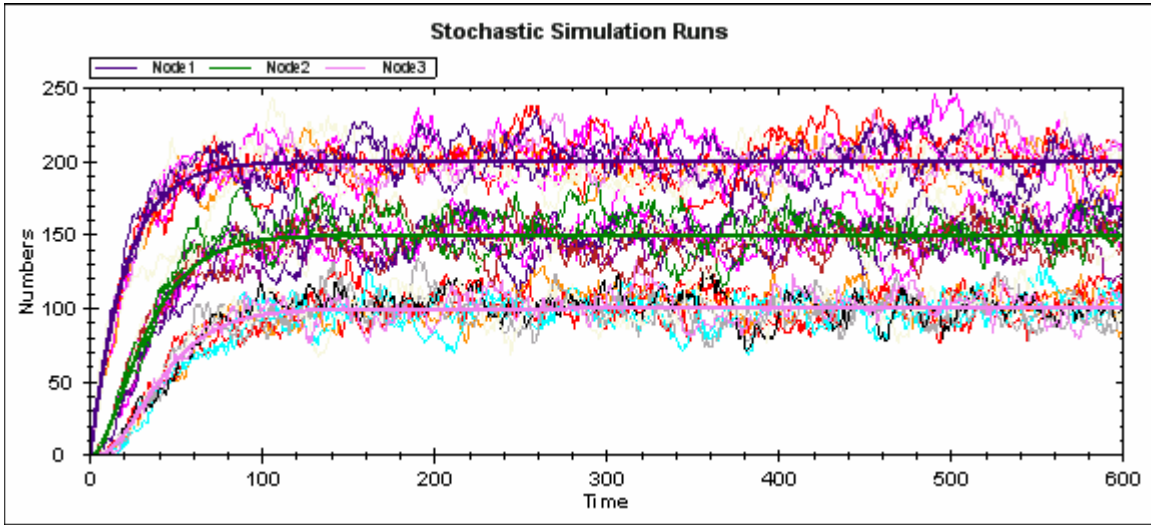


Once the simulation has been started (here, it is assumed that the transient ignore option has been enabled), the user has the option to select if one or more species data can be plotted. These can be controlled from the “Modify Display Settings” panel, shown below.

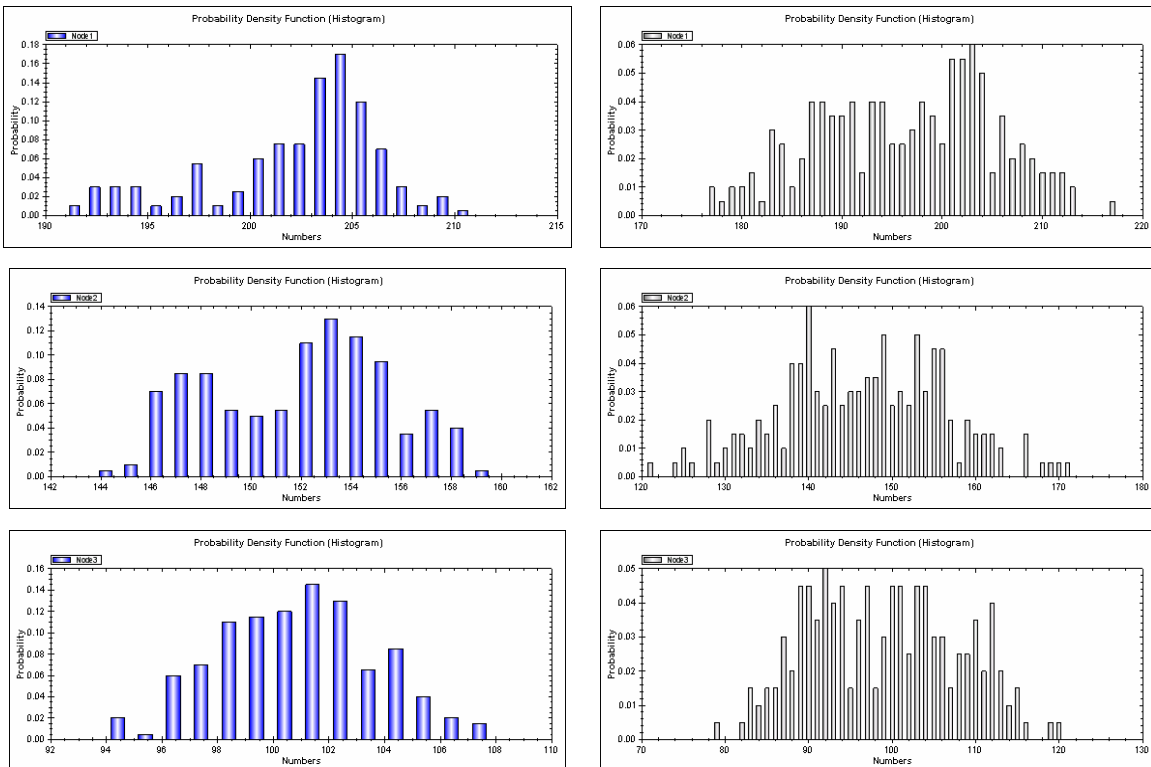


This panel shows the list of species, which can be plotted against time or against one another (phase-plot mode). Additionally, users can overlay data from all runs once they have been completed, or plot the species data for each run as they are generated by the

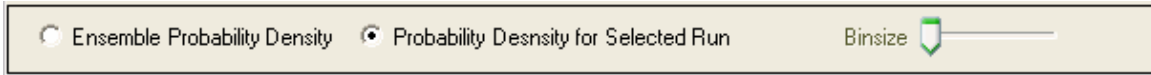
simulator. Further, deterministic simulation results can be overlaid for comparison. The overlaid data from the stochastic simulation for the load network is shown below.



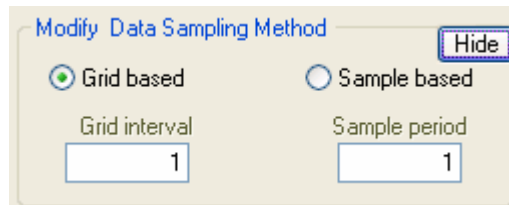
The tool also allows the probability distribution for each species to be constructed, both for each run, as well as for the entire ensemble. These can be obtained by selecting the species of choice in the display settings tab. In the figure below, we show the probability density histogram (note that the data corresponding to the transients has been ignored in the construction of this plot) for the ensemble, as well as for run 5.



The probability distribution histograms for ensemble or individual runs can be toggled by selecting the controls at the bottom of the Stochastic Simulation Results tab, shown here



It should be noted that the simulator produces output between the start and end times specified by the user. However, the data density, that is, the number of points between the start and end times can be controlled by setting the way the simulator steps through time. In this case, two options are offered, which are shown below



The grid based sampling generates data on a equally spaced time grid between the start and end times. This is the default option for the application, as it is feasible to carry out subsequent statistical analysis, owing to the fact that each data run is has the same length with points spread on the same time vector. It is therefore possible to build population statistics, and perform correlation analysis on the results. The grid interval has to be positive number greater than zero. (0.1, 1.0, 10 etc.). The second option allows the simulator to return data points where one or more of the species numbers changes by the specified sample size. In this case, the sample interval corresponds to integer jumps in species numbers, and is therefore an integer greater than or equal to 1. Setting the sample size equal to 1 returns raw data from the SSA algorithm implemented by the simulator, with the highest resolution that can be generated. Sample sizes higher than 1 correspond to averaged versions of this data.

Data using both options for a grid size of 1 and a sample size of 1 is shown below, for the first 10 time points. It can be seen that the grid-based data has equal time increments, while the sample based data does not. As a matter of fact, each run using sample based data can generate a different data length for the same start and end times.

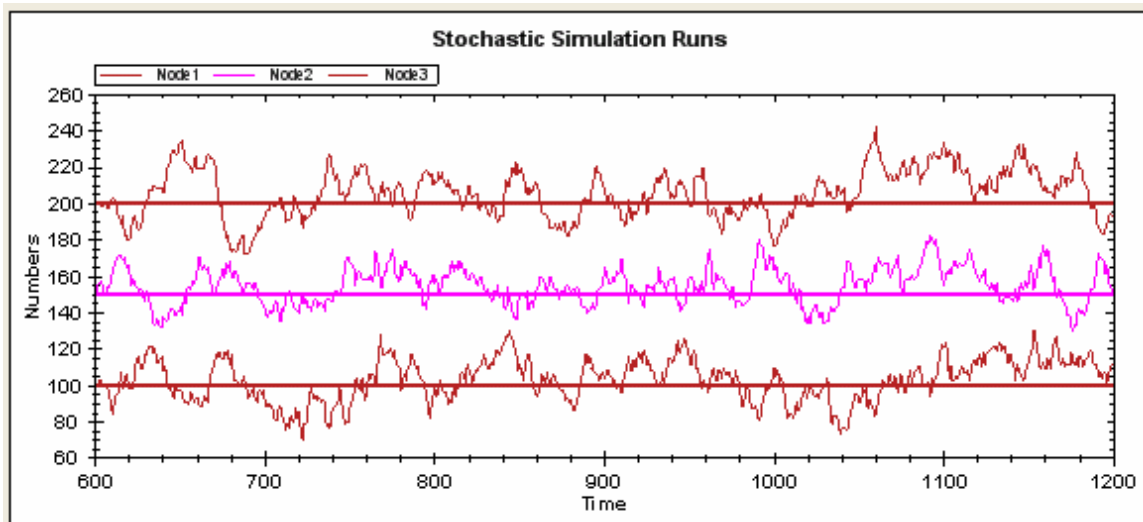
| Grid based data | | | | Sample based data | | | |
|-----------------|-------|-------|-------|-------------------|-------|-------|-------|
| Time | Node0 | Node1 | Node2 | Time | Node0 | Node1 | Node2 |
| 0.0 | 0 | 0 | 0 | 0.0 | 0 | 0 | 0 |
| 1.0 | 11 | 0 | 0 | 0.13224 | 1 | 0 | 0 |
| 2.0 | 19 | 1 | 0 | 0.25257 | 2 | 0 | 0 |
| 3.0 | 30 | 1 | 0 | 0.25498 | 3 | 0 | 0 |
| 4.0 | 39 | 2 | 0 | 0.39944 | 4 | 0 | 0 |
| 5.0 | 48 | 2 | 0 | 0.42714 | 5 | 0 | 0 |
| 6.0 | 59 | 5 | 0 | 0.44293 | 6 | 0 | 0 |
| 7.0 | 65 | 8 | 0 | 0.46249 | 7 | 0 | 0 |
| 8.0 | 72 | 11 | 0 | 0.53306 | 8 | 0 | 0 |
| 9.0 | 78 | 12 | 1 | 0.60575 | 9 | 0 | 0 |
| 10.0 | 89 | 16 | 1 | 0.71592 | 10 | 0 | 0 |

We restrict further analysis to grid-based data, due to the fact that each run has the same length. Future versions may use a modified interpolation algorithm to average sequences of varying lengths and time points to generate statistics with high resolution.

A useful feature that allows a model to be simulated multiple times makes it possible to use the species numbers at the end of one simulation be carried over to the next. If the first simulation has M runs (population), and the next simulation uses the same model with N runs, then the tool checks the following to specify initial conditions.

1. If the user presses the “Reset” button before initiating the next simulation, default initial conditions are assigned to all runs. These are values of the species that were originally specified in the SBML. Pressing the “Run Model” button after resetting thus starts a fresh run.
2. If the user presses the “Run Model” button after the end of the first simulation, without doing a reset, then the species numbers at the end of the simulation are assigned to the second simulation. Here, the tool performs a check on the new set of runs (N) and compares this to the old set (M).
 - a. If $M = N$, then all runs use the end-of-first simulation data as initial conditions for the second simulation.
 - b. If $M > N$, only the first N runs from the first simulation set are used to initialize the second simulation runs.
 - c. If $M < N$, the tool sets the first M sets using data from the first simulation sets, and the remaining $(N-M)$ sets are assigned default initial conditions. (perhaps a better alternative here is to assign an average of the M runs)

The consequence of this assignment of initial conditions is that the model can continue running from the previous state. The time axis is updated accordingly, as shown below.

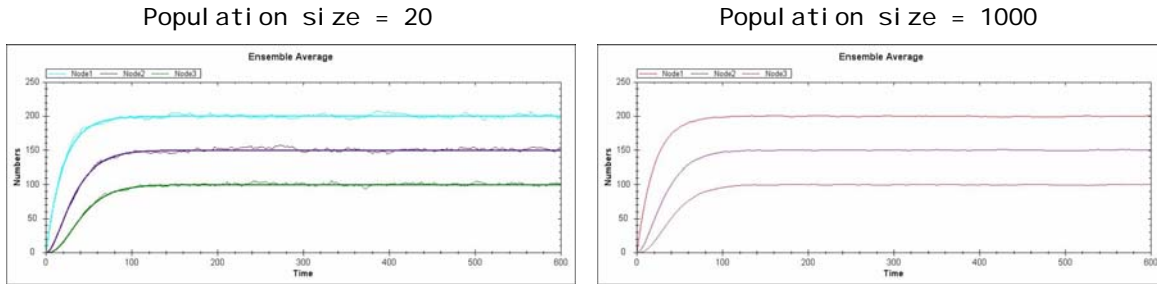


This is a continuation of the simulation from the previous plot which showed an overlay of all runs. The species numbers at the end of that run (from $t=0$ to $t=600$) were used to initiate the simulation for this run (from $t=600$ to $t=1200$). As can be seen, the model has reached steady-state, so the deterministic data lines are constant for all the species.

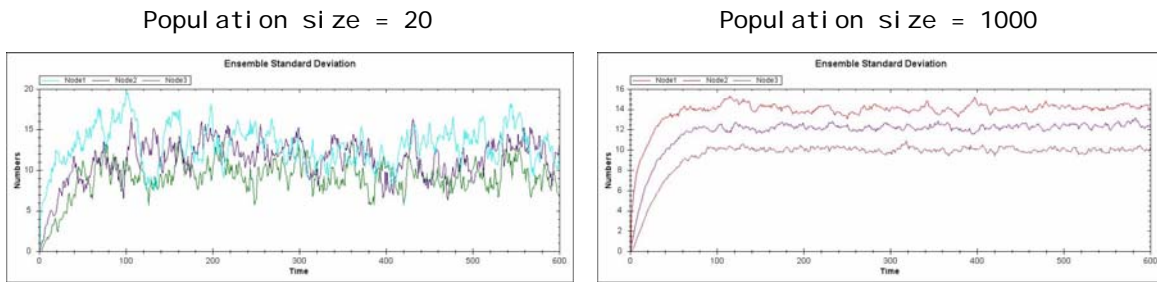
Ensemble Statistics

The ensemble statistics are computed if the data generated was obtained using a time-grid option. As mentioned earlier, the lengths of all runs are same in this case, which allows evaluation of population statistics. In particular, we are interested in seeing how the ensemble means and standard deviations behave for all the species. These are computed by averaging over all the population runs.

Here we show how the ensemble mean approaches the deterministic result for a small number of runs (20), and for a large number of runs (1000).



The figure below compares ensemble standard deviations for the same population sizes of 20 and 1000 respectively. It can be seen that the mean and the variance are roughly same, reflecting the underlying Poissonian sampling characteristic of the SSA algorithm.



Another useful statistical information corresponds to the coefficient of variation (CoV), which is given as the ratio of the mean to the standard deviation. The CoV is an indicator of the spread of the mean across the population, and is calculated as a single number (in this version) by averaging the mean and the standard deviation for all time (assuming steady-state has been reached). This is computed for each species, and is displayed at the bottom of the Ensemble averages tab, where users can see alongside the ensemble mean value, standard deviation and variance, along with the coefficient of variation. These are shown below for Node1, Node2 and Node3 respectively.

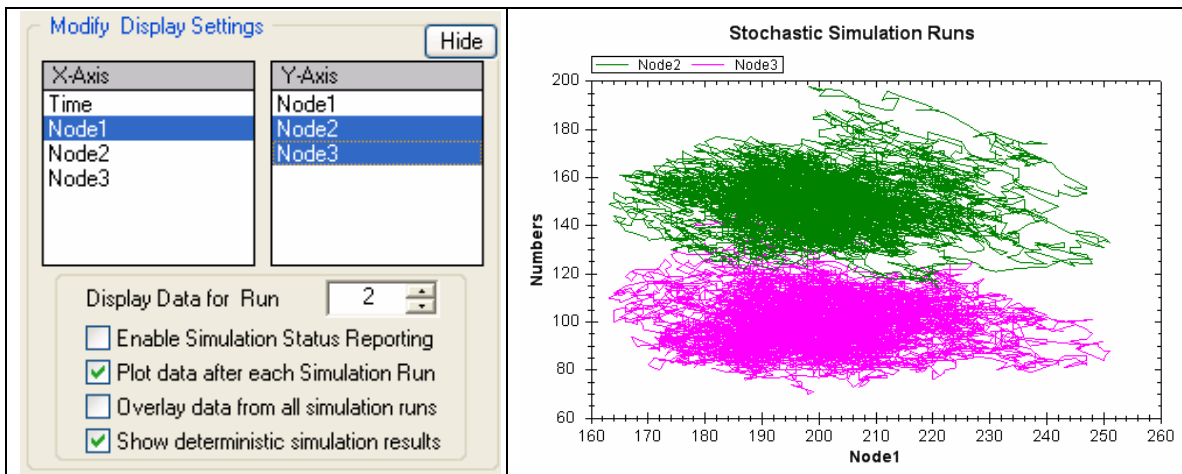
| | | | |
|-----------------|--------------------------|-------------------------|------------------------------|
| Mean = 200.3198 | Std. Deviation = 13.5917 | Variance = 184.73430889 | Coeff. of Variation = 6.79 % |
| Mean = 151.0678 | Std. Deviation = 11.5184 | Variance = 132.67353856 | Coeff. of Variation = 7.63 % |
| Mean = 100.8381 | Std. Deviation = 9.3347 | Variance = 87.13662409 | Coeff. of Variation = 9.26 % |

It would be useful to plot the coefficient of variation for the entire time period, as it would show variations in the statistics when other factors such as regulation come into play. This could be implemented in the future versions of this application.

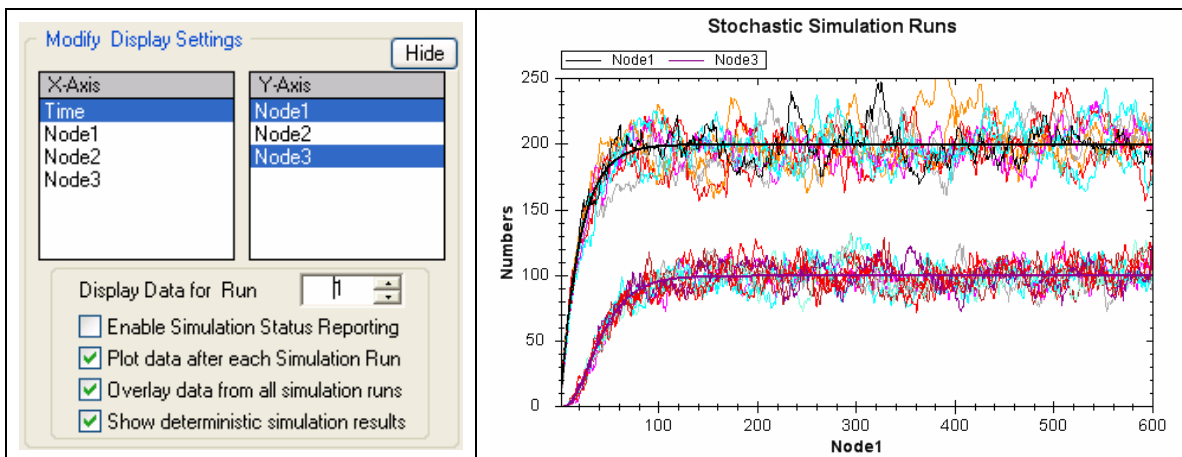
Display Panel Properties

The display panel has a collection of options which are described below. The panel appears expanded when the user loads a new model, either through JDesigner, or another SBW enabled application, or via the loadSBML button. The panel has two parts, the first where one or more species can be selected to be plotted (against time or against each other), and a second part, where the species being plotted can be further controlled.

This panel is shown here, in the phase plot mode, where Node2 and Node3 are plotted against Node1 on the same plot. (Here the same model was simulated from $t=0$ to $t=2000$, with a grid-interval of 0.2) Note that the phase plot ignores the transient data (if the user decides to ignore it for analysis as described earlier)

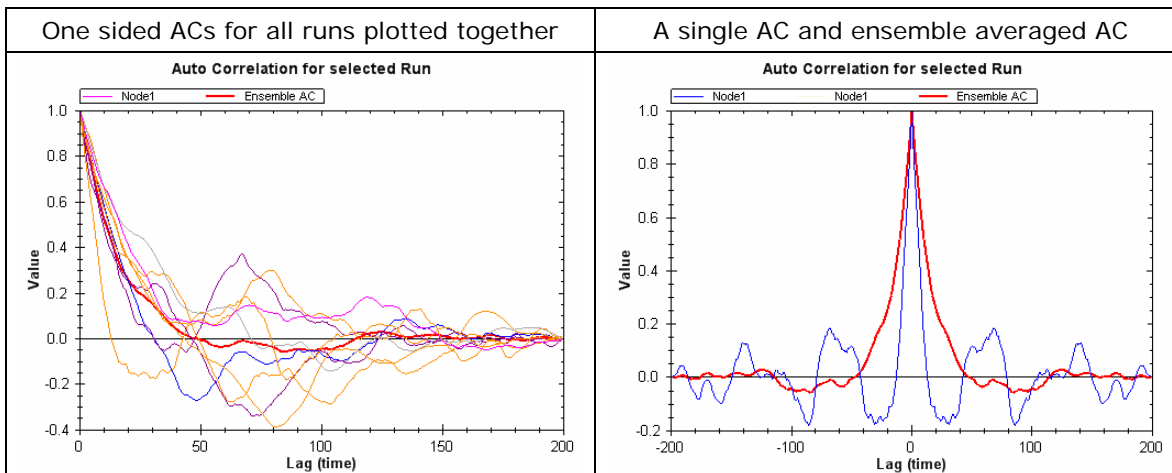


The data from all the runs can be plotted together, using the Overlay data checkbox. This allows data from all runs to be easily compared, as shown below for Node1 and Node3



A note of caution here is that this option should be avoided if the number of runs to be plotted is more than 20 or 25, as the memory used to retrieve and overlay all the data on the plot can freeze up the system for a considerable amount of time.

This option can also be used to overlay the auto-correlations for all the runs for one or more species. This is especially useful when one-sided auto-correlation are plotted. One such example is shown below where the one-sided autocorrelations from all runs have been plotted together for Node1 in the left figure, showing how the variation in the data is reflected in the auto-correlations.



The thick red line in the above figures corresponds to the ensemble averaged auto-correlation, shown with a single run auto-correlation in the two-sided form in the right figure. The appearance of both these plots can be controlled by using the additional options panel at the bottom of the correlations tab, shown below.

Auto-Correlations Show One-Sided Ensemble Averaged PSD Auto-Refine Use dB

This panel also includes controls for displaying the Power Spectral Density, which will be described in an upcoming section.

The display panel also includes a checkbox option to control the status reporting from the simulator. This option allows users to receive information on the progress of each run from the simulator by means of the progress-bar at the bottom of the user interface, which also includes other details such as a status label that is updated through the simulation stages, as well as information on the model that is currently loaded into the application. Enabling the status reporting checkbox is useful when running simulations that involve long data runs. The simulations would run much faster with this option turned off (as the simulator does not have to continuously report progress), which indeed is the default for this control. This is shown below.

Data generation completed after 10 runs. LinearChain : 3 Floating species, 4 Reactions and 2 Boundary species.

Sections to be added

Power Spectral Density
Transfer Functions

Noise Injection into Boundary Nodes
SBML Modification

Summary and Conclusions