GeNGe: Systematic Generation of Gene Regulatory Networks
Hendrik Hache, Christoph Wierling, Hans Lehrach, and Ralf Herwig
Vertebrate Genomics - Bioinformatics Group, Max Planck Institute for Molecular Genetics, Ihnestraße 63–73, 14194 Berlin, Germany

ABSTRACT
Summary: The analysis of gene regulatory networks (GRNs) is a central goal of bioinformatics highly accelerated by the advent of new experimental techniques, such as RNA interference. A battery of reverse engineering methods has been developed in recent years to reconstruct the underlying GRNs from these and other experimental data. However, the performance of the individual methods is poorly understood and validation of algorithmic performances is still missing to a large extent.

To enable such systematic validation, we have developed the web application GeNGe (GEne Network GEnerator), a controlled framework for the automatic generation of GRNs. The theoretical model for a GRN is a nonlinear differential equation system. Networks can be user-defined or constructed in a modular way with the option to introduce global and local network perturbations. Resulting data can be used, e.g., as benchmark data for evaluating GRN reconstruction methods or for predicting effects of perturbations as theoretical counterparts of biological experiments.

Availability: Available online at http://genge.molgen.mpg.de
Supplementary information: Supplementary data is available.
Contact: hache@molgen.mpg.de

1 INTRODUCTION
Inferring gene regulatory networks (GRNs) from experimental data is a challenging task becoming increasingly important with routine practical use of corresponding experimental techniques, such as RNA interference combined with microarray or next generation sequencing. Various computational algorithms for reconstructing GRNs from experimental data have been developed in the last decades (see Suppl. data for an overview). Besides the algorithmic developments, the actual assessment of methods performances remains a challenge, primarily due to the lack of experimental benchmark data. However, such systematic validation is crucial, since it shows strengths and weaknesses of the methods and their suitability for the specific problem domain (time series or perturbation experiments, noisiness of data, etc.)

Availability of experimental data, with a few exceptions, such as the network described by Davidson et al. (2002), is still the major bottleneck for GRN reconstruction. Hence, generating simulated data derived from theoretical considerations is still the method of choice for constructing benchmark data sets and for conducting performance studies on individual methods. These theoretical models should reflect features and complexity of real regulatory processes. They allow performance analysis under well-defined conditions using appropriate network characteristics, network complexity, noise levels, missing data, or other hidden information. This knowledge can aid further algorithmic developments and guide improvements of experimental as well as analytical methods.

There are some tools that provide such forward GRN modeling approaches, such as SynTReN (den Bulcke et al., 2006), RENCO (Roy et al., 2008), or SynBioSS (Hill et al., 2008). However, despite of their usefulness they lack some features such as automatic gene generation of different network types, manipulation of network structure, simulation of global and local perturbation, and visualization of simulation results in a single framework, specialized for GRNs (see Suppl. data).

To meet the above mentioned requirements we have developed the GRN generator GeNGe (GEne Network GEnerator), a web application to model GRNs of different types. The GRNs are used to set up a deterministic ordinary differential equation (ODE) system. The gene regulatory model system is composed of instances of mRNAs and proteins acting as transcription factors (TFs) and their corresponding target genes. Nonlinear kinetics based on the logic described by Schilstra and Nehaniv (2008) are used to describe the influence of sets of independently or jointly binding TFs on...
Hache et al.

2 FUNCTIONALITY

The workflow of GeNGe is divided into three levels (see Fig. 1). In the first level, the network level, networks are added to a network repository that will be used for further analyses and simulations. GeNGe provides several pre-defined GRNs, such as a part of the repository that will be used for further analyses and simulations.

In the next level, the kinetic level, kinetics of the model are specified. Degradation of mRNA and protein can be modeled by a linear or a Michaelis-Menten kinetic. The translation is described by a linear kinetic law. For the transcription dynamic different non-linear kinetic laws can be selected. In the third level, the simulation level, parameters of individual kinetic laws can be specified or set randomly. Based on the network topology, the kinetics, and the parameters an ODE system of the network is set-up and exported to PyBioS simulation engine via a web-services based API (Wierling et al., 2007). Besides unperturbed time series analysis, global perturbations (such as Gaussian noise) as well as single or multiple local network perturbations (e.g., knock-downs) can be introduced and the resulting steady-states of the system are computed. The procedures can be repeated with different settings and used in an iterative way.

3 EXAMPLE

An Example workflow in GeNGe is shown in Fig. 2 which is adapted from the synthetic repressilator by Elowitz and Leibler (2000). The pre-defined network “Simple Oscillator” is added to the network repository. A local perturbation of gene lacI is introduced with a knock-down degree of 80%. The control and knock-down time series are calculated and visualized in the network graph. All results, including the networks in the format of Systems Biology Markup Language (SBML), time series and simulation parameters can be downloaded for further analyses. More details about the network and data generator is given in the Suppl. data.
REFERENCES


