Comparing Genetic Programming and Evolution Strategies on Inferring Gene Regulatory Networks

Felix Streichert, Hannes Planatscher, Christian Spieth, Holger Ulmer, and Andreas Zell

Centre for Bioinformatics Tübingen (ZBIT), University of Tübingen,
Sand 1, 72076 Tübingen, Germany,
streiche@informatik.uni-tuebingen.de
http://www-ra.informatik.uni-tuebingen.de/

Abstract. In recent years several strategies for inferring gene regulatory networks from observed time series data of gene expression have been suggested based on Evolutionary Algorithms. But often only few problem instances are investigated and the proposed strategies are rarely compared to alternative strategies. In this paper we compare Evolution Strategies and Genetic Programming with respect to their performance on multiple problem instances with varying parameters. We show that single problem instances are not sufficient to prove the effectiveness of a given strategy and that the Genetic Programming approach is less prone to varying instances than the Evolution Strategy.

1 Introduction

In recent years modern technologies like microarrays allowed scientists to measure large numbers of gene expression data for thousands of genes at the same time. With this technique at hand scientists are also able to measure gene activities through time. Such time series nourish the idea that it could be possible to reconstruct or infer the underlying gene regulatory networks. This problem of inferring the real gene regulatory networks from time series data has recently become one of the major topics in bioinformatics.

The strategies for inferring regulatory networks depend on the mathematical model used to represent the behavior of the real gene regulatory network. Currently, both discrete and continuous models are used to model regulatory networks, but to represent the activity of real regulatory networks continuous models are believed to be the most suitable.

For discrete models like boolean or random boolean networks [17], several efficient heuristics have been suggested [1]. More realistic models are given by qualitative networks, which use several levels of activation rather than just 'on' or 'off' [12]. For qualitative networks, Akutsu et al. have suggested a special heuristic for inferring such networks from time series data [2].

Quantitative networks on the other hand consider the continuous level of gene expression and are therefore more realistic. A parametrized model with discrete

K. Deb et al. (Eds.): GECCO 2004, LNCS 3102, pp. 471–480, 2004.

time and a linear relationship between the genes is given by weight matrices [16]. The parameters for such weight matrices have been reverse engineered by means of Genetic Algorithms (GA) [15]. Other researchers use linear differential equations to model regulatory networks and use special heuristics to find the necessary parameters [3]. Another parameterized model based on differential equations is given by S-systems (*synergistic* and *saturable* systems) [9]. To infer the unknown parameters of this model Tominaga et al. applied a GA with special operators biased towards few non-zero parameters leading to sparsely connected regulatory networks [14].

An example for non-parameterized quantitative networks are arbitrary systems of differential equations, which are more powerful and flexible to describe the relations between genes, since the real structure underlying the observed data is unknown. The most prominent method that is able to optimize the structure and the parameters of differential equations to fit a given time series is Genetic Programming (GP) [5]. Sakamoto et al. applied a GP augmented with a least mean square method for parameter optimization for inferring differential equations for regulatory networks [8].

In this paper we compare two inferring strategies for quantitative networks based on Evolutionary Algorithms (EA). On the one hand to fix the network model *a priori* and reduce the inferring problem to a parameter optimization problem, which can be solved by means of Evolution Strategies (ES). We decided to use S-systems as a parameterized quantitative network, since they derive from a Taylor approximation of a general ordinary differential equation and are rather flexible. And on the other hand to leave the choice of the network structure to the inferring algorithm. This non-parameterized network model requires GP for inferring a suitable structure to met the target.

We compare both approaches on several examples generated from artificial data to determine, which approach is more suitable for inferring gene regulatory networks. We also try to identify, which are the most important properties of a regulatory network that make it difficult to reconstruct from time series data. Therefore, we vary multiple parameters of the artificial network and examine how the changes impact the performance.

In sec. 2 we give details on the experimental settings and our implementation of the optimization algorithms used. ES and GP are then compared in sec. 3 on several examples generated from artificial regulatory networks. Conclusions and an outlook of future research are given in sec. 4.

2 Experimental Settings

Since there are only few publicly available time series for gene expression and the correct or best model is not known for those regulatory networks, a comparison of inferring strategies cannot rely on real data. Therefore, it is necessary to create time series from artificial regulatory networks as benchmark problems. In this way one is in control of all properties of the target network and can arbitrarily vary the dimension of the problem and the connectivity of the network. One may also validate the results obtained by comparing it to the artificial target.

Unfortunately, it is unknown, which kind of model is most suitable to represent the same dynamical properties as real regulatory networks. Also there are currently artificial benchmark problems for inferring regulatory network that researchers commonly agree on. Only recently Mendes et al. proposed a set of benchmark networks [6], which were published after our experiments were conducted, but will be included in our future studies.

In this paper we examine S-systems as an example for parameterized quantitative networks as tentative benchmark problem, since they were derived from a simplified Taylor expansion of a general ordinary differential equation and should be rather flexible. An S-system for n artificial genes is given by a parameterized set of nonlinear differential equations:

$$\frac{dx_i(t)}{dt} = \alpha_i \prod_{j=1}^{n} x_j(t)^{\mathcal{G}_{i,j}} - \beta_i \prod_{j=1}^{n} x_j(t)^{\mathcal{H}_{i,j}} \tag{1}$$

where x_i is the state variable of the measured expression level of gene i. With $\alpha_i \geq 0$ and $\beta_i \geq 0$ the first product describes all synthesizing influences and the second product all degrading influences. Depending on the values of $\mathcal{G}_{i,j}$ and $\mathcal{H}_{i,j}$ the influence may be inhibitory, if the value in the matrix is smaller than zero, or excitatory, if greater than zero.

With this model for an artificial gene regulatory network we can generate example problems by choosing random values for α_i, β_i, $\mathcal{G}_{ij}$ and $\mathcal{H}_{ij}$ while checking whether they are stable or not. We can increase the dimensionality of the regulatory network by increasing n and we can change the level of interdependence between the genes by adding or removing zero-valued parameters.

In our experiments we generate a problem instance by simulating the target artificial regulatory network and store the calculated expression values at certain time points. This corresponds to real experiments where the number of microarrays and therefore the number of actually measured time points is limited.

We compare the performance of an EA approach based on a parameterized S-system using Evolution Strategies (ES) to identify suitable parameters to fit the measured time series to a Genetic Programming (GP) approach to search for the proper right hand side of a system of ordinary differential equations. For both EA methods the fitness f of an individual a is given by the Relative Standard Error (RSE) of the resulting estimation of gene expression $\hat{x}$ to the measured gene expression x:

$$f = \sum_{i=1}^{N} \sum_{k=1}^{T} \left(\frac{\hat{x}_i(t_k) - x_i(t_k)}{x_i(t_k)} \right)^2 \tag{2}$$

over all measured time points t_j over all genes n.

This fitness function is very much straight forward and commonly used in this area of research. But it suffers from a serious problem: a single measured time series is not sufficient to identify a unique solution. It is only one path in a phase diagram and from such a single path no general conclusions of the overall behavior of the dynamic system can be drawn. An extreme example is given in the phase diagram of a simple two dimensional example in fig. 2. The two genes

x_1 and x_2 oscillate on a stable orbit (attractor). With a single time series given, whose starting conditions are outside the orbit, no predictions can be made about the dynamics inside the orbit. Therefore, there are multiple alternative solutions, which can not be distinguished without additional problem specific knowledge. The problem of ambiguity is already known in the literature, but is rarely addressed. A common approach is to take the mean of multiple runs [15]. Others prefer sparsely connected networks over fully connected networks to identify the simplest solution [13]. A different approach was used by Morishita et al. [7], they solved the problem of ambiguity by searching for as many candidate solutions as possible. Unfortunately, they do not address the question of how to identify the correct one from more than two hundred candidate solutions they found for a five-dimensional artificial problem.

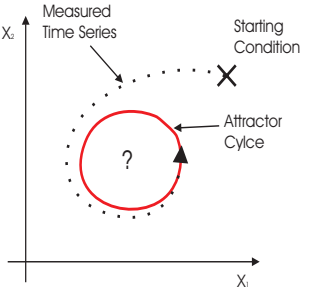

Fig. 1. Phase diagram

An alternative approach solves the ambiguity by requesting additional experimental data [11]. Similar to the approach of Morishita et al. they start searching for multiple candidate solutions through a multi-start EA. If the candidate solutions are significantly different from each other they request an additional experiment. The suggested experiment is selected through *in silico* simulations to reduce as much ambiguity as possible. This process is iteratively repeated until all multi-start EA runs converge on the same solution.

Currently we will not address the problem of ambiguity in this paper. We will simply examine which EA approach is more efficient to identify a possible candidate solution using equ. 2 as fitness function, but our future research will apply the approach used in [11], since it is currently the only one that actually removes the ambiguity.

2.1 Implementation Details for Evolution Strategies

Evolution Strategies were introduced by Rechenberg and Schwefel and are based on a real valued encoding of the decision parameters and sophisticated methods for adapting the strategy parameters for the mutation operator [10]. Preliminary experiments showed that the inferring problem for the parameterized network model is multi-modal, deceptive and correlated. Therefore, we decided to use a bigger population size ($\mu = 20, \lambda = 100$). To increase the quality of convergence, we decided to apply the Covariance Matrix Adaption (CMA) mutation operator, which is able to adapt to correlated decision parameters better than standard ES mutation operators [4]. Due to the sensitivity of the CMA to crossover, we omitted the crossover operator from the ES. The ES was performed for 250 generations resulting in 250,000 fitness evaluations for each run on each example.

2.2 Implementation Details for Genetic Programming

Genetic Programming is a variant of Genetic Algorithms, which is able to represent functions and programs by operating on LISP based genotypes [5]. It is used here to generate ordinary differential equations to represent the dynamics of the target network. A GP individual uses n GP-trees to encode the right hand sides of ordinary differential equations, which are built from mathematical operators, constants, and the values of X_{est} as input. In our implementation the function set was restricted to $\{+, -, \cdot\}$ similar to the implementation in [8] and the ephemeral constants where initialized between 0 and 1. The maximal tree depth was set to five and the ramped-half and half method was applied for initialization. The population size was set to 500 and tournament selection was used, with a tournament group size of 50. The GP was performed for 50 generations with elitism, which also resulted in 250,000 evaluations, to give a fair comparison.

3 Experimental Results on Artificial Data

We compared the S-system based ES to the GP on inferring artificial regulatory networks created from multiple S-system based target networks with varying parameters and varying number of genes ($n = 2$ and $n = 5$, this leads to $i = 12$ and $i = 50$ parameters to optimize in case of the S-system based ES).

Although the dimensionality of the test problems used here is very low compared to real world requirements, it proved to be sufficient for this comparison. Since both methods do not scale up satisfactorily we can only point out that the commonly known separation technique can be used to simplify the inferring problem. Instead of inferring all interactions of n genes, each gene can be inferred independently one at a time.

The fact that the S-system based ES has the same structure as the target network could give the ES an advantage over the GP. But in the course of our experiments, we will show that this is not of importance for the results presented here. Further, although the function set of GP was chosen to be insufficient, the GP proved to be competitive to the S-system based ES.

Two examples which were also used by other authors as reference for stable S-systems are given by Tominaga et al. [13], the first example consists of two genes, and the second, which is additionally biologically motivated, consists of five genes. These examples were used as starting points for our first experiments to generate new examples with varying connectivity of the target network.

For each example we created the artificial time series by integrating the S-system from $t_0 = 0$ to t_{max} using a Runge-Kutta algorithm and taking 20 equidistant sample points. Each EA strategy was repeated 25 times for each examined problem instance. The results are given as mean best RSE, 95% confidence intervall, deviation of best RSE and min/max values of RSE for each problem instance.

α_i	$\mathcal{G}_{ij}$	
3	0.0	-2.5
3	2.5	0.0

β_i	$\mathcal{H}_{ij}$	
3	-1.0	0.0
3	0.0	2.0

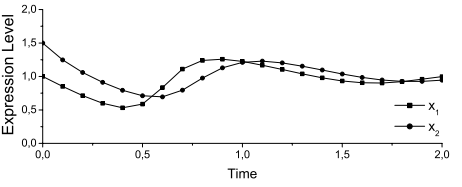

Fig. 2. Parameters and dynamics of the 2D example given in [13]

3.1 Two-Dimensional Examples: Increase of Connectivity

The parameters and the dynamics of the original two-dimensional S-system are given in fig. 2. To examine the impact of different levels of interdependence between the genes (connectivity) we varied the number of non-zero parameters from 0 to 8. We expected that the inference problem would become more difficult with increasing connectivity.

Fig. 3 shows that both methods perform well on all problems in each example, the RSE drops below 0.02 or even 0.01 in case of the S-system based ES. The ES performs slightly better, but when taking into account the overall low level of RSE the difference becomes marginal. And it has to be noted that the function set of the GP is insufficient while the ES has the same structure as the target.

Unfortunately, the performance of the inferring strategies seems to be independent of the connectivity level of the target network. This can be explained by taking into account that the inferred solutions are not necessarily similar to the target systems. Even if the target system has a low connectivity (sparse matrices $\mathcal{G}_{ij}$ and $\mathcal{H}_{ij}$) the solution is usually not sparse. Therefore, the problem is not really easier for targets with low connectivity.

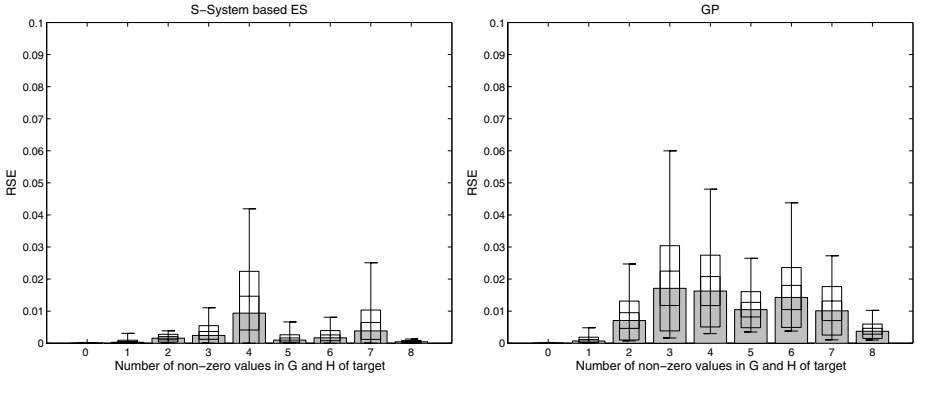

Fig. 3. Comparing ES and GP on two-dimensional examples, $t_{max} = 2.0$

α_i	$\mathcal{G}_{ij}$				
15	0.0	0.0	1.0	0.0	-0.1
10	2.0	0.0	0.0	0.0	0.0
10	0.0	-0.1	0.0	0.0	0.0
8	0.0	0.0	2.0	0.0	-1.0
10	0.0	0.0	0.0	2.0	0.0

β_i	$\mathcal{H}_{ij}$				
10	2.0	0.0	1.0	0.0	0.0
10	0.0	2.0	0.0	0.0	0.0
10	0.0	-0.1	2.0	0.0	0.0
10	0.0	0.0	0.0	2.0	0.0
10	0.0	0.0	0.0	0.0	2.0

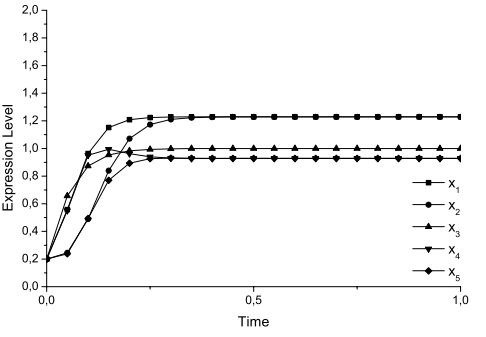

Fig. 4. Parameters and dynamics of the 5D example given in [13]

3.2 Five-Dimensional Examples: Increase of Connectivity

In the second experiment we increased the problem dimension to five and again varied the connectivity of the artificial network. The parameters and the dynamics of the original five-dimensional S-system are given in fig. 4.

First, it has to be noted that the overall performance dropped considerably with the increased problem dimension, see fig. 5. But the performance did not suffer from increasing the connectivity of the target network. Instead the ES showed the worst results on the examples with 23-38 non-zero parameters, but performed well again on the examples with 43 and 48 non-zero parameters.

Comparing the ES to the GP, the GP performed better and more reliable than the ES on all five-dimensional examples regarding the mean RSE and the standard deviation. Only the best results of the S-system based ES are better than the best results of the GP in nearly all examples. The high variation of the

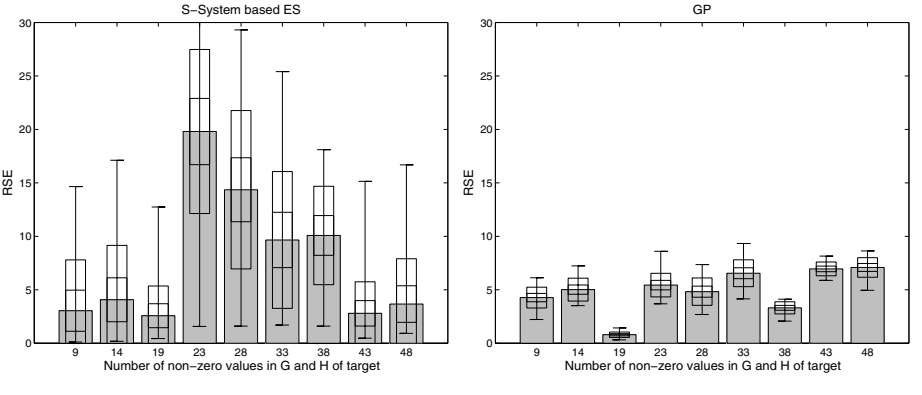

Fig. 5. Comparing ES and GP on five-dimensional examples, $t_{max} = 1.0$

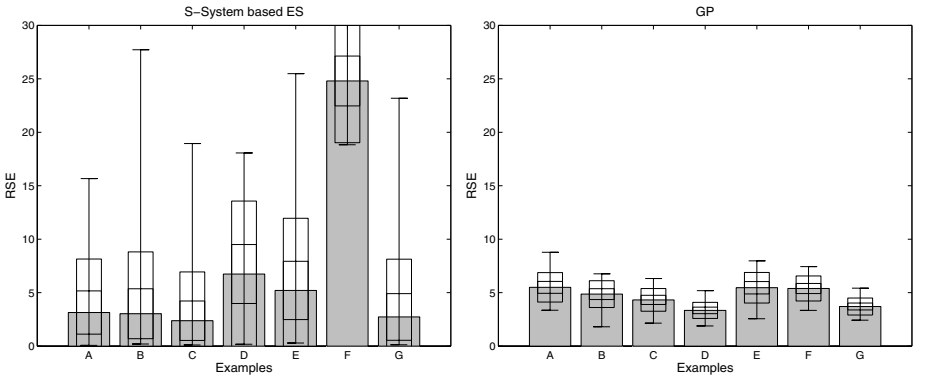

Fig. 6. Comparing ES and GP on five-dimensional examples of same connectivity, $t_{max} = 1.0$

ES could be accounted for by the multi-modal search space of the ES, but on the other hand the search space of the GP is even rougher and also multi-modal.

Interestingly enough the performance of the S-system based ES depends heavily on the instance of the problem regardless of the dimension of the problem, see also example 4 in fig. 3. Therefore, we decided to conduct another experiment with multiple problem instances with the same dimension and the same level of interdependence between the genes.

3.3 Five-Dimensional Examples: Constant Connectivity

We created seven artificial regulatory networks of the same dimension and with same number of non-zero values in $\mathcal{G}_{ij}$ and $\mathcal{H}_{ij}$. Again the performance of the ES depends on the problem instance, see fig. 6. For example the ES fails for problem instance F, while it performs quite well on all other problem instances and even excellent on A, C and G, at least regarding the mean value and the best solution found. The GP on the other hand performs well on all examined problem instances, but does not equal the best runs of the S-system based ES. Again the variance of the GP results is much lower than of the ES results.

This experiment supports the assumption that there are other properties than just the level of interdependence between the genes that impacts the performance of the S-system based ES. The GP on the other hand seems not to be susceptible to such properties, at least for those problems which have been examined here.

4 Discussion

Several conclusions can be drawn from our experiments: first, although the S-system based ES performs very well on most examples, the GP is more reliable and more versatile. Secondly, good RSE values do not necessarily indicate similarity to the target system, neither for the S-system based ES nor for the GP

strategy. Third, and most important, the performance on a single example is not sufficient to evaluate a strategy as it is currently the case in most publications.

Regarding the primary objective of this paper we were able to show that the GP proved to be competitive to the S-system based ES regarding the RSE values reached, although the function set of the GP was insufficient, while the ES utilized the very same model that was used to generate the artificial data. While the ES had the best average results on at least half the examples examined, there were several problem instances where the ES performed much worse than the GP. Also, on all examples the worst results of single runs were produced by the ES. The GP on the other hand performed not as well as the ES regarding the quality of the best solutions found, but the results were more reliable. Further, the GP showed a smaller standard deviation of the results on each example. The GP also proved to be more robust on the different problem instances than the ES. Taking into account that it is unknown whether or not S-systems or similar mathematical models are suitable to represent true gene regulatory networks, GP seems to be the method of choice, since it requires no a priori assumptions about the structure of the gene regulatory network. Regarding the secondary objective we could, show that both approaches suffer from increased problem dimension, while the level of connectivity seems not to be of major relevance. The GP performed slightly better with increased problem dimension than the ES. This could be accounted to the quadratic increase of parameters in case of the S-system based ES compared to the linear increase of complexity for GP.

Secondly, although good RSE values in the experiments might suggest that the target system had been correctly identified, this was not the case in most examples. Especially the S-system based ES produced parameter sets that where often neither sparse like the target system nor were the parameters of the same magnitude as in the target system. The same holds true for GP, although most GP solutions could be considered 'sparse', due to the limited tree depth.

Finally, the experiments showed that the performance of the S-system based ES was heavily depending on the problem instance. This suggests that multiple problem instances are necessary to reliably specify the performance of a given inference strategy, instead of testing the strategy on just one or two examples. Therefore, it is necessary to find a whole set of artificial benchmark regulatory networks based on multiple mathematical models to evaluate inferring strategies.

To actually infer gene regulatory networks from real microarray time series data two issues need to be addressed in future work. First, the problem of ambiguity needs to be resolved. Either by utilizing additional experimental data to remove ambiguity or by introducing biologically motivated constraints to the fitness function like for example partially known gene interactions, preferring sparse networks over fully connected networks or favoring 'robust' networks regarding disturbance over networks with instable dynamics. Second, to tackle problems with higher dimension than the typical five to ten dimensions used in most papers, we need to explore separation strategies or develop new problem specific strategies to escape the curse of dimensionality. Otherwise we need to limit ourself to simpler models like RBN or weight matrices, where more efficient heuristics than EA can be applied.

References

1. T. Akutsu, S. Miyano, and S. Kuhara. Identification of genetic networks from a small number of gene expression patterns under the boolean network model. *Pacific Symposium on Biocomputing*, 4:17–28, 1999.
2. T. Akutsu, S. Miyano, and S. Kuhara. Algorithms for identifying boolean networks and related biological networks based on matrix multiplication and fingerprint function. *Journal of Computational Biology*, 7(3):331–343, 2000.
3. T. Chen, H. L. He, and G. M. Church. Modeling gene expression with differential equations. *Pacific Symposium on Biocomputing*, 4:29–40, 1999.
4. N. Hansen and A. Ostermeier. Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaption. In *Proceedings of the 1996 IEEE International Conference on Evolutionary Computation*, pages 312–317, 1996.
5. J. R. Koza, David Andre, F. H. Bennett III, and M. Keane. *Genetic Programming 3: Darwinian Invention and Problem Solving*. Morgan Kaufman, Apr. 1999.
6. P. Mendes, W. Sha, and K. Ye. Artificial gene networks for objective comparison of analysis algorithms. *Bioinformatics*, 19(2):122–129, 2003.
7. R. Morishita, H. Imade, I. Ono, N. Ono, and M. Okamoto. Finding multiple solutions based on an evolutionary algorithm for inference of genetic networks by s-system. In *Congress on Evolutionary Computation*, pages 615–622, 2003.
8. E. Sakamoto and H. Iba. Inferring a system of differential equations for a gene regulatory network by using genetic programming. In *Proceedings of Congress on Evolutionary Computation*, pages 720–726. IEEE Press, 27-30 2001.
9. M. Savageau. 20 years of S-systems. In E. Voit, editor, *Canonical Nonlinear Modeling. S-systems Approach to Understand Complexity*, pages 1–44, New York, 1991. Van Nostrand Reinhold.
10. H.-P. Schwefel. *Evolution and Optimum Seeking*. John Wiley & Sons, New York, 1995.
11. C. Spieth, F. Streichert, N. Speer, and A. Zell. Iteratively inferring gene regulatory networks with virtual knockout experiments. In *Applications of Evolutionary Computing, EvoWorkshops2004: EvoBIO, EvoCOMNET, EvoHOT, EvoIASP, EvoMUSART, EvoSTOC*, volume 3005 of *LNCS*, pages 102–111, Coimbra, Portugal, 5-7 April 2004. Springer Verlag.
12. D. Thieffry and R. Thomas. Qualitative analysis of gene networks. *Pacific Symposium on Biocomputing*, 3:77–88, 1998.
13. D. Tominaga, N. Kog, and M. Okamoto. Efficient numerical optimization technique based on genetic algorithm for inverse problem. In *Proceedings of German Conference on Bioinformatics*, pages 127–140, 1999.
14. D. Tominaga, M. Okamoto, Y. Maki, S. Watanabe, and Y. Eguchi. Nonlinear numerical optimzation technique based on genetic algorithm for inverse problem: Towards the inference of genetic networks. In *Proceedings of Genetic and Evolutionary Computation Conference*, pages 251–258. Morgan Kaufmann, 2000.
15. M. Wahde and J. Hertz. Coarse-grained reverse engineering of genetic regulatory networks. *Biosystems*, 55:129–136, 2000.
16. D. C. Weaver, C. T. Workman, and G. D. Stormo. Modeling regulatory networks with weight matrices. In *Pacific Symposium on Biocomputing*, volume 4, pages 112–123, Singapore, 1999. World Scientific Press.
17. A. Wuensche. Genomic regulation modeled as a network with basins of attraction. In R. Altman, A. Dunker, L. Hunter, and T. Klein, editors, *Pacific Symposium on Biocomputing*, volume 3, pages 89–102, Singapore, 1998. World Scientific Press.