Bioinformatics, 32(5), 2016, 798–800
doi: 10.1093/bioinformatics/btv655
Advance Access Publication Date: 9 November 2015
Applications Note

Systems biology

BioNetFit: a fitting tool compatible with BioNetGen, NFsim and distributed computing environments

Brandon R. Thomas[1], Lily A. Chylek[2,1], Joshua Colvin[1], Suman Sirimulla[3], Andrew H.A. Clayton[4], William S. Hlavacek[5,*] and Richard G. Posner[1,*]

[1]Department of Biological Sciences, Northern Arizona University, Flagstaff, AZ, USA, [2]Department of Chemistry and Chemical Biology, Cornell University, Ithaca, NY, USA, [3]Department of Basic Sciences, Saint Louis College of Pharmacy, Saint Louis, MO, USA, [4]Center for Microphotonics, Faculty of Science, Engineering and Technology, Cell Biophysics Laboratory, Swinburne University of Technology, Hawthorn, VIC, Australia and [5]Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM, USA

*To whom correspondence should be addressed.

Associate Editor: Jonathan Wren

Received on June 24, 2015; revised on October 20, 2015; accepted on November 3, 2015

Abstract

Summary: Rule-based models are analyzed with specialized simulators, such as those provided by the BioNetGen and NFsim open-source software packages. Here, we present BioNetFit, a general-purpose fitting tool that is compatible with BioNetGen and NFsim. BioNetFit is designed to take advantage of distributed computing resources. This feature facilitates fitting (i.e. optimization of parameter values for consistency with data) when simulations are computationally expensive.

Availability and implementation: BioNetFit can be used on stand-alone Mac, Windows/Cygwin, and Linux platforms and on Linux-based clusters running SLURM, Torque/PBS, or SGE. The BioNetFit source code (Perl) is freely available (http://bionetfit.nau.edu).

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Contact: bionetgen.help@gmail.com

1 Introduction

Biomolecular interactions can be represented by formalized rules (Chylek *et al.*, 2014a; Stefan *et al.*, 2014). Collections of rules form rule-based models, which provide concise representations of biomolecular interaction networks and can be analyzed to obtain insights into how system-level behavior emerges from biomolecular interactions (Chylek *et al.*, 2014a; Stefan *et al.*, 2014). Rule-based models must be analyzed with specialized algorithms and software tools (Chylek *et al.*, 2014a; Stefan *et al.*, 2014), such as BioNetGen (Harris *et al.*, 2015), which interprets models encoded in the BioNetGen language (BNGL) and provides deterministic, stochastic and hybrid forward simulation capabilities (Harris *et al.*, 2015). To date, other critical methods of analysis, such as fitting (parameter estimation), have typically been applied *ad hoc* (e.g. by writing problem-specific programs), as in the study of Kozer *et al.* (2013) or Chylek *et al.* (2014b), which leads to duplication of effort and hinders the ability to reproduce results.

Although many algorithms and software implementations are available for analysis of models specified in traditional forms (Press *et al.*, 2007), such as that of ordinary differential equations (ODEs), standard techniques are not easily applied to rule-based models except for those rule-based models that can be recast into a traditional model form. Translating a set of rules into a reaction network (i.e. a list of reactions) or the corresponding ODEs for the mass-action kinetics of the network, a capability provided by BioNetGen, is expensive and sometimes impracticable. In such cases, direct simulation methods must be used (Chylek *et al.*, 2014a). These methods have been called network-free methods, because they do not involve processing a set of rules to obtain a list of reactions. Rather, rules are used as reaction-event generators within a particle-based kinetic Monte Carlo (KMC) simulation. Network-free simulators, such as NFsim (Sneddon *et al.*, 2011), which is compatible with BNGL-specified models, expand the range of biomolecular interaction networks that can be studied, because the cost of network-free simulation scales with the number of rules in a model, not the number of reactions implied by the rules (Danos *et al.*, 2007). Software that provides a general-purpose fitting capability and that interfaces with a network-free simulator has not hitherto been available.

Here, we present BioNetFit, a general-purpose fitting program for rule-based models that is compatible with BioNetGen and NFsim. Because of the potentially high cost of network-free simulation (Chylek *et al.*, 2014a), BioNetFit has been designed to be used on clusters. It can also be used on stand-alone computers. BioNetFit implements a genetic algorithm (Whitley, 1994; Smith and Eiben, 2008). Below, we provide an overview of BioNetFit and a demonstration of its capabilities.

2 Methods

Source code and a user manual, which includes installation instructions, are available at the BioNetFit Web site (http://bionetfit.nau.edu). Dependencies are described in the user manual and in Supplementary Methods. Collections of files needed to run example fitting jobs are included in the BioNetFit distribution in the 'examples' directory and six subdirectories named 'example1' through 'example6.' Step-by-step instructions for running these fitting jobs are included in the user manual. The fitting procedure implemented in BioNetFit is described in Supplementary Methods.

Running BioNetFit requires several problem-specific plain-text files: a BNGL model-specification (.bngl) file (i.e. a BioNetGen/NFsim input file); one or more data (.exp) files, each with two or more white space-separated columns; and a BioNetFit configuration (.conf) file (see the 'examples' directory in the BioNetFit distribution). If NFsim is being used, a .rnf file may also be required. To run BioNetFit, a user must identify the parameters that are free to vary in fitting in the .conf file and include a simulation command in the .bngl file for each .exp file. Each simulation command should generate outputs corresponding to the data in the associated .exp file. See the BioNetFit user guide and Supplementary Methods for additional information.

Demonstration results were obtained by running BioNetFit on the Monsoon cluster at Northern Arizona University (http://nau.edu/hpc/). The .bngl, .exp and .conf files required to repeat the demonstration results presented in the Results section are provided as Supplementary Files S1–S6. These files are also provided in the 'example1' and 'example2' subdirectories of the 'examples' directory within the BioNetFit distribution.

3 Results

To provide a demonstration of capabilities, we used BioNetFit together with BioNetGen, NFsim and experimental data of Kozer *et al.* (2013) to estimate parameter values for a published (Kozer *et al.*, 2013) and an extended version of a model for activation of the epidermal growth factor (EGF) receptor (EGFR). Each version of the model accounts for EGF-induced oligomerization of EGFR. The published model is defined in Supplementary File S1, which is a .bngl file that directs BioNetGen's ODE solver to produce simulation results. The file is derived from the .bngl file used by Kozer *et al.* (2013). This version of the model limits EGFR oligomers to a maximum size of four receptors per oligomer. This limitation was introduced by Kozer *et al.* (2013) to avoid the need to use computationally expensive network-free simulation in fitting, which was performed using problem-specific code. The extended model is defined in Supplementary File S2, which is a .bngl file that directs NFsim to produce simulation results. The file is derived from the .bngl file used by Kozer *et al.* (2014). This version of the model imposes no constraints on the size of linear oligomers. It is otherwise identical to the published model of Kozer *et al.* (2013). The steady-state dose-response and time-series data used in each of the two fitting runs are provided in Supplementary Files S3 and S4, which are .exp files. Supplementary Files S5 and S6 are the BioNetFit configuration (.conf) files for the two fitting runs. These files complement Supplementary Files S1 and S2, respectively.

In the fitting procedure, BioNetFit evaluated the consistency of simulation results with all available data (i.e. BioNetFit found a global fit). The chi-square metric was used to determine goodness of fit. Parameter estimates are summarized in Supplementary Table S1. The number of free parameters was nine, which included five physical parameters (e.g. rate constants) and four scaling factors (parameters of a measurement model), which define linear relationships between measured quantities and model-predicted quantities. The quality of the fit for the original and extended models is illustrated in Figure 1. For best-fit parameter values, the extended model predicts that EGFR oligomers larger than tetramers do not form appreciably (Supplementary Fig. S1). This finding justifies the omission of these oligomers in the analysis of Kozer *et al.* (2013).

BioNetFit's ability to take advantage of simulation runs performed in parallel on a cluster enabled consideration of the extended model, which had to be simulated by calling NFsim. The KMC (stochastic simulation) method implemented in NFsim tends to be computationally expensive, because in such methods, system state is advanced one reaction event at a time (Chylek *et al.*, 2014a). Use of NFsim was required because the extended model places no constraint on the size of (linear) EGFR oligomers, which form via polymerization-like reactions. Network-free simulation is typically required to cope with polymerization-reactions (Chylek *et al.*, 2014a). As expected, we found that simulations performed by NFsim (for the extended model) were significantly more expensive than simulations performed by BioNetGen's ODE solver (for the original model). The cost difference depends on various factors, including hardware, model parameter values and algorithmic parameter settings. For a representative CPU in the cluster we used and the parameter settings of Supplementary Files S1 and S2, an NFsim simulation is ~5-fold more expensive than an ODE-based simulation. It should be noted that the cost difference depends sensitively on the value of one particular algorithmic parameter called f in Supplementary File S2, which scales system size. In general, the cost of a discrete-event stochastic simulation, the type of simulation performed by NFsim, depends on the number of reaction events per unit time, which in turn depends on system size (measured by the number of reactive molecules in the system). In Supplementary File S2, we set $f = 0.01$, which corresponds to a system size of 1% of a cell. This setting reduces the cost of individual simulation runs (and memory requirements) at the expense of introducing noise (i.e. fluctuations arising from probabilistic reaction events). The noisiness of simulation results complicates the comparison of predicted and measured (population averaged) quantities. Because of this issue, in fitting, multiple NFsim simulations were performed for each

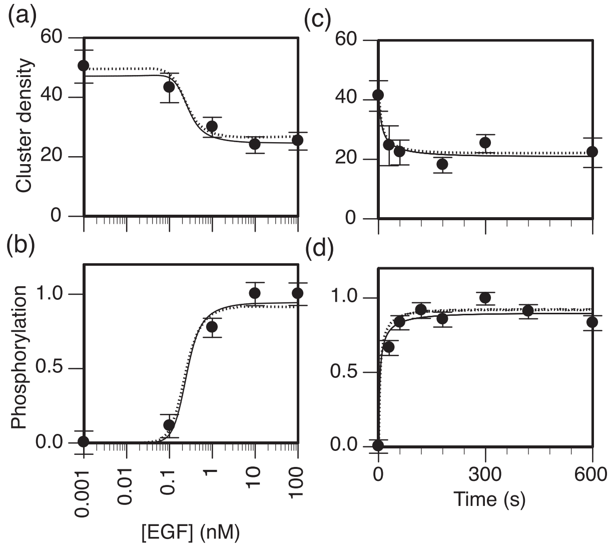

Fig. 1. Plots illustrating the quality of fits found by BioNetFit. Together with Supplementary Files S1 and S3–S5 and BioNetGen's ODE solver, BioNetFit was used to estimate values for the free parameters of the model of Kozer *et al.* (2013) for EGFR activation. Together with Supplementary Files S2–S4 and S6 and NFsim, BioNetFit was also used to estimate values for the corresponding parameters of an extended model, which can only be analyzed using network-free simulation. In each case, parameter estimates are based on the data of Kozer *et al.* (2013), which are represented by points with error bars. The left panels, (a) and (b), show fits to the steady-state dose-response data. The right panels, (c) and (d), show fits to the time-series data. Solid curves show deterministic simulation results produced by BioNetGen's ODE solver. Dotted curves show stochastic simulation results produced by NFsim. Curves are based on the best-fit parameter values, which are summarized in Supplementary Table S1. The stochastic simulation results shown here represent averages from multiple simulation runs. In fitting, each stochastic simulation run was repeated and the results averaged to obtain smoothed time courses. The CPU time typically required for a single stochastic simulation was ~3 min. The fitting procedure for the extended model involved a total of ~10 000 (stochastic) simulations. Because of parallel processing, the procedure took only ~40 h. The number of EGFR clusters (the sum of EGFR monomers, dimers, etc.) per μm^2 is indicated on the vertical axis of panel (a) and (c). The relative level of EGFR phosphorylation (arbitrary units) is indicated on the vertical axis of panel (b) and (d)

goodness-of-fit evaluation. BioNetFit allows a user to cope with the noisiness of stochastic simulation results in a brute-force manner by repeating simulation runs a user-specified number of times and then averaging the results. This procedure serves to smooth stochastic simulation results.

Results from two additional non-trivial fitting demonstrations are summarized in Supplementary Tables S2 and S3, which provide best-fit parameter estimates, and Supplementary Figures S2 and S3, which illustrate the quality of fits obtained. The files needed to reproduce these results are provided in the 'example3' and 'example4' subdirectories of the 'examples' directory in the BioNetFit distribution. The demonstrations each require NFsim. The 'example3' demonstration concerns a rule-based model for interaction of a trivalent ligand with a bivalent cell-surface receptor and parameter estimation on the basis of equilibrium data collected via flow cytometry (Monine *et al.*, 2010). The 'example4' demonstration concerns a rule-based model for early events in T-cell receptor signaling and parameter estimation on the basis of temporal phosphoproteomic data collected via mass spectrometry-based proteomics (Chylek *et al.*, 2014b). Notably, because changes in phosphorylation levels were measured relative to basal phosphorylation levels, this fitting problem required normalization of model outputs at the basal steady state, which is explained in the BioNetFit user manual.

4 Conclusions

Parameter estimation is a critical step in model building and model-guided analysis of data. BioNetFit provides a fitting capability that is suitable for a broad range of rule-based modeling applications because of its compatibility with BioNetGen and NFsim. Moreover, BioNetFit can potentially be used with traditionally formulated models (e.g. ODE models), because BioNetGen has the capability to translate models encoded in the SBML format (Hucka *et al.*, 2003) into BNGL format (Harris *et al.*, 2015). Importantly, BioNetFit is designed to take advantage of distributed computing resources, which can make fitting feasible even when simulation runs are computationally expensive.

Funding

This work was supported by NIH/NIGMS grant R01GM111510. The Moonson cluster at Northern Arizona University is supported by Arizona's Technology and Research Initiative Fund.

Acknowledgements

We thank the High Performance Computing support staff at Northern Arizona University for assistance in using the Monsoon cluster. We thank the New Mexico Consortium for providing temporary office space to LAC.

Conflict of Interest: none declared.

References

Chylek,L.A. *et al.* (2014a) Rule-based modeling: a computational approach for studying biomolecular site dynamics in cell signaling systems. *Wiley Interdiscip. Rev. Syst. Biol. Med.*, **6**, 13–36.

Chylek,L.A. *et al.* (2014b) Phosphorylation site dynamics of early T-cell receptor signaling. *Plos One*, **9**, e104240.

Danos,V. *et al.* (2007) Scalable simulation of cellular signaling networks. *Lect. Notes Comput. Sci.*, **4807**, 139–157.

Harris,L.A. *et al.* (2015) BioNetGen 2.2: advances in rule-based modeling. *Bioinformatics* http://arxiv.org/pdf/1507.03572.pdf

Hucka,M. *et al.* (2003) The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics*, **19**, 524–531.

Kozer,N. *et al.* (2013) Exploring higher-order oligomerisation and phosphorylation—a combined experimental and theoretical approach. *Mol. BioSyst.*, **9**, 1849–1863.

Kozer,N. *et al.* (2014) Recruitment of the adaptor protein Grb2 to EGFR tetramers. *Biochemistry*, **53**, 2594–2604.

Monine,M.I. *et al.* (2010) Modeling multivalent ligand-receptor interactions with steric constraints on configurations of cell-surface receptor aggregates. *Biophys. J.*, **98**, 48–56.

Press,W.H. *et al.* (2007) *Numerical Recipes: The Art of Scientific Computing*, 3rd Edition. Cambridge University Press, Cambridge, UK.

Smith,J.E. and Eiben,A.E. (2008) *Introduction to Evolutionary Computing*. Springer, Berlin.

Sneddon,M.W. *et al.* (2011) Efficient modeling, simulation and coarse-grain ing of biological complexity with NFsim. *Nat. Methods*, **8**, 177–183.

Stefan,M.I. *et al.* (2014) Multi-state modeling of biomolecules. *PLOS Comput. Biol.*, **10**, e1003844.

Whitley,D. (1994) A genetic algorithm tutorial *Stat. Comput.*, **4**, 65–85.