Inference of Genetic Regulatory Networks by Evolutionary Algorithm and H_∞ Filtering

Lijun Qian and Haixin Wang
Department of Electrical and Computer Engineering
Prairie View A&M University
Prairie View, Texas 77446
Email: LiQian, HWang@pvamu.edu

***Abstract*— The correct inference of genetic regulatory networks plays a critical role in understanding biological regulation in phenotypic determination and it can affect advanced genome-based therapeutics. In this study, we propose a joint evolutionary algorithm and H_∞ filtering approach to infer genetic regulatory networks using noisy time series data from microarray measurements. Specifically, an iterative algorithm is proposed where genetic programming is applied to identify the structure of the model and H_∞ filtering is used to estimate the parameters in each iteration. The proposed method can obtain accurate dynamic nonlinear ordinary differential equation (ODE) model of genetic regulatory networks even when the noise statistics is unknown. Both synthetic data and experimental data from microarray measurements are used to demonstrate the effectiveness of the proposed method. With the increasing availability of time series microarray data, the algorithm developed in this paper could be applied to construct models to characterize cancer evolution and serve as the basis for developing new regulatory therapies.**

I. Introduction

A genetic regulatory network (GRN) is a collection of DNA segments in a cell which interact with each other and with other substances in the cell, thereby governing the gene transcriptions. The correct inference of genetic regulatory networks plays a critical role in understanding biological regulation in phenotypic determination and it can affect advanced genome-based therapeutics. In light of the recent development of high-throughput DNA microarray technology, it becomes possible to discover GRNs, which are complex and nonlinear in nature. Specifically, the increasing existence of microarray time-series data makes possible the characterization of dynamic nonlinear regulatory interactions among genes. The modeling, analysis and control of GRNs are critical for studying cancer evolution and may serve as the basis for developing new regulatory therapies.

Because GRN models are difficult to deduce solely by means of experimental techniques, computational and mathematical methods are indispensable. Much research has been done on GRN modeling by *linear* differential/difference equations using time-series data, for example, [1-8], just to name a few. The basic idea is to approximate the combined effects of different genes by means of a weighted sum of their expression levels. In [5], a connectionist model is used to model small gene networks operating in the blastoderm of Drosophila. In [1], the concentrations of mRNA and protein are modeled by linear differential equations. A simple form of linear additive functions is suggested by [2], where $dx_i/dt = \sum_{j=1}^{n} w_{ij}x_j$. The degradation rate of gene i's mRNA and environmental effects are assumed to be incorporated in the parameters w_{ij} and their influence on gene i's expression level x_i is assumed to be linear. A method to obtain a continuous linear differential equation model from sampled time-series data is proposed in [7]. For added biological realism (all concentrations get saturated at some point in time), a sigmoid (squashing) function may be included into the equation. It has been shown that this sort of quasi-linear model can be solved by first applying the inverse of the squashing function [3].

In our study, a GRN is modeled by continuous nonlinear Ordinary Differential Equations (ODEs). Compared to linear models, identification of the nonlinear differential equation model is computationally more intensive and can require more data; however, the range of nonlinear behaviors exhibited by GRNs can be more thoroughly understood with nonlinear differential equations. In addition, well established dynamical systems theory is available to characterize the dynamics produced by these models. When more time-series data become available owing to advances in microarray or other technologies, and assuming continued improvement in computational capability, it can be expected that continuous nonlinear dynamic models will play a critical role in revealing complicated gene behavior.

In general, modeling gene regulatory networks is a nonlinear identification problem. Assuming there are N genes of interest and x_i denotes the state (such as the microarray reading) of the i^{th} gene, then the dynamics of the GRN may be modeled as

$$\frac{dx_i}{dt} = f_i(x_1, x_2, \cdots, x_N) + \nu_i \qquad i = 1, 2, \cdots, N. \tag{1}$$

where the nonlinear functions f_i need to be determined from time-series microarray measurements. In this study, we assume the functions (f_i, $\forall i$) are in the form of polynomials.

$$f_i = \sum_{j=1}^{L_i} [(w_{ij} + \mu_{ij})\ \Omega_{ij}(x_1, x_2, \cdots, x_N)] \qquad i = 1, 2, \cdots, N. \tag{2}$$

where L_i is the number of terms in f_i, w_{ij} are the parameters to be estimated and $\Omega_{ij}(x_1, x_2, \cdots, x_N)$ is the j^{th} component of the nonlinear function f_i. The polynomials are utilized

as universal approximators. In order to mitigate the effect of "the curse of dimensionality", only second-degree polynomials are selected. Note that an advantage of using low-degree polynomial models is that even when there exists some model mismatch, these models may be sufficiently accurate to represent many real systems, and thus are widely utilized in practice [9]. We note that a similar GRN model has been adopted by [10], but without noise being included in the model. μ_{ij} and ν_i are parameter noise and external noise, respectively, and it is assumed that the noise statistics such as the covariance matrices are *unknown*.

The noisy nature of GRNs is modeled explicitly in this study. The deterministic model (without noise) corresponds to the nominal case, while the various stochastic effects are included as noise disturbances. For example, there is considerable experimental evidence that indicates the presence of significant stochasticity in transcriptional regulation in both eukaryotes and prokaryotes [11]. The inherent stochasticity of biochemical processes (transcription and translation) is modeled as noise in the parameters (μ_{ij}), which corresponds to the "intrinsic noise" mentioned in the literature [12]. Other effects, such as those from genes not been included in the microarray, the amount of RNA polymerase, levels of regulatory proteins, and the effects of mRNA and protein degradation, are modeled by the external noise (ν_i) [12]. Previous work has modeled these noise types by Gaussian white noise processes [13]. If the noise statistics are known, then Kalman filter can be applied to get the optimal estimates of the parameters [21]. On the contrary, a robust filter such as an H_∞ filter, has to be used to obtain the optimal parameter estimates when the noise statistics such as the covariance matrices are *unknown*.

In this paper, a two-step procedure is proposed to identify f_i. Firstly, genetic programming (GP) is applied to determine the nonlinear terms; then the corresponding parameters associated with each term are estimated by H_∞ filtering.

The remainder of the paper is organized as follows: The proposed framework and the iterative algorithm are illustrated in Section II. Simulation results are given in Section III. Section IV contains some concluding remarks.

II. Algorithm Description

The task of identifying gene regulatory networks may be considered as an optimization problem. The goal is to minimize the identification error and keep the model as simple as possible, which may be achieved by minimizing the following fitness function

$$fitness = \sum_{i=1}^{N}[\eta_1 \sum_{k=1}^{M}(x_i(k) - x_i^{tar}(k))^2 + \eta_2 \Gamma_i] \quad (3)$$

where M is the number of data points, x_i^{tar} be the target time series and x_i be the obtained time series given by the obtained differential equation. Γ_i is a penalty term and it is chosen as the number of terms in f_i, i.e., $\Gamma_i = L_i$. $\eta_1 > 0$ and $\eta_2 > 0$ are the weights on the estimation error and the model complexity, respectively.

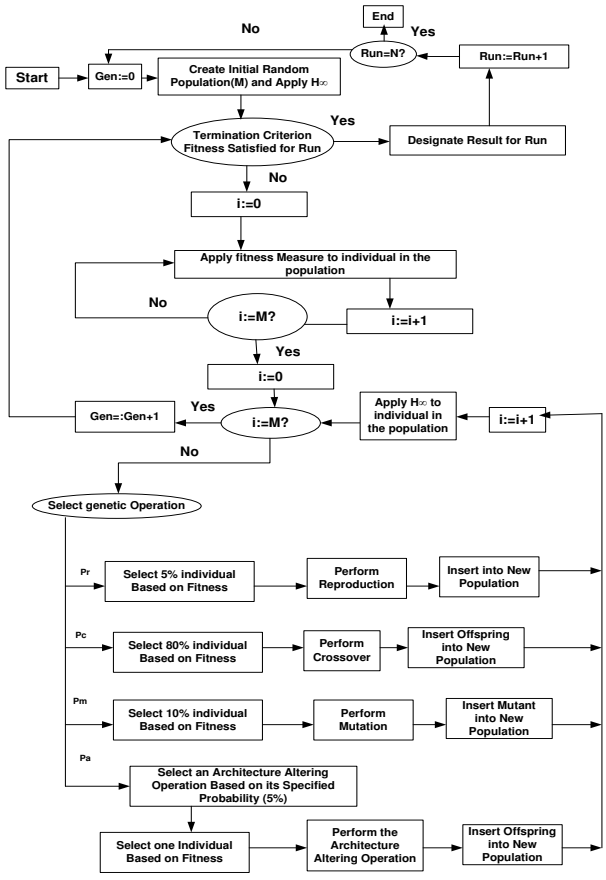

Fig. 1. The iterative process of joint GP and H_∞ filtering. (The genetic programming process has four operations: reproduction, crossover, mutation and selection. H_∞ filtering is employed to estimate the parameters for every generation.)

Since it is a global nonlinear optimization problem, a nested optimization structure is adopted, where genetic programming is applied to determine the nonlinear terms (global optimization) while H_∞ filtering is employed to estimate the corresponding parameters for each term (local optimization) in each iteration. Such a decomposition of the problem into a structural part solved by GP and a parameter optimization part solved by H_∞ filtering reduce the complexity significantly and speed up convergence. The detailed procedures of the proposed iterative algorithm is illustrated in Fig. 1.

In order to deal with large numbers of genes, the optimization problem is decoupled into N sub-problems with the i^{th} sub-problem focusing on the i^{th} gene. Because the time-series data of other genes are fixed (from measurements) when we are focusing on an individual gene, we can solve the optimization problem one gene at a time. This approach makes the inference of large GRNs feasible.

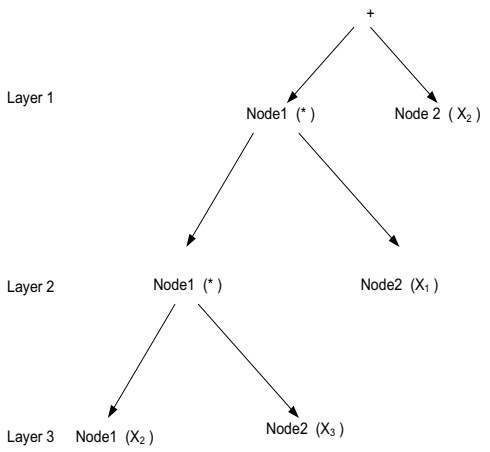

Fig. 2. An example of the tree sturcture of a diferential equation.

A. Genetic Programming

Genetic programming [19] is a type of evolutionary algorithms. All evolutionary algorithms work with a population of individuals, where each individual may be a solution of the optimization problem. GP operates on a tree structure, which is flexible enough to represent relationships efficiently. The leaves of a tree represent variables or constants, while the other nodes implement operators. An example of a tree structure is shown in Fig. 2, where two operations, multiplication (*) and addition (+), are used. The corresponding equation is $\frac{dx_1}{dt} = x_2 + x_1 * x_2 * x_3$. Mutation and crossover operations may be performed to generate offsprings. Selection of better performing individuals (with smaller fitness value, thus minimizing identification error while favoring the simplest model structure) ensures that the population evolves towards solving the optimization problem.

B. H_∞ Filter

The development of efficient linear estimation algorithms, e.g., the Kalman filter, has been based mainly on the minimization of the L_2 -norm of the corresponding estimation error. This type of estimation assumes that the message generating process has a known dynamics and that the exogenous inputs have known statistical properties. The well-known Kalman filter offers optimal filtering algorithm when the system model parameters and power spectral density of the noise are known [20]. However, these assumptions may limit the application of the Kalman filter, because in many situations, only approximate signal models are available and/or the statistics of the noise sources are not fully known or are unavailable. Furthermore, Kalman filter may not be robust against parameter uncertainty of the models [20].

Recent developments in robust estimation have focused on the H_∞ filter [14], [15], [16]. The H_∞ filter is designed to guarantee that the operator relating the noise signals to the resulting estimation errors should possess an H_∞ norm less than a prescribed positive value (the noise suppression level) [18]. In the H_∞ filtering, the noise sources can be arbitrary signals with only a requirement of bounded noise. Since the H_∞ filtering involves the minimization of the worst possible amplification of the error signal, the goal of the H_∞ filter is to provide an uniformly small estimation error for any processes and measurement noises and any initial states. It is shown that the H_∞ filter is more robust compared with Kalman filter in terms of model uncertainty and gives better estimates, for example, in speech processing [17].

Let the L_i-dimensional vector $w(n)$ denotes the state of the system (parameters to be estimated), the process equation and the measurement equation are

$$w(n) = w(n-1) + \mu(n-1) \tag{4}$$

$$d(n) = C(n)w(n) + \nu(n) \tag{5}$$

where $d(n)$ can be calculated as $d(n) = \frac{x(n+1)-x(n)}{\Delta t}$. C contains all the modules, i.e., $C_i = [\Omega_{i1}\ \Omega_{i2}\ \cdots\ \Omega_{iL_i}]$.

Compared to that the Kalman filter minimizes the variance of the estimation error, H_∞ filter is to provide an uniformly small estimation error $e(n) = w(n) - \hat{w}(n)$ with any process and measurement noise. The cost function is given as

$$J = \frac{\sum_{n=1}^{M} \parallel w(n) - \hat{w}(n) \parallel^2_{S(n)}}{\parallel w(0) - \hat{w}(0) \parallel^2_{P(0)^{-1}} + \sum_{n=1}^{M} (\parallel \mu(n) \parallel^2_{Q_1^{-1}(n)} + \parallel \nu(n) \parallel^2_{Q_2^{-1}(n)})}$$

where $P(0), Q_1^{-1}, Q_2^{-1}$and $S(n)$ are positive definite symmetric matrices chosen by the designer based on the performance requirement.

The cost function should be less than a prescribed level, $\frac{1}{\theta}$, i.e.,

$$\sup J < \frac{1}{\theta}$$

where $\sup$ stands for least upper bound.

The implementation of the H_∞ filter is given by the following equations

$$Z(n) = I - \theta S(n)P(n) + C^T(n)Q_2(n)^{-1}C(n)P(n) \tag{6}$$

$$K(n) = P(n)Z(n)^{-1}C^T(n)Q_2(n)^{-1} \tag{7}$$

$$\hat{w}(n+1) = \hat{w}(n) + K(n)(d(n) - C(n)\hat{w}(n)) \tag{8}$$

$$P(n+1) = P(n)Z(n)^{-1} + Q_1(n) \tag{9}$$

Note that although the GRN model itself is nonlinear, the parameters estimation problem is linear given the time-series data.

III. Simulation Evaluation

In the simulation study, the proposed scheme (GP plus H_∞ filter) is compared with the approach in [21], where GP and Kalman filter were combined to deduce the differential equation model and the noise statistics are assumed to be known. Here, H_∞ filter is used to get better estimates when

noise statistics are unknown. We also apply our method to the real measurement data from microarray experiment.

A. Synthetic Data

In this part of the simulation, we use data of a metabolic network, called the E-cell system (a part of the biological phospholipid pathway), that consists of three substances. This network can be approximated as:

$$\begin{aligned} \dot{x_1} &= -10.32 x_1 x_3 \\ \dot{x_2} &= 9.72 x_1 x_3 - 17.5 x_2 \\ \dot{x_3} &= -9.7 x_1 x_3 + 17.5 x_2 \end{aligned} \tag{10}$$

Here, we apply Runge-Kutta method to calculate the synthetic data and add the intrinsic noise and the external noise (both are assumed to be Gaussian white noise).

Since there are three substances in the E-cell system, the tree structure should include a subset of the following terms on the right-hand side of the differential equation: x_1, x_2, x_3, x_1x_2, x_1x_3, x_2x_3, x_1^2, x_2^2, x_3^2. 1000 individuals are first produced and ranked according to the fitness value. 5% of the individuals with the minimum fitness value are kept for the next generation. 80% individuals are performed crossover and 10% individuals are performed mutation and the remaining 5% are for other operations.

We compare our algorithm with the approach in [21] for two different cases. In case 1, the noise covariance is assumed to be known. The covariance matrices are $Q_1 = 10$, $Q_2 = \begin{bmatrix} 0.2 & 0 \\ 0 & 0.2 \end{bmatrix}$, $Q_3 = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$, $R_1 = 20$, $R_2 = 0.2$, $R_3 = 0.01$. It is assumed that μ_{ij} and ν_i are uncorrelated for all i and j. While in case 2, instead of fixed covariance matrices, it is assumed that the covariance matrices are not known exactly, that is, the covariance matrices of μ_{ij} and ν_i are $\widetilde{Q_i} = (1+q_i)Q_i$ and $\widetilde{R_i} = (1+r_i)R_i$, where q_i and r_i are random variables with $E[q_i] = E[r_i] = 0$ and $E[q_i^2] = \sigma_{q_i}^2$, $E[r_i^2] = \sigma_{r_i}^2$. Variances are given by $\sigma_{q_1}^2 = 10$, $\sigma_{q_2}^2 = 0.2$, $\sigma_{q_3}^2 = 10$, $\sigma_{r_1}^2 = 20$, $\sigma_{r_2}^2 = 0.3$, and $\sigma_{r_3}^2 = 15$. The random variables q_i and r_i are uncorrelated for all i.

The results are summarized in Table I. It is observed that GP plus Kalman filter performs very well when the noise covariance is known. However, GP plus H_∞ filter outperforms GP plus Kalman filter when the noise covariance is not known exactly. This is also confirmed by the time series shown in Fig. 3.

The coefficients in the E-Cell model are determined by H_∞ filtering. The convergence of the H_∞ filtering algorithm is an important issue when applying H_∞ filter to the noisy inputs. The convergence of the H_∞ filter includes the convergence of the estimate $\hat{w}(n)$ and the convergence of the estimation error $\hat{e}(n)$. Fig. 4 shows the convergence of the H_∞ filter.

B. Microarray Data

We consider time-series gene-expression data corresponding to yeast protein synthesis. Here, the data for 3 genes (HAP1, CYC7 and CYB2) are picked because the relations among them have been revealed by biological experiments. The trace of the time-series microarray measurement data from [22] is used in this part of the simulation, where 17 sampling data points are provided for each gene by the experiments. The sampling data points are evenly spaced and the observation interval is 10 minutes. In the simulation, 1000 individuals are produced in each generation. 100 generations are calculated to reach the minimum fitness values.

The following model is obtained by the proposed algorithm:

$$\begin{aligned} \dot{x_1} &= (-0.006 + \mu_{11})x_1 + (0.00835 + \mu_{12})x_2 + \nu_1 \\ \dot{x_2} &= (-0.3661 + \mu_{21})x_1 + (-0.476 + \mu_{22})x_2 + \nu_2 \\ \dot{x_3} &= (-1.6124 + \mu_{31})x_1 + (0.45 + \mu_{32})x_2 + \nu_3 \end{aligned} \tag{11}$$

The trajectories for CYB2, HAP1 and CYC7 are shown in Fig. 5.

The obtained relationships among genes are in agreement with biological experimental findings. For example, we observe that HAP1 represses gene CYC7 and CYB2 activates CYC7. HAP1 behaves as a repressor [23].

	true parameter values	GP+KL w/ exact covariance	GP+KL w/ uncertain covariance	GP+H_∞ w/ uncertain covariance
w_{11}	-10.32	-10.34	-7.12	-9.83
w_{21}	9.72	8.87	8.264	9.17
w_{22}	-17.5	-17.42	-16.14	-16.84
w_{31}	-9.72	-9.74	-10.43	-9.03
w_{32}	17.5	17.15	19.78	17.14

TABLE I
THE OBTAINED PARAMETERS BY GP+KF AND GP+H_∞ WHEN NOISE PRESENTS

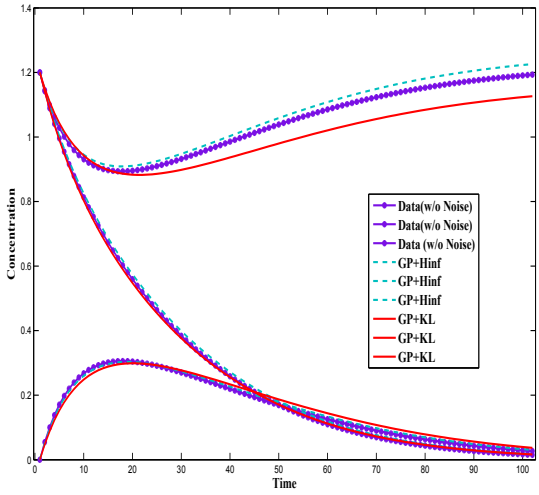

Fig. 3. Time Series for E-Cell simulation by Kalman and H_∞ filtering. "data" is the original data without noise. "GP+KL" and "GP+Hinf" are the simulation data from GP plus Kalman filter and GP plus H_∞ filter, respectively.

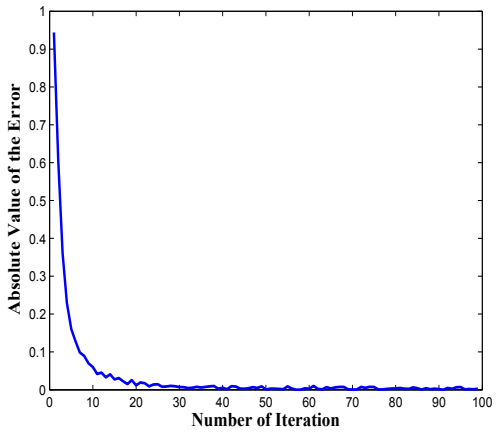

Fig. 4. Experimental learning curves of the E-cell model by H_∞ filtering

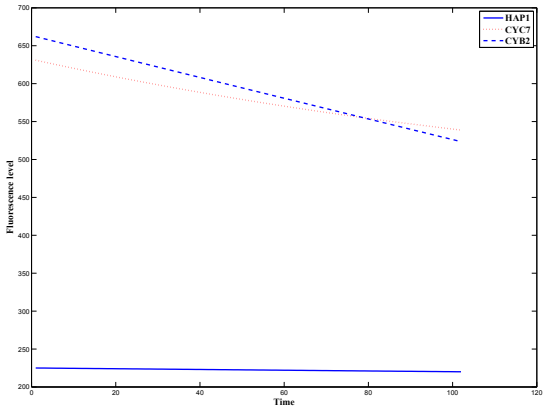

Fig. 5. The trajectories for CYB2, HAP1 and CYC7.

IV. Conclusions

The microarray measurements usually contain rather large noise and the statistics of the noise are not known exactly. In this study, noise is modeled explicitly in the proposed nonlinear ODE model of the GRN. A joint GP and H_∞ filtering approach is applied to infer the GRN, where H_∞ filtering provides optimal parameter estimations under uncertainties. Simulation results demonstrate the effectiveness of the proposed scheme.

References

[1] T. Chen, H.L. He, and G.M. Church, "Modeling gene expression with differential equations", *Pac. Symp. Biocomputing*, 4:29 - 40, 1999.

[2] M.K.S. Yeung, J. Tegnăr, and J.J. Collins, "Reverse engineering gene networks using singular value decomposition and robust regression", *Proc. Natl. Acad. Sci. USA*, 99:6163 - 6168, 2002.

[3] D.C. Weaver, C.T. Workman, G.D. Stormo, "Modeling regulatory networks with weight matrices", *Pac. Symp. Biocomputing*, 4: 112 - 123, 1999.

[4] P. D'haeseleer, X. Wen, S. Fuhrman, and R. Somogyi, "Linear Modeling of mRNA expression levels during CNS development and injury", *Pac. Symp. Biocomputing*, 4: 41 - 52, 1999.

[5] E. Mjolsness, D.H. Sharp, and J. Reinitz, "A connectionist model of development", *J Theor Biol.*, 152(4):429 - 53, Oct 1991.

[6] H.de Jong, "Modeling and simulation of genetic regulatory systems: a literature review", *Journal of Computational Biology*, 9(1):67 - 103, 2002.

[7] I. Tabus, C.D. Giurcaneanu, and J. Astola, "Genetic networks inferred from time series of gene expression data", *First International Symposium on Control, Communications and Signal Processing*, pp. 755 - 758, Hammamet, Tunisia, 2004.

[8] M.J.L. de Hoon, S. Imoto, K. Kobayashi, N. Ogasawara, and S. Miyano, "Inferring gene regulatory networks from time-ordered gene expression data of Bacillus subtilis using differential equations", *Pac. Symp. Biocomputing*, 8: 17 - 28, 2003.

[9] O. Nelles, *Nonlinear System Identification*, Springer, 2001.

[10] S. Ando, E. Sakamoto and H. Iba, "Evolutionary Modeling and Inference of Gene Network", *Information Science*, Vol.145, pp.237 - 259, 2002.

[11] T. Kepler and T. Elston, "Stochasticity in Transcriptional Regulation: Origins, Consequences, and Mathematical Representations", *Biophys J*, Vol. 81, No. 6, pp. 3116 - 3136, Dec 2001.

[12] P. Swain, M. Elowitz, and E. Siggia, "Intrinsic and extrinsic contributions to stochasticity in gene expression", *Proc. Natl. Acad. Sci. USA*, 99:12795 - 12800, 2002.

[13] J. Hasty, J. Pradines, M. Dolnik, and J. J. Collins, "Noise-based switches and amplifiers for gene expression", *Proc. Natl. Acad. Sci. USA*, 97:2075 - 2080, 2000.

[14] U. Shaked and Y. Theodor, "H_∞ optimal estimation: A tutorial", *in Proc. 31st IEEE CDC*, pp. 2278 - 2286, 1992.

[15] B. Hassibi and T. Kailath, "H_∞ adaptive filtering", *in Proc. IEEE ICASSP95*, Detroit, MI, pp. 949 - 952, 1995.

[16] X. Shen and L. Deng, "Game theory approach to discrete H_∞ filter design", *IEEE Transactions on Signal Processing*, Vol. 45, No. 4, pp. 1092-1095, April 1997.

[17] X. Shen and L. Deng, "A dynamic system approach to speech enhancement using H-infinity filtering algorithm", *IEEE Transactions on Speech and Audio Processing*, Vol. 7, pp. 391-399, 1998.

[18] D. Simon, *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*, Wiley, 2006.

[19] J.R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, MIT press, 1992.

[20] M. Grewal and A. Andrews, *Kalman Filtering: Theory and Practice*, Prentice Hall, 1993.

[21] H. Wang, L. Qian, and E. Dougherty, "Inference of Gene Regulatory Networks using Genetic Programming and Kalman Filter", *Gensips*, 2006.

[22] http://sgdlite.princeton.edu/download/yeast_datasets

[23] P. Woolf and Y. Wang, "A fuzzy logic approach to analyzing gene expression data", *Physiol. Genomics*, 3: 9-15, 2000.