RESEARCH ARTICLE

Asmparts: assembly of biological model parts

**Guillermo Rodrigo · Javier Carrera ·
Alfonso Jaramillo**

Abstract We propose a new computational tool to produce models of biological systems by assembling models from biological parts. Our software not only takes advantage of modularity, but it also enforces standardisation in part characterisation by considering a model of each part. We have used model parts in SBML to design transcriptional networks. Our software is open source, it works in linux and windows platforms, and it could be used to automatically produce models in a server. Our tool not only facilitates model design, but it will also help to promote the establishment of a registry of model parts.

Abbreviations
SBMP   Standard biological model
TF   Transcription factor
POPS   Polymerase per second
RIPS   Ribosome per second
RBS   Ribosome binding site

G. Rodrigo · J. Carrera
IBMCP, CSIC-UPV, Camino de Vera s/n, 46022 Valencia,
Spain

A. Jaramillo (✉)
Laboratoire de Biochimie, Ecole Polytechnique, CNRS, 91128
Palaiseau Cedex, France
e-mail: Alfonso.Jaramillo@polytechnique.edu

Introduction

The design of biological systems significantly benefits from the feedback gained through model simulation. For instance, the use of models based on ordinary differential equations (ODEs) in Atkinson et al. (2003) allowed to characterise the required conditions for an oscillatory behaviour. This modelling requires the use of kinetic parameters for each reaction, which are not available in most cases. The simulation of complex biological systems very often requires the use of ODEs with effective models based on protein concentrations.

The modelling of natural biological systems often use in vitro data that may not properly describe the dynamics of the cell (Teusink et al. 2000) or break modularity. There has also been some attempts to use in vivo data to fit the kinetic models (Visser and Heijnen 2003; Segre et al. 2003). The use of in vivo characterizations of real and synthetic systems, such as in Cox et al. (2007), will serve to generate a large library of part models that will end up integrated in artificial circuits. On the other hand, synthetic biological systems are expected to be designed by assembling well-characterised modules (Hartwell et al. 1999), with the probable limitation of discarding those designs where the crosstalk among the modules is unbearable. In addition, the context (e.g., the chassis or the environmental conditions) plays an important role in the expression system. However, despite of the complexity of biological systems, we can observe general abstraction principles which allow simplifying such complexity during the modelling (Alon 2007).

The availability of off-the-shelf biological parts (Rettberg 2006) will allow the rapid construction of synthetic biological systems, thanks to the enforcing of standardisation and modularisation. To be able to cope with the complexity of biological designs, we should have computational tools that

would allow constructing systems models from component modules, such as model parts. In fact, there is a strong interest in the Synthetic Biology community for creating a registry of model parts, as it is demonstrated by recent work (Rouilly et al. 2007) using CellML (Lloyd et al. 2004). Although this format contains more explicit modularity than SBML, we consider more important taking advantage of the rewards of using a standard. This is exemplified by the availability of a much larger amount of simulation packages (SBML, Funahashi et al. 2003) and generalised public resources, such as the BioModels database (Le Novere et al. 2006).

During the design of a biological system from parts, we are confronted to many alternative designs. It would be highly beneficial to have a computational assembly tool to go from a set of model parts to a model that could be simulated in most software. This would allow to quickly simulate the dynamics of possible constructions using previously deposited data characterizing each part. There are no available tools for prototype design using a modular approach. The use of automatic tools in biology constitutes one of the challenges for synthetic biologists. In addition, we need computational approaches to construct large networks incorporating all the previous information. We propose a new computational tool to produce models of biological systems by assembling models from biological parts. Although living organisms use both strands of the DNA to hold genes, we are going to restrict ourselves to synthetic biology models using the forward strand. Thus, our tool assembles just in forward direction. However, in those systems the transcription mechanism could interfere at the single cell level, being difficult to quantify it at the population level. Our software not only takes advantage of modularity, but it also enforces standardisation in part characterisation by defining a model for each part. As there is no established consensus to develop models for biological parts, we have only considered model parts to design transcriptional networks. We have constructed model parts for promoters, ribosome-binding sites (RBS), coding sequences (CDS) and transcriptional terminators (TT). We use the SBML format because it has a larger availability of tools than CellML and because we want to contribute enforcing a standard in systems biology.

Methods

The dynamics of genetic networks is often described by a set of nonlinear differential equations. The mRNA dynamics is described by

$$\frac{d}{dt}[mRNA] = POPS - \delta[mRNA], \tag{1}$$

where POPS represents the number of RNA-polymerases per unit of time transcribing the DNA, and δ stands for the mRNA degradation rate. POPS depends on the dynamics of the transcription factors regulating the promoter. We use an effective model based on Hill functions assuming fast binding reactions at the promoter region. According to that, the expression is given by

$$POPS = \frac{\alpha_0 + \alpha\left(\frac{[TF]}{K}\right)^n}{1 + \left(\frac{[TF]}{K}\right)^n}, \tag{2}$$

where α_0 and α are the transcription rates at low and high transcription factor (TF) concentrations, respectively. K is the regulatory coefficient and n is the Hill coefficient. This expression can be enlarged for more TFs. On the other hand, the protein dynamics is described by

$$\frac{d}{dt}[Protein] = RIPS - \beta[Protein], \tag{3}$$

where RIPS denotes the number of ribosomes per unit of time translating the mRNA, and β is the protein degradation rate. We consider that RIPS depends linearly on the amount of mRNA:

$$RIPS = \lambda[mRNA], \tag{4}$$

where λ corresponds to the ribosome binding rate.

We consider four types of parts to design a given genetic network: promoters, RBSs, CDSs, and TTs. Our approach consists of generating the full model describing the system dynamics by assembling the models for each component part. Each model will be described by a SBML file (we provide examples that could be used as templates). The final model will also be a SBML file that could be post-processed by any SBML-compatible software such as CellDesigner.

We consider that a promoter has as inputs its regulating transcription factors and also POPS (e.g., from a TT or from another promoter), and as output POPS and mRNA. With the variable POPS we can calculate the mRNA dynamics. We do not consider RNA-polymerase as an input for a promoter because we assume that its amount in the medium is enough for transcription and it does not affect in the dynamics of the network. In fact, we consider that the amount of cellular resources, such as RNAPs, ribosomes, nucleotides or amino-acids, is sufficient to sustain a foreign system. Thus, RIPS only depends on the amount of mRNA. For the RBS we consider as input POPS and mRNA, and as output POPS and RIPS. The CDS has as input POPS and RIPS, and as output POPS and protein. Finally, the TT has POPS as input and output (see Fig. 1). This choice allows generating arbitrary complex transcriptional circuits using models for promoters and TF from the registry of parts.

Implementation

This application has been developed in C and it uses the LIBSBML library (Bornstein et al. 2007) to perform the

Fig. 1 Diagram showing how to build a synthetic network by assembling different model parts from a library
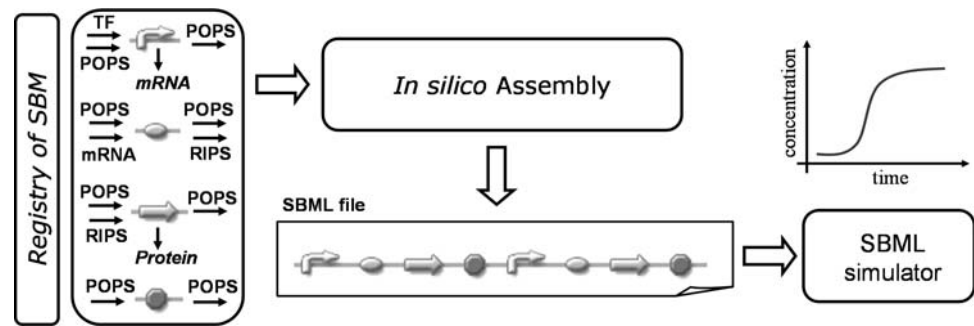
assembly between two parts. Our application is easily compiled and executed in Linux environments. The user performs the construction of its own synthetic network (or biological device) by assembling the corresponding parts, which could be taken from a library. In addition, it is possible to assemble multiple parts with a single command line. Importantly, the new generated parts are in SBML format and maintain the required format, hence they can be added to the library of model parts.

Discussion

Our software could be easily adapted to incorporate new parts (such as RNA or enzymes with their catalysed reactions), the future SBML level 3 (Hucka et al. 2003), or to consider more refined models for the parts shown in this work. Our assembly tool could be combined with an automatic procedure for transcriptional network design such as in (Rodrigo et al. 2007), which would open the way to the automated design of systems with targeted behaviour using a registry of model parts. By providing an open source tool to automatically assemble SBML models of biological parts we are favouring the means to create models on-the-fly that could be simulated to explore the feasibility of a given design. This is extremely important if we want to encourage the upload of model parts into a common repository. In addition, this will encourage experiments aimed to characterise the parameters of each model, opening the way to a standardisation of model parts. Rewiring networks from modular components is the key point towards the construction of functional synthetic organisms (Sprinzak and Elowitz 2005). However, the fact of understanding a system as the sum of its counterparts poses some difficulties due to the influence of the context. This can be addressed by designing robust circuits (Barkai and Leibler 1997). Another approach is constructing orthogonal systems. The use of these parts will constitute a big step in engineering synthetic circuits, such as using T7 RNA-polymerase or ortho-ribosomes (Rackham and Chin 2005). This software will be very valuable for the Synthetic

Biology community, not only as a tool to facilitate model design, but as a way to promote the establishment of a registry of model parts.

References

Alon U (2007) Simplicity in biology. Nature 446:497

Atkinson MR, Savageau MA, Myers JT, Ninfa AJ (2003) Development of genetic circuitry exhibiting toggle switch or oscillatory behavior in *Escherichia coli*. Cell 113:597–607

Barkai N, Leibler S (1997) Robustness in simple biochemical networks. Nature 387:855–857

Bornstein BJ et al (2007) libSBML. http://sbml.org/software/libsbml

Cox RS III, Surette MG, Elowitz MB (2007) Programming gene expression with combinatorial promoters. Mol Syst Biol 3:145

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

Funahashi A, Tanimura N, Morohashi M, Kitano H (2003) CellDesigner: a process diagram editor for gene-regulatory and biochemical networks. BIOSILICO 1:159–162

Hartwell LH, Hopfield JJ, Leibler S, Murray AW (1999) From molecularto modular cell biology. Nature 402:C47–C52

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

Le Novere N et al (2006) BioModels Database: a free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. Nucleic Acids Res 34:D689–D691. http://www.ebi.ac.uk/biomodels

Lloyd C, Halstead MD, Nielsen PF (2004) CellML: its future, present and past. Biophys Mol Biol 85:433–450

Rackham O, Chin JW (2005) A network of orthogonal ribosome-mRNA pairs. Nature Chem Biol 1:159–166

Rettberg R (2006) Registry of Standard Biological Parts. Massachusetts Institute of Technology, USA. http://parts.mit.edu

Rodrigo G, Carrera J, Jaramillo A (2007) Genetdes: automatic design of transcriptional networks. Bioinformatics 23:1857–1858

Rouilly V, Canton B, Nielsen P, Kitney R (2007) Registry of BioBricks models using CellML. BMC Syst Biol 1:P79

SBML. ERATO Kitano Symbiotic Systems Project. http://sbml.org

Segre D et al (2003) From annotated genomes to metabolic flux models and kinetic parameter fitting. OMICS 7:301–316

Sprinzak D, Elowitz MB (2005) Reconstruction of genetic circuits. Nature 428:443–448

Teusink B et al (2000) Can yeast glycolysis be understood in terms of in vitro kinetics of the constituent enzymes? Testing biochemistry. Eur J Biochem 267:5313–5329

Visser D, Heijnen JJ (2003) Dynamic simulation and metabolic redesign of a branched pathway using linlog kinetics. Metab Eng 5:164–176