BIOINFORMATICS APPLICATIONS NOTE Vol. 23 no. 14 2007, pages 1857–1858
doi:10.1093/bioinformatics/btm237

Systems Biology

Genetdes: automatic design of transcriptional networks

Guillermo Rodrigo[1], Javier Carrera[1] and Alfonso Jaramillo[2,*]

[1]Dep. de Matematica Aplicada, Universidad Politecnica de Valencia, Camino de Vera s/n, 46022 Valencia, Spain and [2]Lab. de Biochimie, CNRS - UMR 7654, Ecole Polytechnique, Route de Saclay, 91128 Palaiseau, France

Received on February 28, 2007; revised on April 22, 2007; accepted on April 26, 2007

Advance Access publication May 7, 2007

Associate Editor: Alfonso Valencia

ABSTRACT

Motivation: The rational design of biological networks with prescribed functions is limited to gene circuits of a few genes. Larger networks involve complex interactions with many parameters and the use of automated computational tools can be very valuable. We propose a new tool to design transcriptional networks with targeted behavior that could be used to better understand the design principles of genetic circuits.

Results: We have implemented a Simulated Annealing optimization algorithm that explores throughout the space of transcription networks to obtain a specific behavior. The software outputs a transcriptional network with all the corresponding kinetic parameters in SBML format. We provide examples of transcriptional circuits with logical and oscillatory behaviors. Our tool can also be applied to design networks with multiple external input and output genes.

Availability: The software, a tutorial manual, parameter sets and examples are freely available at http://synth-bio.yi.org/genetdes.html

Contact: Alfonso.Jaramillo@polytechnique.edu

1 INTRODUCTION

Our understanding of transcriptional networks is very incomplete, although it has considerably increased over the last years (Babu and Teichmann, 2003). New developments in synthetic biology (Endy, 2005) aim to the design of genetic networks with targeted behavior. The experimental use of combinatorial techniques has allowed the design of 3-gene synthetic transcriptional networks having NAND, NOR and NOT behaviors using a library of five possible promoters (Guet *et al.*, 2002). Recent rational design of synthetic networks took advantage of simulation techniques to obtain a set of valid kinetic parameters (Atkinson *et al.*, 2003; Elowitz and Leibler, 2000). Nevertheless, the computer simulation is mainly focused on the computation of the corresponding dynamics for a given choice of kinetic constants and network topology. The use of automated tools can provide the means to optimize the kinetic parameters (Feng *et al.*, 2004) or even to optimize a protein network topology (Chickarmane *et al.*, 2005; Francois and Hakim, 2004) to obtain a desired dynamics. We have developed a new bioinformatics tool that finds a transcriptional circuit with a desired dynamical behavior. We can use this tool not only to design synthetic transcriptional networks, but also to complete the topology (or the parameters) of a given natural network by assuming a given dynamics.

*To whom correspondence should be addressed.

2 APPROACH

The dynamics of a genetic system can be described by nonlinear and coupled ordinary differential equations. For simplicity, we have just considered transcriptional regulations together with an effective model of protein concentrations. The dynamics of a transcription factor concentration (Y_i) is given by the following differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t}[Y_i] = \alpha_i R_i - \beta_i [Y_i] + \gamma_i, \quad (1)$$

where α_i is the transcription-translation rate of gene i, β_i the corresponding degradation rate, and γ_i the basal rate. The function R_i defines the regulatory factor for the promoter of gene i and it is specified by the following expansion

$$R_i = c_{00} + c_{10} R_{iA} + c_{01} R_{iB} + c_{11} R_{iA} R_{iB}, \quad (2)$$

where c_{pq} are a set of coefficients that determines a particular promoter behavior. R_{iA} and R_{iB} are defined by $R^r_{iA} = 1/(1 + ([Y_A]/K_{iA})^{n_{iA}})$, when the promoter contains a repression site, and by $R^a_{iA} = (Y_A/K_{iA})^{n_{iA}}/(1 + ([Y_A]/K_{iA})^{n_{iA}})$, when it contains an activation site. K_{iA} is the regulatory coefficient and n_{iA} is the Hill coefficient. We have only considered the c_{pq} corresponding to a small library of promoters, although we could have used other choices for c_{pq}. We have selected promoters able to implement digital behaviors, such as the logic gates YES, NOT, AND, OR, NAND, NOR, XOR and XNOR, as shown in (Bintu *et al.*, 2005; Buchler *et al.*, 2003). The parameters c_{pq} could also be fitted against experimental data, which would allow us to enlarge this library. In addition, we could also have considered promoters regulated by more than two transcription factors. To compute the objective function, we define $z(t)$ as the targeted dynamics, and $y(t)$ as the dynamics of the corresponding circuit to evaluate,

$$J = \int_{t_i}^{t_f} |y - z| \chi \mathrm{d}t, \quad (3)$$

where χ is a weighting factor used to only compute a region of interest (e.g. to avoid transients or to impose an oscillatory dynamics). When we want to design a transcriptional

Downloaded from http://bioinformatics.oxfordjournals.org/ by guest on June 1, 2013

© The Author 2007. Published by Oxford University Press. All rights reserved. For Permissions, please email: journals.permissions@oxfordjournals.org

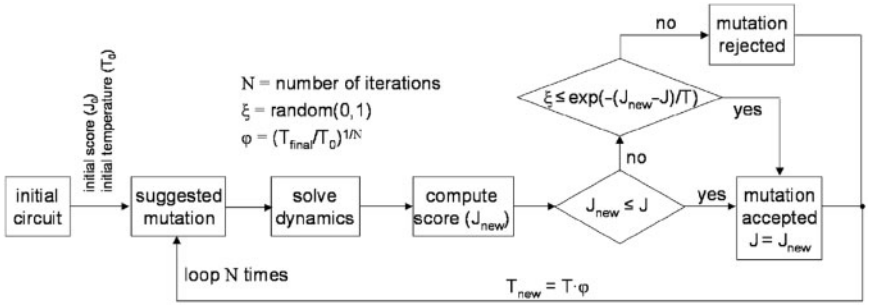

Fig. 1. Scheme of our algorithm to design transcriptional networks based on Simulated Annealing. We start from an initial circuit (usually random) and compute its corresponding score after solving its dynamics. Then we suggest a random modification (mutation step) of either a kinetic parameter or a network interaction. Each generated network is simulated and its fitness (J_{new}) is computed. We accept or reject the mutation according to the Metropolis criterion and we go back to the mutation step (we repeat this loop N times).

network having a targeted logical function, we add the score corresponding to each possible entry of the truth table.

We use a Monte Carlo Simulated Annealing (Kirkpatrick *et al.*, 1983) method to perform the optimization in the space of all possible transcriptional networks constructed with our promoter library (Fig. 1). During this process, we suggest a modification of the network by either changing a kinetic parameter or varying its topology. We simulate each new circuit and compute its objective function (J_{new}). If the suggested mutation lowers J ($J_{new} \leq J$), we accept it. Otherwise, we accept it with probability $exp(-(J_{new}-J)/T)$, where T is a temperature parameter that decreases exponentially with the number of iterations. Afterwards, we loop back and suggest a new mutation.

During the suggestion step we consider five types of mutations: (i) change one kinetic parameter in the model, (ii) add a new regulation (we randomly chose between activation or repression) between two genes, (iii) remove a regulation, (iv) add a new gene (which will have a constitutive promoter), and (v) remove a gene. We remove a gene if it does not regulate any other. In order to better explore each network structure, the probability to change a kinetic parameter is taken much bigger than a topology change.

3 IMPLEMENTATION

Genetdes has been developed in C and it uses the CVODE solver from the SUNDIALS library (Hindmarsh *et al.*, 2005) to compute the dynamics. Our application is easily compiled and executed in Linux environments. The user can adjust the number of Monte Carlo iterations depending on the complexity of the problem. The design of networks of a few genes requires several minutes of CPU time to get an optimal solution. Importantly, Genetdes outputs the designed transcriptional networks in SBML format (Hucka *et al.*, 2003), which can be postprocessed with other software, and it can input an initial network in SBML format too.

To exemplify the use, we have targeted digital and oscillatory behaviors (examples are provided in our web site). In Fig. 2 we illustrate an optimal genetic network implementing an AND logic gate with three transcription factors. Genetdes allows the user to select the number of input and output genes to design networks with higher complexity. The user may choose to only optimize the kinetic parameters of a given network.

[u1] uM	[u2] uM	[y] uM
0	0	0
0	1	0
1	0	0
1	1	1

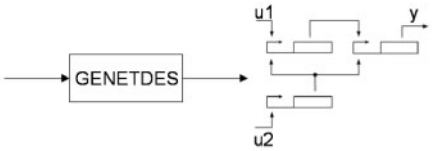

Fig. 2. To design a transcriptional network with a given logic behavior (e.g. AND gate), we introduce the corresponding truth table as input and execute Genetdes to obtain an SBML file for the optimized circuit.

ACKNOWLEDGEMENTS

Work supported by EU grant FP6-NEST 043340.

Conflict of Interest: none declared.

REFERENCES

Atkinson,M.R. *et al.* (2003) Development of genetic circuit exhibiting toggle switch or oscillatory behavior in *Escherichia Coli*. *Cell*, **113**, 597–607.

Babu,M.M. and Teichmann,SA (2003) Evolution of transcription factors and the gene regulatory network in Escherichia coli. *Nucleic Acids Res.*, **31**, 1234–1244.

Bintu,L. *et al.* (2005) Transcriptional regulation by the numbers: applications. *Curr. Opin. Genet. & Dev.*, **15**, 125–135.

Buchler,N.E. *et al.* (2003) On schemes of combinatorial transcription logic. *Proc. Natl. Acad. Sci. USA*, **100**, 5136–5141.

Chickarmane,V. *et al.* (2005) Bifurcation discovery tool. *Bioinformatics*, **21**, 3688–3690.

Elowitz,M.B. and Leibler,S. (2000) A synthetic oscillatory network of transcriptional regulators. *Nature*, **403**, 335–338.

Endy,D. (2005) Foundations for engineering biology. *Nature*, **438**, 449–453.

Feng,X.J. *et al.* (2004) Optimizing genetic circuits by global sensitivity analysis. *Biophys. J.*, **87**, 2195–2202.

Francois,P. and Hakim,V. (2004) Design of genetic networks with specified functions by evolution *in silico*. *Proc. Natl. Acad. Sci. USA*, **101**, 580–585.

Guet,C.C., Elowitz,MB, Hsing,W. and Leibler,S. (2002) Combinatorial synthesis of genetic networks. *Science*, **296**, 1466–1470.

Hindmarsh,A.C. *et al.* (2005) SUNDIALS: Suite of Nonlinear and Differential/Algebraic Equation Solvers. *ACM Trans. Math. Software*, **31**, 363–396.

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.

Kirkpatrick,S. *et al.* (1983) Optimization by simulated anne-aling. *Science*, **220**, 671–680.

Downloaded from http://bioinformatics.oxfordjournals.org/ by guest on June 1, 2013