

SBvar: Varying Parameters Analysis for System Biology

Functional Specification

Rachel Ng

BIOEN 537 Software Project

<https://github.com/racng/SBvar>

Background

Biochemical systems can be modelled mathematically and simulated based on knowledge of reaction networks that convert reactants to products. The dynamic concentration of these floating species can be described by a system of differential equations with given starting conditions and kinetic constants. The properties of such a system, such as its steady state concentration, fluxes, and dynamic behavior, may change to various degrees based on the starting concentration of the floating species and kinetic constants. It is valuable to biologists to characterize the sensitivity of the system to changes in these experimental inputs and assumptions about reaction kinetics. Computational biologists can use this analysis to better design biological systems and choose the range of experimental conditions that would achieve their desired biochemical production.

System/Solution Overview

SBvar is a Python package for performing various robustness analysis to characterize how the dynamics of a biological system change in response to variation in parameters such as species starting concentration, kinetic constants, and reaction knock out. The response of the system across ranges of parameters will be visualized using line plot or contour plots. This tool is designed to be imported into Python script and therefore requires programmatic interaction to use. It requires user input of a biological reaction network in a format (ex. SBML, Antimony) readable by the simulation software Tellurium. Visualization will be performed using Python packages such as matplotlib.

User profile

Computational biologist - a researcher with background in both biology and programming skills in Python. The user would know how to convert their biological network of choice into a usable input format for tellurium in Antimony format. This requires understanding of syntax and potentially debugging skills. In addition, the user would know how to import this package in a Python environment and call basic functions.

Use Cases

Use Case 1

Computational biologist desires to assess the sensitivity of steady state to varying parameters. The user can choose to vary one or two factors at a time and specify the range of levels (continuous or discrete) for each factor. The factors that can be changed are the following:

- Starting concentration of floating species
- Values of reaction rate constants

The response that the user can visualize are the following:

- Steady state concentration

- Steady state reaction rates

First, the user initializes an object with an Antimony model, factor(s), and range of levels. The user can call the object's function to simulate the system for each possible experiment (level or combinations of levels). The program will determine if steady state is reached and store the resulting steady state concentrations/fluxes. After simulation, the steady state concentrations/fluxes of any user-specified species can be visualized using a line plot to visualize the effect of varying one parameter on any floating specie's concentration. For analysis that varies two factors, a contour plot will visualize the response as color across a 2D grid. These plots can be saved as images and the steady state concentrations/fluxes across all simulation experiments can be saved as text files.

Use Case 2

Computational biologist desires to characterize the sensitivity of time courses to varying parameters (starting concentration and/or rate constants). Similar to Use Case 1, the user initializes an object defined by an Antimony model, factor(s), and ranges of levels for 1-dimensional or 2-dimensional parameter scan. However, this analysis assesses the shape of the entire time course instead of just the steady state. After simulating all possible levels, the user can ask the program for a 3D visualization of time courses of concentration/flux of a particular species across levels for varying one factor. These metrics can be visualized using line plot or contour plots for 1-way or 2-way design respectively. Finally, the time courses (flux/concentration) of a species can be clustered using various time series clustering algorithms such as k-means, k-shape, and dbscan. The classification of time series can be visualized using lineplot and timeseries contour for 1-way design, and 2D plots for 2-way design. Additionally functions for analyzing time series such as finding maximum can be incorporated in the future.

Use Case 3 (Not Yet Implemented)

Computational biologist desires to assess the sensitivity of oscillations to varying parameters (starting concentration and/or rate constants). Similar to Use Case 1, the user initializes an object defined by an Antimony model, factor(s), and ranges of levels for 1-dimensional or 2-dimensional parameter scan. The user calls a function for simulation across all possible experiments and quantitative characterization of oscillations. The program would calculate metrics describing the oscillations of floating species such as frequency, wavelength, amplitude, and time delay. The response of these oscillation metrics can be visualized using line plot or contour plots for 1-way or 2-way design respectively.