

Generating Synthetic Gene Regulatory Networks

Ramesh Ram and Madhu Chetty

Gippsland School of IT, Monash University,
Churchill, Victoria 3842, Australia

{Ramesh.ram, Madhu.chetty}@infotech.monash.edu.au

Abstract. Reconstructing GRN from microarray dataset is a very challenging problem as these datasets typically have large number of genes and less number of samples. Moreover, the reconstruction task becomes further complicated as there are no suitable synthetic datasets available for validation and evaluation of GRN reconstruction techniques. Synthetic datasets allow validating new techniques and approaches since the underlying mechanisms of the GRNs, generated from these datasets, are completely known. In this paper, we present an approach for synthetically generating gene networks using causal relationships. The synthetic networks can have varying topologies such as small world, random, scale free, or hierarchical topologies based on the well-defined GRN properties. These artificial but realistic GRN networks provide a simulation environment similar to a real-life laboratory microarray experiment. These networks also provide a mechanism for studying the robustness of reconstruction methods to individual and combination of parametric changes such as topology, noise (background and experimental noise) and time delays. Studies involving complicated interactions such as feedback loops, oscillations, bi-stability, dynamic behavior, vertex in-degree changes and number of samples can also be carried out by the proposed synthetic GRN networks.

Keywords: Causal model, synthetic gene regulatory networks, microarrays.

1 Introduction

The reconstruction of gene regulatory networks (GRNs) using microarray datasets is amongst the major challenges currently being investigated in the field of molecular biology research. Reconstruction of GRN provides researchers an opportunity to form new hypotheses related to the behavior of biological systems a-priori to the experiments to be carried out, which in turn prevents performing expensive and lengthy biological experiments thereby expediting the discovery process. In domains other than bioinformatics, datasets studied typically have very few features and large number of samples. However, in case of microarray dataset, they invariably have large number of genes with very few samples. Consequently, traditional statistical approaches for analyzing these microarrays become inadequate and the need for applying other techniques becomes necessary. A plethora of modeling and inference techniques, starting from standard multivariate statistics to machine-learning and

heuristics [1-5], are available to reconstruct GRN from large-scale gene expression data sets. While using these techniques, the validation and evaluation of methods using real life datasets is limited since such datasets may not always be properly documented. In other words, the validation is limited by the information that was previously established independently by other approaches.

A synthetically reconstructed GRN, while preserving the characteristics of the underlying data generation system, allows different experiments to be performed to investigate the effect of parametric variations. A synthetic dataset permitting large variety of parametric variations is a possible solution which will allow more rigorous testing and evaluation of methods for reconstructing GRNs. Although limited, efforts for generating synthetic data for GRN reconstruction are available. Mendez *et al* [6] proposed a method based on differential equations for generating synthetic microarray data. However, the method allows variation of only noise and topology parameters and does not include the flexibility of varying single or combination of parameters for validating individual features of the GRN methods. Eisen *et al* [7] generated synthetic dataset and applied for studying hierarchical clustering for gene expression data. As the method suffered from the lack of knowledge about the GRN under study, any conclusion vis-à-vis the underlying biology became uncertain. Further, because the data sets were different in each of the studies carried out, it was not possible to make any comparisons amongst studies that employed this approach. Friedman *et al* [1] generated a Boolean synthetic data to validate the robustness of their Bayesian methods. Although useful for generating synthetic datasets, none of these techniques were suitable to examine model specific features such as time-delays, feedback loops, dynamic behavior, etc. Furthermore, all these techniques were limited in their ability to generate a variety of synthetic networks at different stages of refinement of GRN reconstruction methods.

In this paper, we present a novel causal modeling method for synthetically generating GRN which includes all GRN related features that are commonly modeled in reconstruction algorithms. The variation of these features, in a controlled way, determines the desired level of complexity of the synthetically generated gene expression data. The proposed synthetic generation of networks is along the lines of our ongoing work on causal modeling for reconstruction of real-life GRN [8-12] wherein we have investigated the application of causal modeling technique to *Saccharomyces cerevisiae* (yeast) [15] microarray dataset. The obtained results are in close agreement with known biological findings thus validating the modeling process. Briefly, the causal GRN reconstruction from microarray begins with the application of a network structure construction algorithm resulting in a large number of possible GRN structures. A continual evaluation and evolution of these structures results in a structure that best fits the microarray data results and is considered as the desired GRN model. The complexity of reconstruction is increased gradually at each stage of refinement of the model. The rest of the paper is structured as follows: Section 2, a brief overview of causal modeling is given. Section 3 elaborates on the system and methods used to realistically generate the synthetic data. Section 4 provides experiments and results. Finally, section 5 provides concluding remarks on the paper and some future work.

2 The Causal Modeling Approach

A causal GRN structure is represented by a directed graph whose nodes represent the genes and directed edges between nodes indicate their causal relationship. Pioneering work in causal modeling was reported by Pearl *et al*, and Sprites *et al* [13, 14] who proposed algorithms to infer a causal structure from experimental data by using partial correlations if the underlying causal structure was a directed, acyclic graph (DAG). In our approach for causal GRN reconstruction, the central step of determining the fitness of the data given the whole network is decomposed into determining the set of scores of local models that includes fitness of structure, direction of causality and sign (positive/ negative) of regulation. The task of network reconstruction is cast into a search for candidate gene networks whose scores are high. To implement a heuristic search method, we apply a genetic algorithm (GA), whereby creating and evolving different networks to eventually obtain a network that best fits the microarray data. Due to the stochastic nature of the GA, the GA is repeated few times and the resulting network structures are combined in a predefined manner to reconstruct the final gene network. While evaluating the fitness, the putative network is actually decomposed into Markov Blankets (MB) and conditional independence tests are applied in order to detect whether or not connections are direct or indirect. The direction and sign of regulation are recovered by estimating the time delay and correlation between expression profiles of pairs of genes.

3 Methodology for Generation of Synthetic Data

The proposed method for generation of synthetic networks allows for various parametric variations, such as, network topology, varying levels of complexity of interaction, time delays, number of samples and amount of noise in the data.

Figure 1 shows the flow chart of the mechanism of proposed system for synthetic network generation. The synthetic network generator, written in MATLAB, offers an option for choice of topologies that determines the structure of the network and specifies interactions between the genes. With this option, we can generate any number of networks having different topologies. In the next step, by choosing interactions and setting equation parameters, the full dynamics of the gene network (such as feedback loops, oscillations and so on) is described and can be implemented in specified pre-defined ways to produce a required level of complexity of gene interactions. Next, for generating discrete samples, the continuous responses of the genes in the synthetic network are sampled at different time instants to produce a noiseless time course data. Next, to make the sampled data realistic, time delays are added to the samples in a specified manner. Following this, noise is added to the data according to the Gaussian or gamma distributions. Finally, gene expression ratios are calculated which realistically represent the real-life microarray data set.

In the following section, the entire process of generating the network topology and corresponding gene interactions is described in detail.

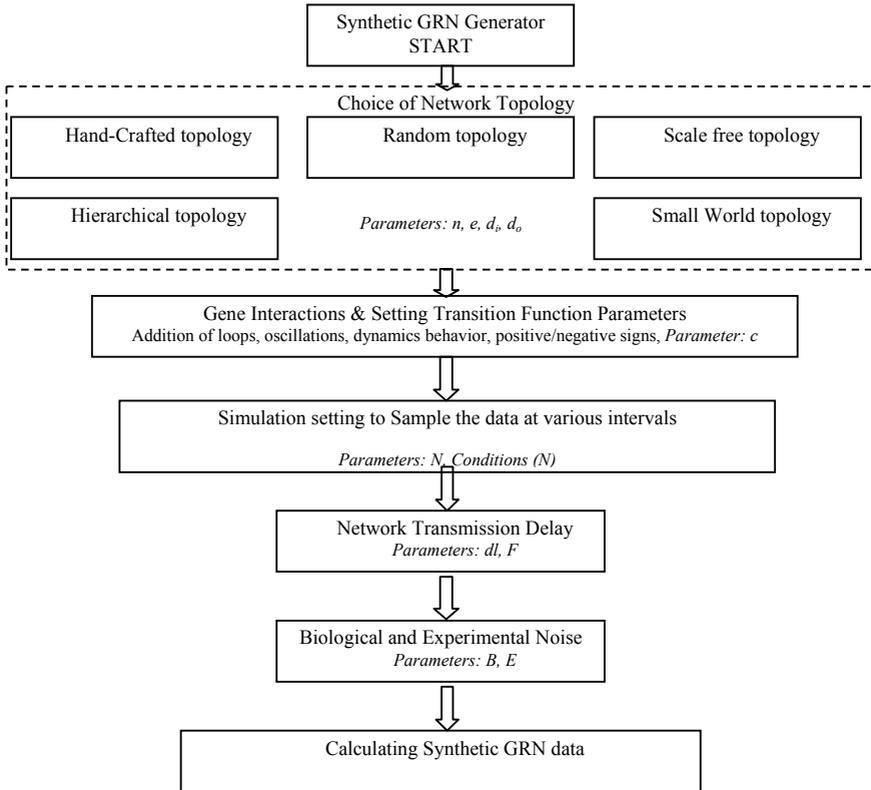


Fig. 1. Proposed methodology of synthetic gene expression data generation, The symbols used are: n - number of genes, e - number of edges, d_i - incoming degree distribution, d_o - outgoing degree distribution, c - percentage of complex interactions, N - number of samples, Condition (N) - specifies experimental conditions for each sample as in real each sample is an experiment, dl - delay levels, F - probability distribution of delays, B - percentage of biological noise in terms of hidden nodes, E - percentage of experimental noise.

3.1 Network Topology

As mentioned earlier, the first step of synthetic data generation is to define a network topology. A topology is chosen by setting following three parameters:

- i) Total number of genes in the network,
- ii) Distribution of the incoming degree of connectivity (i.e. the distribution of the number of parents per gene) and
- iii) Distribution of the outgoing degree of connectivity (i.e. the distribution of the number of children per gene).

Based on the incoming and outgoing degree distribution parameters mentioned above, four different topologies are available for selection (with corresponding distribution provided in parenthesis):

- Random topology (Poisson distribution)
- Scale Free topology (power law distribution)
- Small World topology (power law distribution with small average distance between genes)
- Hierarchical topology (power law distribution with inherent modular structure)

In random topology (RND), the connectivity degree follows a Poisson distribution. The nodes that deviate from the average are rare and decrease exponentially and the clustering coefficient is independent of a node's degree of connectivity [16]. In Scale Free (SF) topology [17], the connectivity degree follows a power law distribution, i.e. the behavior of a network system is controlled by few important nodes. Majority of nodes have only a few connections, while some special nodes connect with many other nodes forming a hub, i.e., most nodes are poorly connected, while a few are highly connected (Hubs). In a Small World networks (SW) [18], the mean shortest path is $1 \sim \log(N)$ indicating that most nodes are connected by a short path. SW networks are characterized by large Clustering Coefficient and small Average Path Length. The Hierarchical network (HR) [19] integrates a scale-free topology with an inherent modular structure by generating a network that has a power-law degree distribution with degree exponent $\gamma = 1 + \ln 4 / \ln 3 = 2.26$.

In cases where the aforementioned topological types are not appropriate due to the uncertainty of GRN topology, we propose another topology, which we will refer henceforth as, 'handcrafted topology'(HC). The choice of any of the network topology is user-definable and can be used for checking robustness of algorithm against topology. To generate a network topology close to real life GRN, network structures previously described in biological literature such as *E. coli* [20] and *S. cerevisiae* [21] were taken into account. These networks are partially random and partially scale free i.e. the distribution of the incoming degree of connectivity follows a Poisson distribution (random topology) while the distribution of the outgoing degree of connectivity follows a power-law (scale free topology). A single topology or combinations of two or more topologies to generate the gene network structure is user definable.

At this stage, the network structure is without any complex interactions, such as self loops, oscillations and dynamic behaviour. In the next section, we present the inclusion of these features to the network topology.

3.2 Gene Interactions and Transition Function Parameters

After generating the topology, transition functions representing the regulatory interactions between the genes are assigned to the edges in the network as follows:

- i) Choosing the regulatory interactions
- ii) Setting the transition function parameters

The entire synthetic modeling of gene networks essentially considers a causal interaction of genetic regulation. It considers each gene to be directly affected by number of other genes and represents the interaction as directed edges. A transition function defines the relationship between gene and its parent genes. The genes are

represented as continuous variables rather than discrete variables, i.e. synthetic gene expression values are continuous rather than 0 or 1. First, while choosing the regulatory interactions, the genes are represented as activators or repressors. Our proposed method of network modeling allows for this positive or negative linear causal relationship between the input (i.e. parent) genes and the gene under consideration. Mathematically, these network models are based on set of linear causal equations. Each equation corresponds to gene expression which is a function of a positive (activation) and negative (repression) terms. When a given gene interacts with more than one regulator, different regulators can either act independently or in a more complex manner (such as complex combinational, short term co-activation, co-repression or a combination) on the target genes resulting in different interactions such as feedback loops, oscillations and dynamic behavior.

To incorporate such complexities, for each combination of a gene and its regulators, appropriate equation is selected, depending on the number of activators and repressors and on the user-defined settings that control the fraction of complex interactions. For genes involved in *cycles*, it is possible that not all inputs of their transition function are known during loop propagation. To model these loops, an approximation compatible with the steady-state transition functions is chosen. This approximation is represented by a parameter to represent complex interactions. It is an extremely useful parameter because it allows initial performance evaluation of a method to be done on relatively easy problems (e.g. small noiseless networks without complex interactions between regulators). Increasingly difficult data sets can subsequently be generated as the GRN inference method is improved or refined. Again, setting transition function parameters involves choosing appropriate correlation parameter settings of the transition function equations. The strength of correlation is an important parameter and is chosen from a distribution that allows a large variation of interaction that are likely to occur in true networks (including linear activation functions, sigmoid functions, sinusoidal functions, etc.), while avoiding very steep transition functions. To explain a simple chain interaction in the network considers, for example, that x causes y and y causes z . That is, $x \rightarrow y \rightarrow z$

$$x(t) = A\sin(Bt) ; y(t) = x(t) ; z(t) = y(t) \quad (1)$$

The expression $x(t)$ is a sinusoid with amplitude A , time period $2\pi/B$ where B is angular frequency. In this case, the strength of correlation between x and y is 1, so the signals are equal, but varied based on parametric specification.

3.3 Data Samples

Using the continuous gene expression output (resulting from the equations written for each node of the synthetic network), data is sampled at either fixed or irregular time spacing between gene expressions. The number of samples and the time step for sampling can be chosen either randomly or it can also be user defined. The sampled data represents the temporal state of synthetic network under different experimental conditions. This is similar to real microarray experiments where each sample of the dataset is an experiment that is repeated at fixed or irregular intervals of time. At this stage, various settings needed for simulation of the network per each sample (simulating a real experiment setup) for N sample is complete. However, note that the

data representing real life conditions is not yet generated as time delay and noise component are yet to be added.

3.4 Network Transmission Delays

A delay in transmission of signals emitted by genes, being an important characteristic of all gene networks; it is important to realistically implement this feature in synthetic datasets. In the proposed modeling approach, we implement the delay levels as a user defined parameter which is nothing but the maximum number of samples on which the delay can be experienced. Further, to make the modeling more realistic, we have also made it possible to specify the fraction of interactions which have delays. Based on the choice of this parameter, a delay distribution is obtained for the links between the genes. Delays are implemented by simply reassigning a new simulation setting for a particular sample explained in section 3.2 based on the delays assigned. This simulates the delay in the real microarray dataset. The fraction of links involved in time delay is determined using a known probability density in case it is not user defined. Investigations involving time delay parameter variation can thus be carried out on the datasets by incorporating/eliminating time delays.

3.5 Biological and Experimental Noise

A real life microarray data contains two types of noises, namely biological and experimental. The biological noise corresponds to stochastic variations in gene expression, and this noise is unrelated to the applied experimental procedures. It is present due to, for example, environmental conditions such as temperature, pressure, etc. While experimental noise is the noise due to the technique used to extract the data. Both these noises also should be appropriately included in the simulated data.

Briefly, biological noise is added by the presence of hidden background nodes which are either genes or conditions and experimental noise is added as Gaussian white noise. First, the background hidden node (for incorporating biological noise), which is a parameter to choose the amount of background noise, is user defined. The equations of the background noise nodes are generally uncorrelated to the genes on which they are acting. A limited number of input nodes are selected that mimic the external conditions and consider the genes not linked to these input genes act as background nodes. These are now part of the simulation set up while the data is not generated.

As the real microarray data also has experimental noise, three user defined choices for addition of experimental noise are made available: i) Lognormal ii) Gaussian iii) Gamma distributions. All these distributions take a percentage of the amount of noise as input which is then applied to make the final output data noisy. However, this experimental noise is added only after the simulated microarray data is generated. This is explained in section 3.7.

3.6 Synthetic Network Generator Parameters

The entire flow chart for the generation of synthetic data is given in Fig. 1 which also shows the system and the parameters controlling the synthetic data generation at every step of the process. These parameters which are listed below can each be varied

independently either before or during the simulation process for conducting simulated experiments with synthetic data:

1. Choice of source network.
2. Size of the network in number of nodes.
3. Number of background nodes.
4. Number of available experiments and samples for each condition.
5. Level of stochastic and experimental noise.
6. Fraction of complex interactions.

3.7 Calculating Synthetic GRN Data

Using the synthetic network generator described earlier, simulations are next performed to generate the synthetic microarray data. The genes without regulatory inputs are assigned an arbitrary expression level which can be changed during an experiment (sample). The expression levels of the genes in the network are calculated, as specified by their transition functions, starting from the input genes. After these noise-free expression values are computed, noise is then appropriately incorporated in the data to reflect noise present in the real microarray data. These computed noisy expression values can be used for analyzing the noise which a GRN reconstruction method under investigation can handle. This feature of adding noise enables the comparison of level of noise in dataset on the reconstruction algorithms. A gene expression profile experiment for different time t corresponds to a vector $[x_1(t) \dots x_n(t)]$. For a set of N samples, a $n \times N$ matrix is constructed which is the *final synthetically generated microarray dataset*. This dataset can be used for investigation and evaluation of various GRN reconstruction algorithms.

4 Experiments and Results

In order to conduct tests using synthetic data set, several datasets are created by varying network generator parameters (one or two at a time). The group of data sets which have similar variations are categorized into one of the four groups A, B, C or D (see Table 1). Although the experiments involved significantly large number of data sets to test robustness of GRN methods, due to space restriction, only a limited number of important models have been included in the paper and shown in Table 1.

The Group A consists of a set of synthetic network models which are used for investigating methods for their robustness against network topology. With this group, we carry out an initial level of testing since it contains no complex interactions and also because the effect of the noise is kept low. Different sample sizes help determine accuracy of reconstruction as generally most methods require higher sample size data to make accurate estimations.

The Group B networks compare two different network topologies, namely SF and RND. Compared to Group A, these are large sized networks of 500 genes and 500 interactions. Fig.2 (a) shows networks that follow a random topology (RND) while the network shown Fig. 2 (b) is a scale-free (SF) network. From the figure, we can observe the differences resulting due to two differing topologies. The random topology has arbitrary arrangement of links throughout the network while the scale

Table 1. The Synthetic data sets are organized in four groups A, B, C, D. Column 2 gives different network topologies: Scale Free (SF), Small World (SW), Random (RND) and Handcrafted (HC). For each group, column 3 shows the number of repeated models for a given experiment. Column 4 and column 5 respectively give the number of genes and the edges in a given model. Column 6 gives the % fraction of complex interactions. Column 7 gives the network transmission delay. Column 8 gives the number of parents while column 9 gives the %ge noise of each model. Column 10 gives number of samples for each condition.

1	2	3	4	5	6	7	8	9	10
Group	Topology	No. of Models	Genes	Edges	% Complexity	Delay	No. of parents	% Noise	Samples
A	SF	50	100	200	20	0	2	1	20, 50, 100
	SW	50	100	200	20	0	2	1	20, 50, 100
	RND	50	100	200	20	0	2	1	20, 50, 100
	HC	50	100	200	20	0	2	1	20, 50, 100
B	SF	5	500	500	40	2	5	5	50
	RND	5	500	500	40	2	5	5	50
C	SF	50	50	50	20	1	3	5	50
	SF	50	50	100	20	1	4	5	50
	SF	50	50	200	20	1	7	5	50
D	SF	10	100	200	40	1	1	5	20, 50
	SF	10	100	200	30	2	2	1	20, 50
	SF	10	100	200	50	-2	3	5	20, 50
	SF	10	100	200	10	3	4	1	20, 50
	SF	10	100	200	40	-3	4	5	20, 50
	SF	10	100	200	30	4	5	1	20, 50
	SF	10	100	200	50	0	3	10	20, 50
SF	10	100	200	10	0	2	1	20, 50	

free network has hubs with large proportion of links in the top right corner of the figure while lesser number of links in the rest of the figure. Note that the number of genes and gene interactions is the same for the two cases under consideration. Since scalability is an important feature of GRN algorithms, this group enables to justify if the algorithm is robust in terms of size.

In Group C, the number of genes in the networks is kept fixed at 50 and the topology chosen for study is Scale Free. The number of links is varied as 50, 100, 200. This group is useful for checking robustness of methods with respect to density of connectivity (i.e. no. of parents per gene) along with accuracy with respect to number of samples.

The Group D is designed to test the combinational effect of density of connectivity and also to include varying delays and noise intensity parameters resulting in an increasing average number of connections per gene. The D group tests are for advanced level testing of GRN algorithms as the data generated is from a complex complicated network of interactions. Because these gene networks are generated with random connectivity for each of the rows in Table 1, we repeated the generation of models for specified number of times (see column 3) and took the average results

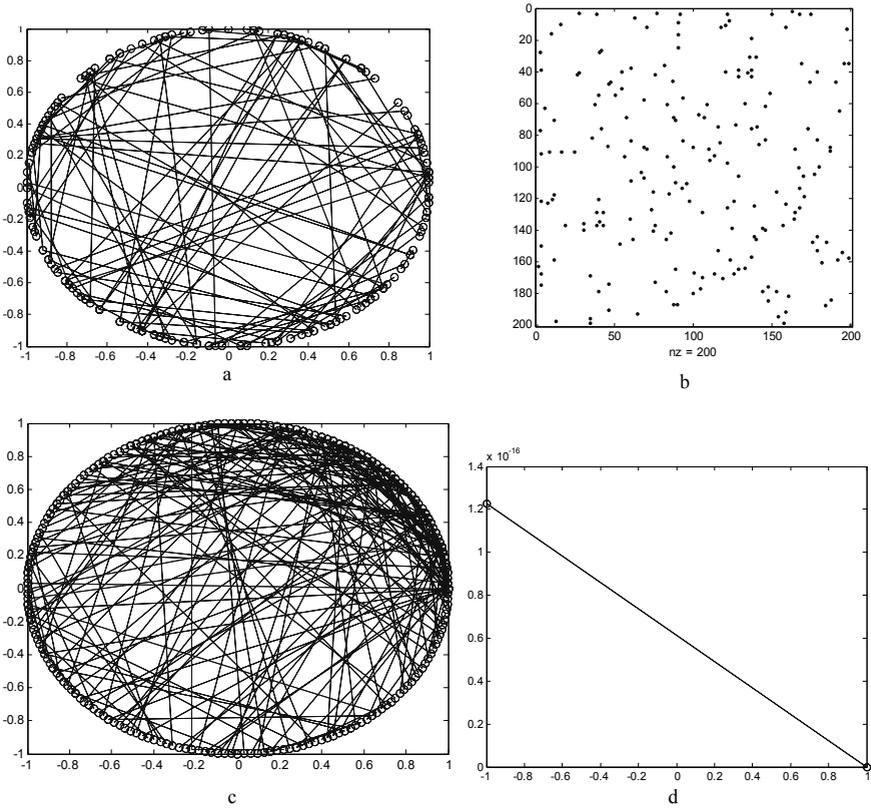


Fig. 2. Two illustrative network topologies of B Group: a- RND Network Topology ($n=200$, $e = 100$, $d_i = d_p =$ Poisson pdf), b – Adjacency matrix ($n \times n$) of the random topology, c - Scale Free Network Topology ($n=200$, $d_i = d_p =$ Power law pdf), d – Power Law function, $f(x)$ on y -axis.

from each row to get a synthetic dataset which is close to real dataset. The simulation results from this synthetic dataset are shown in Figure 3.

For these simulations, an example network given in Table 1 is considered. . The part of the network under consideration (also known as Markov Blanket of gene A is shown in figure 3(a). For investigations involving noise, we consider four different types of noise type’s namely a) Lognormal b) Gaussian c) Linear and d) Constant. The variation of these noises as a function of gene expression is shown in Fig 3(b). For purpose of experimentation, the amount of noise is added as a function of the gene expression (i.e. mRNA accumulation). In Figure 3(c), the expression of synthetic gene A is shown for both conditions: with and without noise. These plots show the effect of noise on the synthetic gene expression. Again, Figure 3(d) shows the effect of an input gene (such as gene B) and an output gene (such as gene D) on another gene A in the presence of time delay and regulation (plus or minus). As can be seen

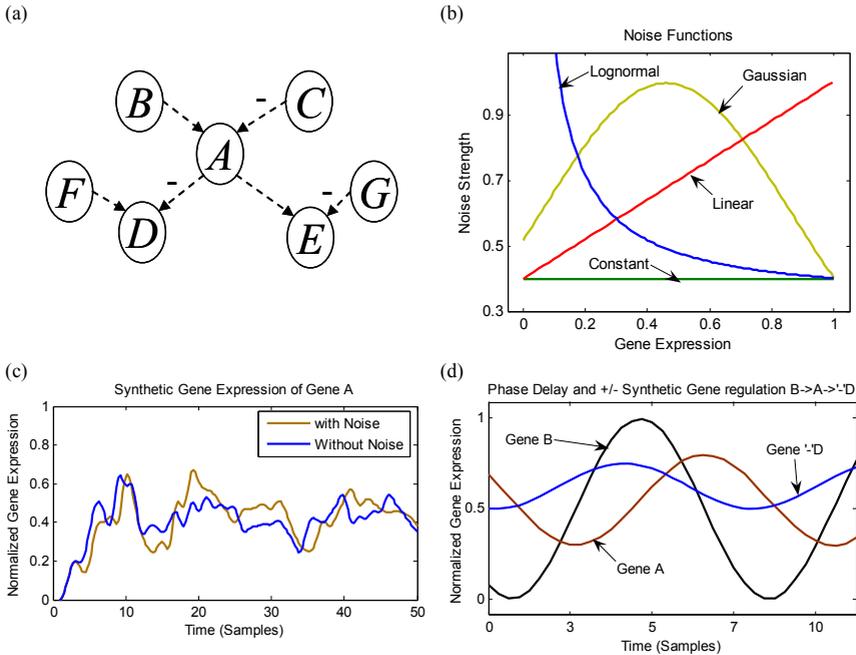


Fig. 3. Simulation results: (a) A subset of genes of the example network (labeled A to G). This sub-network has two input genes and contains repressor gene D. (b) The different noise functions used in the simulation. (c) The expression gene A with and without addition of noise. (d) Shows phase shift (time delay) and plus/ minus regulation between $B \rightarrow A \rightarrow D$.

from Figure 3(a), gene B has a strong effect on gene A, but a pronounced negative effect on the expression level of gene D. This is because gene D has a repressor link and is directly stimulated by gene A and also indirectly by input gene B.

5 Conclusion

The network generator system presented in this paper generates synthetic GRN datasets based on causal modeling approach for GRN [8]. Illustrative investigations using the network generator show the significance of the application of system for synthetic data generation. The proposed system can generate four different network topologies, namely scale free, small world, random and hierarchical. Further, the generated synthetic network is made realistic by incorporating complex network characteristics such as transmission delays, biological and experimental noise. These datasets are generated for evaluation of methodologies based on these synthetic datasets. The system will help other similar methods to computationally determine the robustness and also establish comparisons between the methods. In comparison to other existing methods, the proposed system is useful in carrying out rigorous studies about the GRN methods by particularly varying single and combinational features of the networks such as the topology, interaction types, noise levels, time delay of the

interactions and so on. The complexity of the generated datasets can be easily varied by parametric variation thereby generating a hierarchy of networks in terms of size, scales, samples, etc. In real life GRNs, the products of some specific genes are essential for transcription to take place (e.g. mRNA), and their absence cannot be counteracted by increased expression of other activators (e.g. the transcription factor TF). Although the synthetic gene network models and datasets generated here are simplistic in comparison to what actually happens between genes in the real world biology to produce the microarray dataset, they mimic the characteristics of experimental data which makes it suitable to test the methods used on real datasets. Furthermore, since the ground truth about real world GRN is still unknown to a certain extent and hence to make any significant advances towards understanding gene networks by using artificial synthetic networks and datasets will be highly important and useful for future analysis of very complex GRN models.

Although only additive interactions for different hidden nodes have been included in this paper, it is easily possible to enhance the system further by including non-additive interactions between the activators and repressors. All these improvements in generation of synthetic networks will make the network models more realistic. With a clear understanding gene regulation problem, it would be possible to simulate the process of complex gene regulatory networks. Future developments in this research will include exploring large numbers of diverse synthetic gene networks to search for particular properties similar to real gene networks, and expanding the system to protein-protein interaction networks.

References

- [1] Friedman, N., Linial, M., Nachman, I., Pe'er, D.: Using Bayesian networks to analyse expression data. *Journal on Computational Biology* 7, 601–620 (2000)
- [2] Liang, S., Fuhrman, S., Somogyi, R.: REVEAL, a general reverse engineering algorithm for inference of genetic network architecture. In: *Pacific Symposium on Biocomputing*, vol. 3, pp. 18–29 (1998)
- [3] Ando, S., Iba, H.: Inference of gene regulatory model by genetic algorithms. In: *Proc. Conference on Evolutionary Computation*, pp. 712–719 (2001)
- [4] Wahde, M., Hertz, J.: Modeling genetic regulatory dynamics in neural development. *Journal on Computational Biology* 8, 429–442 (2001)
- [5] Mendes, P., Sha, W., Ye, K.: Artificial gene networks for objective comparison of analysis algorithms. *Bioinformatics* 19, 122–129 (2003)
- [6] Ram, R., Chetty, M., Dix, T.I.: Fuzzy Model for Gene Regulatory Networks. In: *2006 IEEE Congress on Evolutionary Computation (CEC)* (2006)
- [7] Eisen, M.B., Spellman, P.T., Brown, P.O., Botstein, D.: Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. USA* 95, 14863–14868 (1998)
- [8] Ram, R., Chetty, M., Dix, T.I.: Causal Modeling of Gene Regulatory Network. In: *IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)* (2006)
- [9] Ram, R., Chetty, M., Dix, T.I.: Learning Structure of Gene Regulatory Networks. In: *6th IEEE International Conference on Computer and Information Science (ICIS)* (2007)
- [10] Ram, R., Chetty, M.: A Guided genetic algorithm for Gene Regulatory Network. In: *2007 IEEE Congress on Evolutionary Computation (CEC)* (2007)

- [11] Ram, R., Chetty, M.: Framework for path analysis for learning Gene regulatory network. In: Pattern Recognition in Bioinformatics. Springer – LNBI publication, Heidelberg (2007)
- [12] Ram, R., Chetty, M.: Modelling Gene regulatory networks. In: Applications of Computation Intelligence in biomedicine and Bioinformatics. Springer, Heidelberg (accepted, 2007)
- [13] Sprites, P., Glymour, C., Scheines, R.: Causation, Prediction, and Search: Adaptive Computation and Machine Learning, 2nd edn. MIT Press, Cambridge (2000)
- [14] Pearl, J.: Causality: Models, Reasoning and Inference. Cambridge University Press, Cambridge (2000)
- [15] Spellman, P.T., Sherlock, G., Zhang, M.Q., Iyer, V.R., Anders, K., Eisen, M.B., Brown, P.O., Botstein, D., Futcher, B.: Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mol. Biol. Cell.* 9, 3273–3297 (1998)
- [16] Erdős, P., Rényi, A.: On random graphs. *Publ. Math. Debrecen* 6, 290–297 (1959)
- [17] Barabasi, A.L., Albert, R.: Emergence of scaling in random networks. *Science* 286, 509–512 (1999)
- [18] Calvert, K.L., Doar, M.B., Zegura, E.W.: Modeling Internet topology. *IEEE Communications Magazine* 35, 160–163 (1997)
- [19] Rahmel, J.: SplitNet: A Dynamic Hierarchical Network Model. In: AAAI/IAAI, vol. 2, p. 1404 (1996)
- [20] Featherstone, D.E., Broadie, K.: Wrestling with pleiotropy: genomic and topological analysis of the yeast gene expression network. *Bioessays* 24, 267–274 (2002)
- [21] Jeong, H., Tombor, B., Albert, R., Oltvai, Z.N., Barabási, A.-L.: The large-scale organization of metabolic networks. *Nature* 407, 651–654 (2000)