Inferring S-system Models of Genetic Networks from a Time-series Real Data Set of Gene Expression Profiles

Hui-Ling Huang, Kuan-Wei Chen, Shinn-Jang Ho, and Shinn-Ying Ho

Abstract—**It is desirable to infer cellular dynamic regulation networks from gene expression profiles to discover more delicate and substantial functions in molecular biology, biochemistry, bioengineering, and pharmaceutics. The S-system model is suitable to characterize biochemical network systems and capable of analyzing the regulatory system dynamics. To cope with the problem "multiplicity of solutions", a sufficient amount of data sets of time-series gene expression profiles were often used. An efficient newly-developed method iTEA was proposed to effectively obtain S-system models from a large number (e.g., 15) of simulated data sets with/without noise. In this study, we propose an extended optimization method (named iTEAP) based on iTEA to infer the S-system models of genetic networks from a time-series real data set of gene expression profiles (using SOS DNA microarray data in *E. coli* as an example). The algorithm iTEAP generated additionally multiple data sets of gene expression profiles by perturbing the given data set. The results reveal that 1) iTEAP can obtain S-system models with high-quality profiles to best fit the observed profiles; 2) the performance of using multiple data sets is better than that of using a single data set in terms of solution quality, and 3) the effectiveness of iTEAP using a single data set is close to that of iTEA using two real data sets.**

I. ITRODUCTION

Living cells contain several levels of networks, such as genetic networks, protein-protein networks and metabolic pathway. Advancements in technologies such as DNA microarrays now allow us to measure gene expression patterns on a genomic scale [1]. By using gene arrays in a time series paradigm, we are able to observe the emergence of coherent temporal responses of many interacting components. In order to understand the regulation of cells, time series expression profiles provide a more complete picture than single time point expression profile [2]. The inference of genetic networks is a problem in which mutual interactions among genes are estimated using time-series data of gene expression patterns.

Manuscript received March 1, 2008. This work was supported in part by the National Science Council of Taiwan for financially supporting this research under the contract numbers NSC 96-2628-E-009-141-MY3 and NSC 96-2627-B-009-002.

H.-L. Huang is with the Department of Information Management, Jin Wen Institute of Technology, Hsin-Tien, Taipei, Taiwan.

K.-W. Chen is with Institute of Bioinformatics, National Chiao Tung University, Hsinchu, Taiwan.

S.-J. Ho is with the Department of Automation Engineering, National Formosa University, Yunlin 632, Taiwan

S.-Y. Ho is with the Department of Biological Science and Technology, and Institute of Bioinformatics, National Chiao Tung University, Hsinchu, Taiwan (corresponding author to provide e-mail: syho@mail.nctu.edu.tw).

Given a dynamic model of gene interaction, the problem of gene network inference is equivalent to learning the structural and functional parameters from time series representing the gene expression kinetics [3].

Noman and Iba [4] employed Trigonometric Differential Evolution as the optimization engine algorithm for capturing the decoupled S-system formalism. Using an S-system based model for the transcription and translation process, Thomas et al. [5] proposed a heuristic method for resolving networks. A hybrid method of Genetic Programming and Least Mean Square method [6] were combined to identify concise form of regulation between S-system based models. Yamanaka *et al.* [7] used toxic genomics for identifying gene interaction networks with Bayesian model selection. Xu *et al.* [8] proposed a recurrent neural network and particle swarm optimization approach to infer genetic regulatory networks. Ho *et al.* [9] proposed an intelligent two-stage evolutionary algorithm (iTEA) to efficiently infer the S-system models of genetic networks from a number of time-series data sets of gene expression profiles.

The S-system model [10] of gene networks is based on the Biochemical System Theory (BST) – a generalized framework for modeling and analyzing biological systems [11, 12]. S-system is a dynamic model for biochemical pathways, having a good compromise between accuracy and mathematical flexibility. The model is a set of non-linear differential equations of the following form [4-6]:

$$\frac{dX_i}{dt} = \alpha_i \prod_{j=1}^{N} X_j^{g_{ij}} - \beta_i \prod_{j=1}^{N} X_j^{h_{ij}}, \quad i=1, \ldots, N, \qquad (1)$$

where X_i represents the expression level of gene i and N is the number of genes in a genetic network. α_i and β_i are rate constants which indicate the direction of mass flow and must be positive. g_{ij} and h_{ij} are kinetic orders which reflect the intensity of interaction from gene j to i. For inferring an S-system model, it is necessary to estimate all the $2N(N+1)$ S-system parameters (α_i, β_i, g_{ij}, h_{ij}) from experimental time-series data of gene expression.

In this paper, we propose an effective optimization method (named iTEAP) based on iTEA to infer the S-system models of genetic networks from a time-series real data set of gene expression profiles (using SOS DAN microarray data in *E. coli* as an example). The algorithm iTEAP generated additionally multiple data sets of gene expression profiles by perturbing the given data set. High performance of iTEAP arises mainly from 1) an intelligent genetic algorithm (IGA) of iTEA, and 2) the technique of generating perturbed data sets to effectively confine the search space of candidate

978-1-4244-1823-7/08/$25.00©2008 IEEE

solutions.

The effectiveness of iTEA was evaluated using simulated expression patterns with and without noise [9]. It had been shown that: 1) IGA is efficient enough to solve subproblems; 2) IGA is significantly superior to the existing method SPXGA [13] in solving subproblems, and 3) iTEA performs well in inferring S-system models of genetic networks from small-noise gene expression profiles.

The above-mentioned methods often detect important patterns, but cannot definitively identify the targets of transcriptional regulators. Moreover, because of high costs of the experiments and due to the fact that the investigated processes are too short and do not allow for more sampling points in time. To solve this problem of "multiplicity of solutions" [14], additional data sets have to be acquired like knock-out, over-expression experiment data or data sets with different starting conditions that decrease the uncertainties.

From a given real data set of gene expression profiles only, it is desirable to infer accurate S-system models of genetic networks by coping with the problem of "multiplicity of solutions". The proposed iTEAP aims to improve the solutions of iTEA to reconstruct the regulatory pathway by generating a number of perturbed data sets for each of real experimental data. Then, we extend the capabilities of iTEA to predict regulatory pathways by applying the S-system network to time series gene expression data from SOS DAN microarray data in *E. coli.* The results of iTEAP on the SOS DAN microarray data in *E. coli* reveal that 1) the S-system models obtained by iTEAP can generate high-quality profiles to best fit the experimental profiles; 2) the performance of using multiple data sets is better than that of using a single data set in terms of solution quality, and 3) the effectiveness of iTEAP using a single data set is close to that of iTEA using two data sets.

II. METHOD

A. Data Preprocessing

The genetic network inference problem using S-system model suffers from many difficulties, such as high degree of freedom, high dimensionality, multimodality, strong interaction among parameters of the S-system model, and measurement noise. Some of above difficulties can be solved using a large number of real experimental profiles. However, conventional experiments generally produce few real experimental data due to limited resource and money. For this reason, a preprocessing technique "adding noisy duplicates" is introduced to cope with the difficulties caused by insufficient real data.

To increasing the number of data, M-1 sets of additional profiles are produced by adding k% random Gaussian noises to each of the real experimental profiles using the following equation,

$$X^{l}_{pseudo,i,t} = X_{obs,i,t} + N\left(0,\sigma^2\right), \qquad (2)$$

where $X_{obs,i,t}$ is the experimental expression level of gene i at time t, and $X^{l}_{pseudo,i,t}$ is the l-th produced pseudo expression data, $N(0, \sigma^2)$ is a normal distributed random number function with zero mean and variance σ^2. Here, σ is assigned as $X_{obs,i,t} \times k\%$.

B. Objective Functions

The optimization problem is first decomposed into N subproblems having $2(N+1)$ parameters. To solve i-th individual subproblem of gene i using IGA and refine the combined solutions using OSA, the corresponding objective function for guiding the searching process is required. At the first stage of iTEA, the following objective function is adopted:

$$\min f_i = \sum_{t=1}^{T}\left(\frac{X_{\inf,i,t} - X_{obs,i,t}}{X_{obs,i,t}}\right)^2 + c\sum_{j=1}^{N-I}\left(\left|G_{ij}\right| + \left|H_{ij}\right|\right), \quad (3)$$

where $X_{inf,i,t}$ is an inferred expression level of gene i at time t, T is the number of sampling points of observed data, c is a penalty weight, I is a maximum indegree that the maximal number of genes which directly affect gene i. G_{ij} and H_{ij} are given by rearrange g_{ij} and h_{ij} in ascending order of their absolute values.

However, the objective function (3) may be biased, especially when the experimental expression level is very small. For this reason, the objective function is modified using the absolute error instead of the relative error:

$$\min f_i = \sum_{t=1}^{T}\left(X_{\inf,i,t} - X_{obs,i,t}\right)^2 + c\sum_{j=1}^{N-I}\left(\left|G_{ij}\right| + \left|H_{ij}\right|\right), \qquad (4)$$

To obtain a robust solution, the objective function used in iTEAP is modified by adding a penalty term SD_i:

$$\min F_i = \frac{1}{M}\sum_{m=1}^{M}\sum_{t=1}^{T}\left(X^{m}_{\inf,i,t} - X^{m}_{obs,i,t}\right)^2 + c_1 SD_i + c_2\sum_{j=1}^{N-I}\left(\left|G_{ij}\right| + \left|H_{ij}\right|\right), \qquad (5)$$

where M is the number of profiles of gene i, SD_i is the standard deviation of the squared errors for M sets of profiles of gene i, c_1 and c_2 are penalty weights, and $X^{m}_{\inf,i,t}$ and $X^{m}_{obs,i,t}$ are the inferred and observed experimental expression levels of gene i at time t using m-th profile, respectively.

For the same reason above, the objective function of OSA at the second stage is modified as follows:

$$\min F = \frac{1}{M}\sum_{i=1}^{N}\sum_{m=1}^{M}\sum_{t=1}^{T}\left(X^{m}_{\inf,i,t} - X^{m}_{obs,i,t}\right)^2 + c_1\sum_{i=1}^{N} SD_i + c_2\sum_{i=1}^{N}\sum_{j=1}^{N-I}\left(\left|G_{ij}\right| + \left|H_{ij}\right|\right). \qquad (6)$$

C. iTEAP for Reconstructing Genetic Networks

Besides dealing with the problem of insufficient experimental data, reconstructing the genetic network still needs an effective method to solve such a large-scale optimization problem. In this paper, an intelligent two-stage evolutionary algorithm (iTEA) [9] is adopted, which is able to infer the S-system models of genetic networks from a large

number of simulated small-noise gene expression profiles with different starting conditions efficiently. In the following paragraphs, a brief introduction of iTEA is presented.

To handle the curse of dimensionality, iTEA uses a divide-and-conquer strategy [15] to decompose the optimization problem into N subproblems having $2(N+1)$ parameters, and utilize the structure skeletalizing techniques [16] to reduce computation cost in both stages. At first stage, each subproblem is solved by using an intelligent genetic algorithm (IGA) [17] which introduces an intelligent crossover operation based on orthogonal experimental design (OED) [18, 19]. At second stage, the obtained N solutions to N subproblems are combined and refined using an orthogonal simulated annealing algorithm (OSA) [20] which incorporates an intelligent generation mechanism (IGM) based on OED. Both IGA and OSA are based on orthogonal experimental design to speed up the search by using a systematic reasoning method instead of the conventional random generation method. The detail description of OED-based intelligent crossover and IGM can be found in [9].

1) The Used IGA

The main difference between the used IGA and the conventional genetic algorithms are chromosome encoding, crossover operation, and Cauchy-Lorentz probability distribution based mutation [21] mentioned in [9]. The used IGA is described simply below.

Step 1: (Initialization) Generate an initial population with N_{pop} feasible individuals of $2(N+1)$ real-value parameters randomly.

Step 2: (Evaluation) Evaluate fitness value of all individuals.

Step 3: (Selection) Use the simple ranking selection that replaces the worst $P_s \times N_{pop}$ individuals with the best $P_s \times N_{pop}$ individuals to form a new population, where P_s is a selection probability. Let I_{best} be the best individual in the population.

Step 4: (Crossover) Select $P_c \times N_{pop}$ individuals including I_{best} randomly, where P_c is the crossover probability. Applying intelligent crossover operation on the selected pairs of parents.

Step 5: (Mutation) Besides the best individual, the Cauchy-Lorentz probability distribution based mutation operation is applied on all other individuals according to mutation probability P_m.

Step 6: (Termination test) If the prespecified number N_{eval} of fitness evaluation is achieved or some stopping condition is met, then stop the algorithm. Otherwise, go to Step 2.

2) The Used OSA

With the help of IGM, OSA can search for a good candidate solution efficiently. OSA uses a simple geometric cooling rule by updating the temperature at the $(i+1)$-th temperature step using the formula:

$$Temp_{i+1} = CR \bullet Temp_i, i = 0,1,\ldots, \tag{7}$$

where CR is the cooling rate which is a constant smaller than 1 but close to 1 (e.g., CR=0.999). In the higher temperature, it is more possible to accept the candidate solution worse than current solution. The OSA used in iTEA is described below.

Step 1: (Initialization) Initialize $Temp = Temp_0$ and CR. Set the combined solution S as the initial solution and compute fitness of solution S, $F(S)$.

Step 2: (Perturbation) Perform an IGM operation using S to generate a candidate solution Q.

Step 3: (Acceptance test) Accept Q to be the new solution S with probability $P(Q)$:

$$P(Q) = \begin{cases} 1, & if \ \ F(Q) \le F(S) \\ \exp\left(\dfrac{F(S) - F(Q)}{Temp} \right), & if \ \ F(Q) > F(S) \end{cases} \tag{8}$$

Step 4: (Decreasing temperature) Let the new value of $Temp$ be $CR \cdot Temp$.

Step 5: (Termination test) if a prespecified stopping criterion is met, stop the algorithm. Otherwise, go to Step 2.

3) iTEAP Using IGA and OSA

The proposed algorithm iTEAP uses both IGA and OSA in the Steps 1 and 2, respectively. IGA aims to obtain solutions to subproblems with significant accuracy in terms of the objective function value which can best fit the given gene expression profiles. If noise is very small, IGA is effective enough and the improvement of OSA in Step 2 is not significant. When noise becomes larger, the best fit of the observed gene expression profiles is leaved to OSA from the aspect of global optimization. Fig. 1 illustrates the flowchart of iTEAP based on the two-stage evolutionary algorithm iTEA. The algorithm of iTEAP is given as follows:

Step 1: Using IGA to solve N individual subproblems independently using the objective function (5). For each subproblem, R>1 independent runs are conducted and the best solution of each solution is selected.

Step 2: Combine all N best solutions (α_i, g_{i1},…, g_{iN}, β_i, h_{i1},…, h_{iN}), $i = 1, \ldots, N$ as an initial solution of OSA. Then, applying OSA to refine the solution with the objective function (6).

Step 3: Use the Z-score [24] technique to determine the structure of gene networks from the obtained S-system models.

III. EXPERIMENTAL REAULTS

A. Material

We employed iTEAP to evaluate the SOS DNA repair network in bacterium Escherichia coli depicted in Fig. 2 [22]. The well-known gene network responses for repairing the DNA after damages. The SOS system consisting of around 30 genes regulates at the transcriptional level. The experimental data can be downloaded from the website of Uri Alon Lab [23]. For the data set, four experiments have been conducted with difference light intensities (Exp. 1&2: 5 Jm^{-2}, Exp. 3&4: 20 Jm^{-2}). Each experiment includes the expression measurements for eight major genes through 50 time points, sampled every 6 minutes.

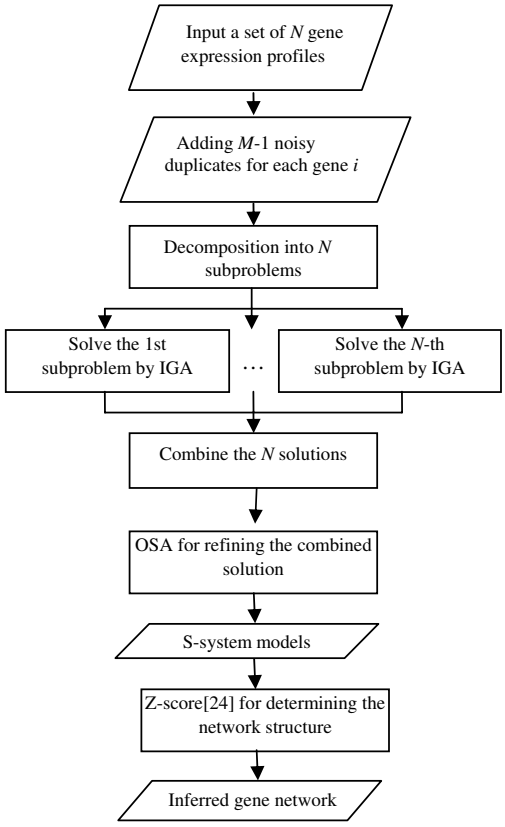


Fig. 1. Flowchart of the proposed method iTEAP

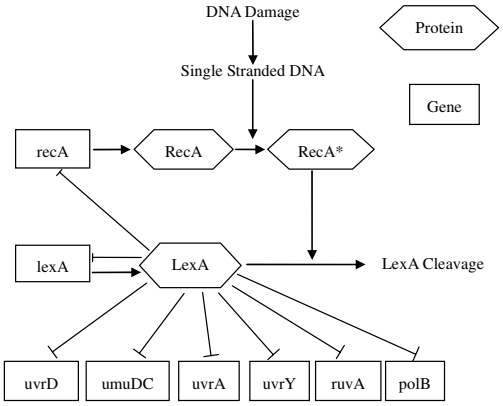


Fig. 2. The bacterial E. coli SOS DNA repair netwok. Activations are represented by arrows (→), while inhibitions are represented by T (⊣). Genes (square) initials are in lower case, proteins (hexagon) in capital letters.

B. Experiment Performance of gPT

We conducted some experiments to evaluate the proposed perturbation approach. The used parameters for iTEAP are as follows: population size = 20, number of OA factors = 7, N (number of gene) = 8, stopping condition: N_{eval} = 7000 x N, T (number of time point) = 50, I (maximum degree of regulation) = 5, c_1 and c_2 (penalty weights) = 1.0, and δ_s (skeletal threshold) = 0.1.

In the experiment, the four types of subsets of all the data from Alon's four experiment data sets (Exp_i, i = 1, …, 4) are list in Table. 1. We generated 14 sets (M=15) as the perturbed real time-series microarray data for each 1-set type data (named p-set), where 2% Gaussian noises was added. For all types with t data sets (t-set), let M=t in the iTEAP. The values of gene expression levels were normalized in the range (0, 1.0] and all the zero expression levels were replaced with a very small value. 30 independent runs were carried out to assure the statistical significance of the probabilistic capture.

The results of total standard deviations of 8 genes from 30 independent runs are shown in Fig. 3. The results real that 1) the more the real time-series data sets were used, the smaller the total standard deviations (i.e., 1-set > 2-set > 3-set > 4-set), and 2) the perturbed data set p14 using only one real time-series data set is better than 1-set data set and comparable to 2-set in terms of robustness quality.

TABLE 1
THE FOUR TYPES OF SUBSETS OF ALL THE DATA FROM ALON'S FOUR EXPERIMENT DATA SETS.

TYPE NAME	THE SELECTED SUBSETS FROM ALON'S DATA SETS
1-set	(Exp_i), i = 1,…,4
2-set	(Exp_i, Exp_j), i, j = 1,…,4, $i \neq j$
3-set	(Exp_i, Exp_j, Exp_k), i, j, k = 1,…,4, $i \neq j \neq k$
4-set	(Exp_1, Exp_2, Exp_3, Exp_4)

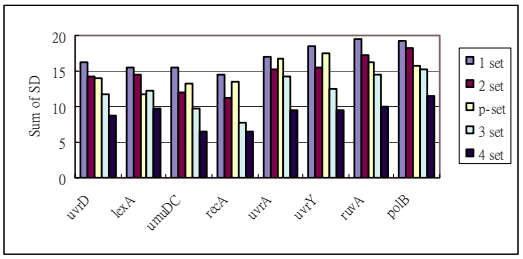


Fig. 3. Sum of standard deviations of 8 genes from 30 independent runs.

We applied iTEAP to estimate the parameters of S-system model. The fitness value of (6) reflecting the fitting quality of time-series data is used to evaluate the ability of the used optimization algorithm. However, the major concern is to obtain a correct network structure with accurate parameter values.

Fig. 4 (a) and (b) shows the given gene expression profiles of the Exp_1 dataset and the estimated profiles with the best fitness value using iTEAP on 1-set (i.e., no perturbed profiles were added). The evolutionary model can effectively capture the dynamics of most of genes in the system, with the major change trends of the gene expression levels reflected in the evolutionary curve. We can see the final inferred profile is very similar to the observed profiles.

The real time-series data sets are noisy and the results were dispersed. We apply the Z-score technique [24] to determine the structure of the gene network using the results of Fig. 4

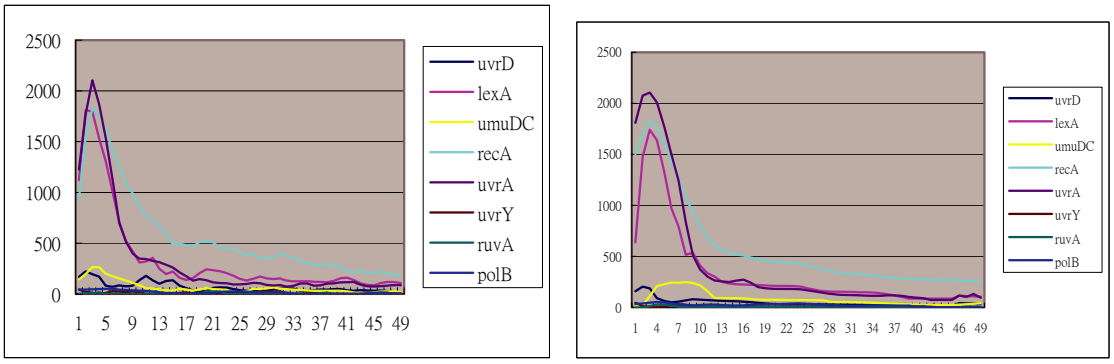
(b). After analyzing the Z-score value with the threshold Z_{th} = 1.0, the gene network is shown in Table 2. The result is not very good because the *lex*A regulations of *lex*A and *rec*A were not identified, which had been identified in these works [4, 8, 22, 25]. It induced some insufficient regulations, such as *pol*B, *uvr*A, and *uvr*Y.

To examine the effectiveness of adding noisy duplicates, Table 3 shows the regulation of genes in SOS repair network identified by iTEAP applied on the Exp_3 data set with perturbed profiles. The result is more promising, compared with the results in Table 2. We can find out the regulations which have been identified in these works [4, 8, 22, 25], such as:

1. Corresponding to inhibition of *lex*A on *umu*DC.
2. The inhibition of *lex*A on *lex*A , *uvr*D, *rec*A, *uvr*A, and *pol*B.
3. *rec*A → *Rec*A ⊣ *Lex*A ⊣ *uvr*A.
4. The activation of *lex*A by *rec*A

IV. CONCLUSIONS

We have proposed an efficient method iTEAP (using the well-known SOS DNA microarray data in *E. coli* as a test example) to infer cellular dynamic regulation networks from gene expression profiles to discover more delicate and substantial functions. The S-system model is suitable to characterize biochemical network systems and capable of analyzing the regulatory system dynamics. To cope with the problem "multiplicity of solutions", iTEAP uses noisy duplicates to obtain a rather accurate and robust solution to the gene reconstruction problem. The simulation results reveal that iTEAP can obtain S-system models with high-quality generated profiles to best fit the observed profiles, and the effectiveness of iTEAP using a single data set is comparable to that of iTEA using two real data sets. The iTEAP is more useful when only one real data set of gene expression profiles is available. Of course, iTEAP also benefits from multiple real data sets for achieving more accurate solutions. Furthermore, the proposed iTEAP can also work efficiently for other biological systems such as reconstruction and analyzes of metabolic pathway.



(a) The given gene expression profiles. (b) The estimated profiles.

Fig.4 The comparison of profiles using iTEAP on the Exp_1 dataset.

TABLE 2.
RECONSTRUCTION OF THE GENE NETWORK USING FIG. 4 (B) BY APPLYING ITEAP ON THE EXP_1 DATASET WITH NO PERTURBED PROFILES.

	Activation								**Inhibition**							
	uvrD	**lexA**	**umuDC**	**recA**	**uvrA**	**uvrY**	**ruvA**	**polB**	**uvrD**	**lexA**	**umuDC**	**recA**	**uvrA**	**uvrY**	**ruvA**	**polB**
uvrD					uvrA	uvrY		polB	uvrD					uvrY		
lexA				recA	uvrA	uvrY		polB			umuDC	recA	uvrA	0	ruvA	polB
umuDC		lexA		recA	uvrA		ruvA	polB		lexA	umuDC	recA			ruvA	polB
recA		lexA			uvrA	uvrY	ruvA	polB			umuDC	recA				
uvrA	uvrD				uvrA	uvrY	ruvA	polB			umuDC	recA	uvrA			polB
uvrY	uvrD			recA		uvrY	ruvA	polB	uvrD				uvrA		ruvA	polB
ruvA	uvrD					uvrY	ruvA		uvrD				uvrA		ruvA	polB
polB	uvrD	lexA		recA	uvrA			polB			umuDC	recA	uvrA		ruvA	polB

TABLE 3.
THE REGULATION OF GENES IN THE SOS REPAIR NETWORK IDENTIFIED BY USING ITEAP ON EXP_3 WITH NOISY DUPLICATES.

	Activation								Inhibition							
	uvrD	**lexA**	**umuDC**	**recA**	**uvrA**	**uvrY**	**ruvA**	**polB**	**uvrD**	**lexA**	**umuDC**	**recA**	**uvrA**	**uvrY**	**ruvA**	**polB**
uvrD		lexA				uvrY		polB	uvrD	lexA			0			
lexA				recA		uvrY				lexA			uvrA			
umuDC	uvrD	lexA	umuDC	recA						lexA	umuDC		0			
recA	uvrD			recA						lexA		recA	0			
uvrA	uvrD			recA					uvrD	lexA			uvrA			
uvrY	uvrD			0					uvrD				0	uvrY		0
ruvA	uvrD			0								recA				polB
polB			umuDC	0			ruvA			lexA						polB

REFERENCES

[1] J. L. Derisi, V. R. Iver and P. O. Brown, "Exploring the metabolic and genetic control of gene expression on a genomic scale," Science, 1997, 278, pp. 680-686.

[2] I. M. Ong, J. D. Glasner, D. Page, " Modelling regulatory pathways in E. coli from time series expression profiles," Bioinformatics, 2002, vol. 18, Suppl 1, pp. S241-S248.

[3] E.Sakamoto, and H.Iba, "Inferring a system of di®erential equations for a gene regulatory network by using genetic programming," Proc. Congress on Evolutionary Computation, 2001, pp. 720-726.

[4] N. Noman and H. Iba, "Reverse engineering genetic networks using evolutionary computation," Genome Informatics, 2005, 16(2), pp. 205–214.

[5] R. Thomas, C. J Paredes, S. Mehrotra, V. Hatzimanikatis, and E. T Papoutsakis, "A model-based optimization framework for the inference of regulatory interactions using time-course DNA microarray expression data," BMC Bioinformatics, 2007, v.8, 228.

[6] S. Ando, E. Sakamoto, H. Iba,"Evolutionary modeling and inference of gene network," Information Sciences, 2002, 145, pp. 237–259.

[7] T. Yamanaka, *et al.*, "The TAO-Gen Algorithm for Identifying Gene Interaction Networks with Application to SOS Repair in E. coli," Environmental Health Perspectives, 2004, vol. 112, Num. 16.

[8] R. Xu, D. C. Wunsch II, and R. L. Frank, "Inference of Genetic Regulatory Networks with Recurrent Neural Network Models Using Particle Swarm Optimization," IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB), 2007, vol. 4, no. 4. pp. 681-692.

[9] S.-Y. Ho, C.-H. Hsieh, F.-C. Yu, and H.-L. Huang, "An intelligent two-stage evolutionary algorithm for dynamic pathway identification from gene expression profiles," IEEE/ACM Trans. Computational Biology and Bioinformatics (TCBB), 2007, vol. 4, Issue 4, pp. 648-704.

[10] N. Noman and H. Iba, "Inference of genetic networks using S-system: information criteria for model selection," GECCO' 06July 8-12, 2006, Seattle, Washington, USA.

[11] M. Savageau, "Biochemical systems analysis. i. some mathematical properties of the rate law for the component enzymatic reactions," Theoretical Biology, 1969, 25, PP. 365-369.

[12] M. Savageau, "Biochemical systems analysis. ii. the steady-state solutions for an n-pool system using a power-law approximation," Theoretical Biology, 1969, 25, pp. 370-379.

[13] S. Kikuchi, D. Tominaga, M. Arita, K. Takahashi, and M. Tomita, "Dynamic modeling of genetic networks using genetic algorithm and S-system," Bioinformatics, 2003, vol. 19, pp. 643-650.

[14] S. G Tringe, A. Wagner, and S. W Ruby , "Enriching for direct regulatory targets in perturbed gene-expression profiles," Genome Biol., 2004, 5(4), R29.

[15] Y. Maki, T. Ueda, M. Okamoto, N. Uematsu, K. Inamura, K. Uchida, Y. Takahashi, and Y. Eguchi, "Inference of genetic network using the expression profile time course data of mouse P19 cells," *Genome Informatics*, 2002, vol. 13, pp. 382-383.

[16] D. Tominaga, N. Koga, and M. Okamoto, "Efficient numerical optimization algorithm based on genetic algorithm for inverse problem," in *Proc. of Genetic and Evolutionary Computation Conference*, Las Vegas, Nevada, USA, 2000, pp. 251-258.

[17] S.-Y. Ho, L.-S. Shu, and J.-H. Chen, "Intelligent evolutionary algorithms for large parameter optimization problems," *IEEE Trans. Evolutionary Computation* 2004, vol. 8, no. 6, pp. 522-541.

[18] T. P. Bagchi, *Taguchi Methods Explained: Practical Steps to Robust Design*: Prentice-Hall, 1993.

[19] A. S. Hedayat, N. J. A. Sloane, and J. Stufken, *Orthogonal Arrays: Theory and Applications*. New York: Springer-Verlag, 1999.

[20] S.-J. Ho, S.-Y. Ho, and L.-S. Shu, "OSA: Orthogonal simulated annealing algorithm and its application to designing mixed H-2/H-infinity optimal controllers," *IEEE Trans. Systems, Man, and Cybernetics-Part A: Systems and Humans*, 2004, vol. 34, no. 5, pp. 588-600.

[21] H. Szu and R. Hartley, "Fast simulated annealing," *Physics Letters*, 1987, vol. 122, pp. 157-162.

[22] M. Ronen, R. Rosenberg, B. Shraiman, and U. Alon, "Assigning Numbers to the Arrows: Parameterizing a Gene RegulationNetwork by Using Accurate Expression Kinetics," Proc. Nat'l Academy of Sciences USA, 2002, vol. 99, no. 16, pp. 10555-10560.

[23] http://www.weizmann.ac.il/mcb/UriAlon/.

[24] P. D' Haeseller, S. Liang, and R.Somogyi, "Genetic network inference: From co-expression clustering to reverse engineering," Bioinformatics, 2000, 16(8):707-726.

[25] B.-E. Perrin, L. Ralaivola, A. Mazurie, S. Bottani, J. Mallet, and F. d' Alché-Buc, "Gene networks inference using dynamic Bayesian networks," Bioinformatics, 2003, 19(Suppl 2):ii138-ii148.