Multi-Objective Model Optimization for Inferring Gene Regulatory Networks

Christian Spieth, Felix Streichert, Nora Speer, and Andreas Zell

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

Abstract. With the invention of microarray technology, researchers are able to measure the expression levels of ten thousands of genes in parallel at various time points of a biological process. The investigation of gene regulatory networks has become one of the major topics in Systems Biology. In this paper we address the problem of finding gene regulatory networks from experimental DNA microarray data. We suggest to use a multi-objective evolutionary algorithm to identify the parameters of a non-linear system given by the observed data. Currently, only limited information on gene regulatory pathways is available in Systems Biology. Not only the actual parameters of the examined system are unknown, also the connectivity of the components is a priori not known. However, this number is crucial for the inference process. Therefore, we propose a method, which uses the connectivity as an optimization objective in addition to the data dissimilarity (relative standard error - RSE) between experimental and simulated data.

1 INTRODUCTION

Gene regulatory networks (GRNs) represent the dependencies of the different actors in a cell operating at the genetic level. They dynamically determine the level of gene expression for each gene in the genome by controlling whether a gene will be transcribed into RNA or not. A simple GRN consists of one or more input signalling pathways, several target genes, and the RNA and proteins produced from those target genes. In addition, such networks often include dynamic feedback loops that provide further network regulation activities and output. In order to understand the underlying structures of activities and interactions of intra-cellular processes one has to understand the dependencies of gene products and their impact on the expression of other genes. Therefore, finding a GRN for a specific biological process would explain this process from a logical point of view, thus explaining many diseases.

Therefore, the model reconstruction of gene regulatory networks has become one of the major topics in bioinformatics. However, the huge number of system components requires a large amount of experimental data to infer genome-wide networks. Recently, DNA microarrays have become one of the major tools in the research area of microbiology. This technology enables researchers to monitor the activities of thousands of genes in parallel and can therefore be used as a powerful tool to understand the regulatory mechanisms of gene expression in a cell. With this technique, cells can be studied under several conditions such as medical treatment or different environmental influences.

Microarray experiments often result in time series of measured values indicating the activation level of each tested gene in a genome. These data series can then be used to examine the reactions of the cell to external stimuli. A model would enable biologists to predict the reactions of intracellular signalling processes. To re-engineer or infer the regulatory processes computationally from these experimental data sets, one has to find a model that is able to produce the same time series data as the experiments. The idea is then that the model reflects the true system dependencies, i.e. the dependencies of the components of the regulatory system.

Several approaches have been made to address this problem. Many of them are only relying on the distance between the experimental data and the simulated data coming from the mathematical model, but the biological plausibility of the system is almost always neglected. And although Biologist know, that regulatory systems are sparse, i.e. one gene relies on average on a small number of other genes, this fact can be found only in some publications.

In this paper we propose a methodology for reverse engineering sets of time series data obtained by artificial expression analysis by combining two objectives into a multi-objective optimization problem. The first objective is the dissimilarity between the experimental and the simulated data (RSE). The second objective is the connectivity of the system. Both objectives are to be minimized to gain a system, which fits the data and at the same time is only sparsely connected and therefore biological plausible. With this approach, we systematically examine the impact of the connectivity of the regulatory network on the overall inference process.

The remainder of this paper is structured as follows. Section 2 of this paper presents an overview over related work and lists associated publications. Detailed description of our proposed method will be given in section 3 and example applications will be shown in section 4. Finally, conclusions and an outlook on future research will be covered by section 5.

2 RELATED WORK

Researchers are interested in understanding the mechanisms of gene regulatory processes and therefore in inferring the underlying networks. The following section briefly describes the work that has been done in this area.

The earliest models to simulate regulatory systems found in the literature are boolean or random boolean networks (RBN) [8]. In boolean networks gene expression levels can be in one of two states: either 1 (on) or 0 (off). The quantitative level of expression is not considered. Two examples of algorithms for inferring GRNs with boolean networks are given by Akutsu [1] and the REVEAL algorithm by Liang *et al.* [11]. Boolean networks have the advantage that they can be solved with only small computational effort, but they suffer from the disadvantage of being tied to discrete system states.

In contrast to discrete models like RBNs, qualitative network models allow for multiple levels of gene regulation. An example for this kind of approach is given by Thieffry and Thomas in [20], where a qualitative model was successfully inferred. Akutsu *et al.* suggested a heuristic for inferring such models in [2]. But those models use only qualitative dependencies and therefore only a small part of the information hidden in the time series data.

Quantitative models based on linear models for gene regulatory networks like the weighted matrix model introduced and inferred by Weaver *et al.* [22] consider the continuous level of gene expression. Another algorithm to infer quantitative models is the singular value decomposition method by Yeung *et al.* [23]. Drawback of these models is that they are restricted to linear dependencies, which makes it difficult to model biological networks due to the fact that biological systems are known to be non-linear.

A general example for a mathematical non-linear model is the S-System, which was introduced by Savageau [15] in the 90s. S-Systems for inferring regulatory mechanisms have been examined by Maki *et al.* [12], Kiguchi *et al.* [10] or Tominaga *et al.* [21]. Non-linear models and especially the S-System face the severe disadvantage of having many model parameters to be inferred ($2(N+N^2)$ with N the number of genes). Tominaga *et al.*, for example, bypassed this problem by inferring only a subset of genes of the original system (2 genes out of 5). Methods using S-Systems or other non-linear models regard only a small number of components within the regulatory network to be reverse engineered, i.e. a total number of genes $N \leq 10$.

So far, only parameterized models have been considered in this section. Other approaches, for example non-parameterized models, to infer regulatory systems from time series data using artificial neural networks [9] or bayesian networks [7] have been recently published, but face some drawbacks as well. Bayesian networks, for example, do not allow for cyclic networks, which are known to exist in biological systems. Another kind of non-parameterized model are arbitrary differential equations, which can also be used to model regulatory structures as Ando *et al.* showed with genetic programming (GP) in [3]. In this publication, GP was used to set up a system of suitable differential equations, whose time series are then compared to the experimental data.

Although the standard single-objective algorithms, which were used in some cases for parameter optimization for the parameterized models, tend to find completely connected networks and thus biologically implausible solutions, the number of connections was not considered to be of any importance. Knowing that the system that is to be inferred is sparse by the very nature of biological systems together with the necessity to find a model with the same time dynamic behavior, the problem of finding a set of parameters for a mathematical model is obviously multi-objective; thus, MOEAs should be used.

Further more, previously work on this topic showed that, due to the multi-modal character of the solution space, several sets of parameter exist, which fit the data satisfactorily. Thus, standard optimization techniques are easily caught in local optima, i.e. finding a solution with a good RSE but with no structural resemblance with the true system. This is known to be a major problem in the inference process [18, 13, 6, 14]. Because MOEAs preserve the diversity of the solution within a population by maintaining the Pareto-front and are therefore able to find multiple optima hopefully including the global optimum.

As a conclusion of this, the dissimilarity between the simulated data and the experimental data has to be minimized and at the same time, the connectivity is to be minimized too. All this suggest to use MOEAs for the inference of GRNs. Thus, we propose a new approach to the inference problem of regulatory systems, namely two MOEA implementations.

3 METHODS

In this section we want to give the details of the mathematical model used to simulate the gene regulatory system in the inference process together with a description of the multi-objective EA.

3.1 Model

On an abstract level, the behavior of a cell is represented by a gene regulatory network of N genes. Each gene g_i produces a certain amount of RNA x_i when expressed and therefore changes the concentration of this RNA level over time: $\boldsymbol{x}(t+1) = h(\boldsymbol{x}(t))$, $\boldsymbol{x}(t) = (x_1, \cdots, x_n)$.

To model and to simulate regulatory networks we decided to use S-Systems since they are well-documented and examined. But there are alternatives as listed in section 2, which will be the subject of research in future applications.

S-Systems are a type of power-law formalism which has been suggested by Savageau [15] and can be described by a 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 $\mathcal{G}_{i,j}$ and $\mathcal{H}_{i,j}$ are kinetic exponents, α_i and β_i are positive rate constants and N is the number of equations in the system. The equations in (1) can be seen as divided into two components: an excitatory and an inhibitory component.

The kinetic exponents $\mathcal{G}_{i,j}$ and $\mathcal{H}_{i,j}$ determine the structure of the regulatory network. In the case $\mathcal{G}_{i,j} > 0$ gene g_j induces the synthesis of gene g_i. If $\mathcal{G}_{i,j} < 0$ gene g_j inhibits the synthesis of gene g_i. Analogously, a positive (negative) value of $\mathcal{H}_{i,j}$ indicates that gene g_j induces (suppresses) the degradation of the mRNA level of gene g_i.

3.2 Multi-objective Algorithm

For examining the connectivity and the RSE in parallel, we used a multi-objective EA, which optimizes the parameters of $\mathcal{G}$, $\mathcal{H}$, α_i and β_i in respect to the following two optimization objectives:

I.) For evaluating the RSE fitness of the individuals we used the following equation for calculation of the fitness values:

$$f_1 = \sum_{i=1}^{N} \sum_{k=1}^{T} \left\{ \left(\frac{\hat{x}_i(t_k) - x_i(t_k)}{x_i(t_k)} \right)^2 \right\} \tag{2}$$

where N is the total number of genes in the system, T is the number of sampling points taken from the experimental time series and $\hat{x}$ and x distinguish between estimated data of the simulated model and data sampled in the experiment. The problem is to minimize the fitness value f_1.

II.) The second optimization objective is to minimize the connectivity of the system, as biologically the gene regulatory network is known to be sparse. The connectivity is defined in two different ways: first, the maximum connectivity of the genes, i.e. the total number of interactions of the system:

$$f_2^a = \sum_{i=1}^{N} (|\text{sign}(\alpha_i)| + |\text{sign}(\beta_i)|) + \sum_{i=1}^{N} \sum_{j=1}^{N} (|\text{sign}(\mathcal{G}_{i,j})| + |\text{sign}(\mathcal{H}_{i,j})|) \tag{3}$$

And secondly, the median average connectivity of all genes, i.e. the median average number of interactions of each gene:

$$f_2^b = \text{median}(\frac{f_2^a}{N}) \tag{4}$$

3.3 Test Cases

The optimization experiments were performed with different configurations. In the first test class (**Test class I: Hybrid**), an EA with hybrid encoding individuals was used to minimize the connectivity of the regulatory model, which was recently developed by the authors [19]. This individual combines a binary and a real valued genotype that are evolved in parallel. The binary variables are used to determine a topology or structure of the network and the double encoded optimization variables represent the corresponding model parameter. The individuals always encode all possible model parameters but only some of them are used for simulation according to the binary representation of the topology. Nevertheless, the unused variables are continuously evolved and subject to random walk and might be incorporated in the simulation if the bitset changes.

This enables the optimizing algorithm to escape local optima. This algorithm has to optimize $2(N + N^2)$ real valued parameters for the S-System plus $2N^2$ bits for each entry $\mathcal{G}_{i,j}$ and $\mathcal{H}_{i,j}$, thus in total $2N + 4N^2$ variables.

Test class II (MA) is a memetic algorithm, where an individual of the MOEA represents the topology together with an evolution strategy, which searches for the best parameters for the given topology. The memetic algorithm uses a MOEA population to evolve populations of topologies of possible networks. These topologies are encoded as bitsets, where each bit represents the existence or absence of an interaction between genes and therefore of non-zero parameters in the mathematical model. The evaluation of the fitness of each individual within the MOEA population applies a local search to find suitable parameters. For evaluation of each structure suggested by the MOEA population an evolution strategy is used, which is suited for the parameter optimizing problem, since it is based on real values. The ES optimizes the parameters of the mathematical model used for representation of the regulatory network. This algorithm was introduced by the authors in [17]. The current implementation is working on both, $\mathcal{G}$ and $\mathcal{H}$, thus having the same number of bits as the algorithm in the previous case ($2N^2$). For future implementations it would be interesting to encode only the logical dependency between genes and set the corresponding bits in $\mathcal{G}$ and $\mathcal{H}$ at the same time. The difference between the hybrid encoding of I and the II is that the latter is optimizing only the S-System parameters for those values of the bitset that are true. Therefore, this algorithm has a dynamic range of the total number of model parameters between 0 (no connectivity at all) and $2(N + N^2)$ for the complete S-System.

Furthermore, a third test class (**Test class III: Skeletalizing**) was implemented, where a technique called *skeletalizing* was used [21]. This is an extension to a standard real-coded GA that introduces a threshold value t_{skel}, which represents a lower boundary for the parameters $\mathcal{G}_{i,j}$ and $\mathcal{H}_{i,j}$ in the mathematical model. If the absolute value of a decoded decision variable of the GA drops below this threshold during optimization the corresponding phenotype value is forced to 0.0 indicating no relationship between the components. Thus, $|\mathcal{G}_{i,j}| < t_{skel} \rightarrow \mathcal{G}_{i,j} = 0.0$. This algorithm has the same total number of parameters to optimize as the MA described above.

Overall, a total number of three different algorithms was tested on two problem instances (f_1, f_2^a) and (f_1, f_2^b) with artificial experimental data to examine the impact of the connectivity on the overall optimization process.

3.4 Algorithm Settings

The settings and chosen parameter for each of the three algorithm test classes will be given in the following sections.

Test Class I: Hybrid The multi-objective runs were performed using a real-valued NSGA-II algorithm with a population size of 500 individuals, crowded tournament selection with a tournament group size of $t_{group} = 8$, Uniform-crossover recombination with $p_c = 1.0$ and a mutation probability $p_m = 0.1$. (Details on the implementation is given in Deb *et al.* [4]).

Test Class II: MA The memetic algorithm used also an NSGA-II to evolve a population of possible topologies with crowded tournament selection with a tournament group size of $t_{group} = 8$, Uniform-crossover with $p_c = 1.0$ and a mutation probability $p_m = 0.1$. The local optimization was performed using a (μ,λ)-ES with $\mu = 10$ parents and $\lambda = 20$ offsprings together with a Covariance Matrix Adaptation (CMA) (see Hansen and Ostermeier [5]) mutation operator without recombination.

Test Class III: Skeletalizing This test class was a standard real-coded NSGA-II using a population of 500 individuals, crowded tournament selection with a tournament group size of $t_{group} = 8$, Uniform-crossover with $p_c = 1.0$ and a mutation probability of $p_m = 0.1$ together with the threshold value $t_{skel} = 0.05$.

To keep track of the Pareto-front the multi-objective algorithms maintained an archive of (population size/2) individuals and used this archive as elite to achieve a faster convergence. Each test class experiment terminated after $100,000$ fitness evaluations. Each example setting was repeated 20 times.

We further compared the MOEAs to a standard (μ,λ)-ES with $\mu = 10$ parents and $\lambda = 20$ offsprings together with CMA and no recombination. We tested this ES on the artificial systems with 20 multiruns and evaluated the overall best individual found in the runs.

4 APPLICATIONS

To illustrate our method, we established two regulatory network systems, which were simulated to gain sets of expression data. After creating the data sets, we used our proposed algorithm to reverse engineer the correct model parameters. The following section show this for a 5-dimensional and a 10-dimensional example.

4.1 Artificial Gene Regulatory Network

Due to the fact that GRNs in nature are sparse systems, we created regulatory networks randomly with a maximum cardinality of $k <= 3$, i.e. each of the N genes depends on three or less other genes within the network. The total connectivity of the 5-dimensional example was 36, i.e. the number of non-zero parameters was 36. As a second and, due to the increased number of participating genes, more difficult test case, we created another regulatory network randomly with a maximum cardinality of $k <= 3$. The total connectivity of the 10-dimensional example was 78.

Target The time dynamics of the systems can be seen in Fig. 1 and 2, respectively. In the figures, each x_i represents the RNA level of a certain gene. At this point, we do not differentiate between closely related molecules like mRNA and distantly related like proteins.

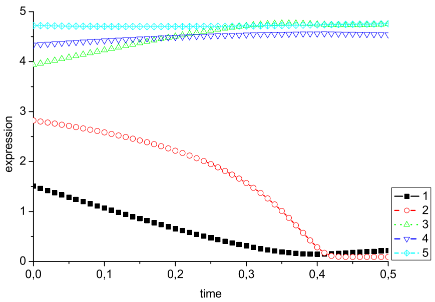

Fig. 1. 5-dimensional GRN. The numbers indicate the component genes.

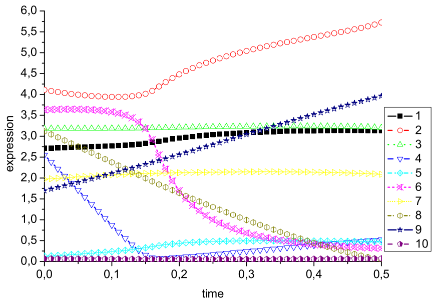

Fig. 2. 10-dimensional GRN. The numbers indicate the component genes.

Obviously, the information about the true connectivity was not used in the optimization process it served only for validation purposes.

4.2 Initial Results

In the initial implementation, the MOEA found the only one Pareto-optimal solution with a connectivity of 0. This is caused by the trivial solution with zero interactions. With a connectivity of 0, the system is static and does not change over time. Because the expression levels of the system are in the range of 0.0 and 5.0, the data distance of the static case is very small compared to many of the systems with a larger connectivity and often unstable dynamics. To bypass this issue, we multiplied the data sets by a factor $f_{scale} = 1,000,000$ and introduced therefore a penalty for being static. During evaluation and postprocessing of the results, we reversed this scaling to gain fitness values for the original data range.

4.3 5-dimensional example

The actual results of the three test classes are shown for the 5-dimensional example in the next figures. Fig. 3 shows the Pareto-front results of the three test classes accumulated over all 20 multiruns with the RSE as ordinate and the total number of interactions in the inferred model as abscissa.

As one can see, with increasing number of interactions, the RSE is getting better due to the ambiguity in the data sets. Because the optimizer has more

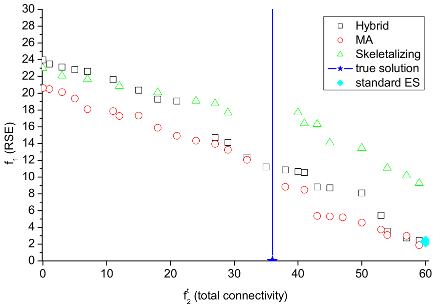

Fig. 3. Pareto-front of the 5-dim example (total connectivity)

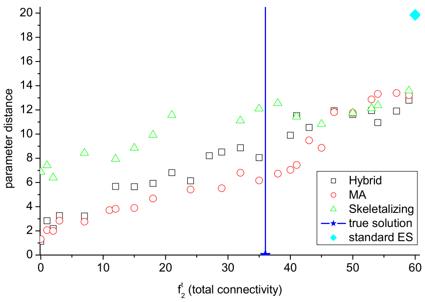

Fig. 4. Distances of the 5-dim example (total connectivity)

parameters and thus more model variables to use for fitting the data, it becomes more easy to actually reach the desired time series course. But as can be seen in the figure to the right (Fig. 4), the euclidian distance between the true system and the model found by the EA is getting worse, due to the sparseness of the true system. This validates the results of previous work [17].

The next two figures show the Pareto-front and the parameter distance of the 5-dimensional example in the case that the median average number of interactions of each gene was taken into account as the second optimization objective.

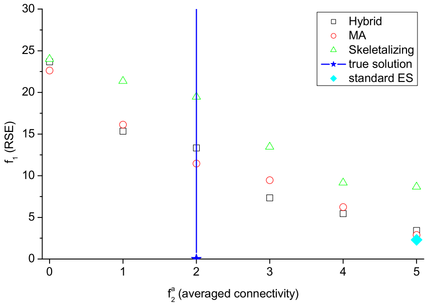

Fig. 5. Pareto-front of the 5-dim example (average connectivity)

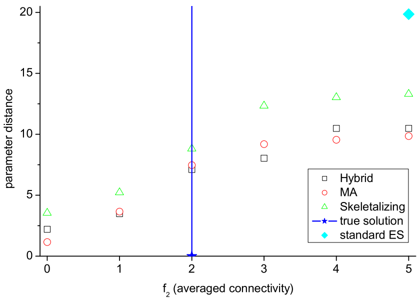

Fig. 6. Distances of the 5-dim example (average connectivity)

There is almost no difference between the two definitions of connectivity as can be see in the figure. All algorithms yield similar results as in the test case of the total connectivity. And as in the previous results shown, there is no significant difference between the two multi-objective implementations (MA and hybrid). Both, the MA and the hybrid algorithm, outperform the skeletalizing GA.

The star in the figures show the true solution with a total connectivity of 36. None of the tested algorithm is able to reach the true solution. This might be partly due to the limited number of total fitness evaluations. But the major problem for all EAs is that the solution space is highly multi-modal and in case of the S-System very large.

For comparison, we tested additionally a standard ES on the artificial systems. These results are indicated in the figures with the rhombus symbol. As one can see, the standard ES prefers fully connected networks. The averaged best RSE fitness values in case of the 5-dimensional example are as good as those of the multi-objective EAs. But the second objective, the distance between the optimized parameters and the true system, is much better for the MOEAs. The ES found solutions, which fit the data comparably good, but show less resemblance with the original system.

Overall, two conclusions can be drawn from the figures: first, the MA and the hybrid algorithm perform significantly better than the algorithm taken from the literature, i.e. the skeletalizing GA and a standard ES. Probably the ability to preserve the diversity in the population supports the MOEA to find better solutions.

Secondly, the memetic algorithm performs on average slightly better than the hybrid encoding individual. One advantage for the MA in future applications might be the fact that EAs tend to yield better solutions with a local search, which refines the found parameters.

4.4 10-dimensional example

The conclusions from the previous section are validated by the results of the 10-dimensional example as given in Fig. 7 and 8.

Again, the fitness values for fitting the data (RSE) are getting better for increasing numbers of interactions in the regulatory system. The distance to the true model parameters is getting worse. And as in the 5-dimensional example, the solutions were not even close to the true solution, indicated with the star. For the true connectivity, no algorithm was able to find the correct parameters. The reason for this can be understood, if one looks closer at the topologies for this connectivity. The algorithms examined the correct degree of dependencies but with wrong interactions partners. In this case, it is obviously not possible to find the correct values for the model parameters. The conclusion from this fact is that more topologies have to be examined, which is easy to perform with the MA or the hybrid algorithm and a higher number of total fitness evaluations.

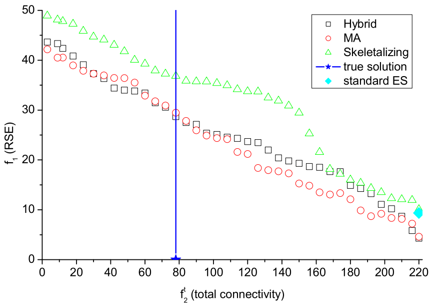

Fig. 7. Pareto-front of the 10-dim example (total connectivity)

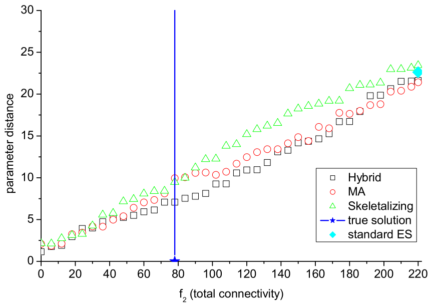

Fig. 8. Distances of the 10-dim example (total connectivity)

The standard ES performed better than in the previous example. The RSE fitness values were comparably good as those of the totaly connected MOEA results. And also the parameter distances of the ES results corresponded with those of the MOEA solutions. But again, the ES was not able to find any solutions that were not fully connected. Considering that the ES had as many fitness evaluations as the MOEA, the MOEA performed much better. This supports the claim of the usefulness of the increased diversity in a MOEA population.

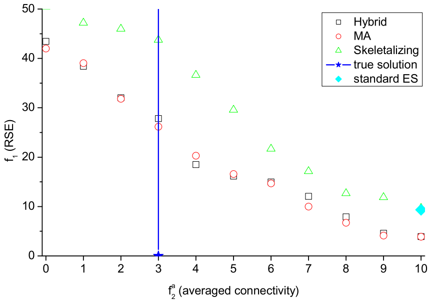

Fig. 9. Pareto-front of the 10-dim example (average connectivity)

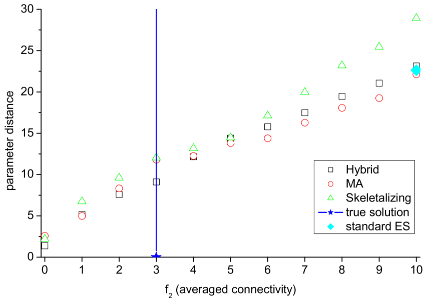

Fig. 10. Distances of the 10-dim example (average connectivity)

Again, the choice of the connectivity definition did not influence the overall results as can be seen in figures 9 and 10. Both implementations resulted in similar solutions in respect to the parameter distance and the RSE fitness.

5 DISCUSSION

5.1 Conclusions

In this paper we compared different multi-objective strategies to infer gene regulatory networks from time-series microarray data. The results of the test examples showed that the number of interactions has a crucial impact on the ability of an EA to fit a mathematical model to a given time series. High numbers of connections between system components yield in better results in respect to the data fitness. This is due to the large number of model parameters and the small number of data sets available, the system of equations is highly underdetermined. Therefore, multiple solutions exist, which fit the given data, but show only little resemblance with the original target system. This problem is known in literature but there are currently only few publications reflecting on this issue. The multi-objective strategy showed promising performance in comparison to standard single-objective algorithms. MOEA are better at preserving the diversity of the solutions in the population as standard algorithms do and are able to better cope with the ambiguity issues than single-objective optimization algorithms as we have shown in this paper. This is especially important for the highly multi-modal solutions space in the case of inference problems.

On the other side, with increasing number of interactions, it becomes more and more difficult to find the true system in respect to correct parameter values. Because biological systems are sparse, algorithm have to take this fact into account to find better solutions. In this example, we knew that the true solutions existed with the known connectivity. In real world applications, researches do not know, which connectivity is the correct one. MOEAs are again a promising approach to this problem as they result in a set of Pareto-optimal solutions from which researchers can choose the most suitable model complying with, for example, biological constraints. Biologist would be able to take one or more solutions as hypotheses and examine them in additional experiments.

5.2 Outlook

Due to the ambiguity in the data, it is difficult for EAs to find the correct solution as concluded above. Recently, the authors published a method to incorporate data sets obtained by additional experiments [16]. As one future enhancement of the proposed methods, we plan to incorporate such additional methods to identify the correct network.

As the initial results showed, precautions have to be taken to prevent the MOEA from finding the trivial solution of no connectivity. This can be done for example by scaling the experiment data as we did in this publication or one could introduce a penalty term to scale the objective function.

In future work we also plan to include a-priori information into the inference process of real microarray data like partially known pathways or information about co-regulated genes, which can be found in literature or in public databases.

Here, the dissimilarity to the known pathway could be included as a third objective, which is also to be minimized. This would enable the MOEA to search for models consistent with current biological knowledge, but would also allow for alternative solutions where biological information is missing or faulty.

Furthermore, additional models for gene regulatory networks will be examined for simulation of the non-linear interaction system as listed in Sect. 2 to overcome the problems with a quadratic number of model parameters of the S-System.

Acknowledgment

This work was supported by the National Genome Research Network (NGFN) of the Federal Ministry of Education and Research in Germany under contract number 0313323.

References

1. T. Akutsu, S. Miyano, and S. Kuhura. Identification of genetic networks from a small number of gene expression patterns under the boolean network model. In *Proceedings of the Pacific Symposium on Biocomputing*, pages 17–28, 1999.
2. T. Akutsu, S. Miyano, and S. Kuhura. Algorithms for identifying boolean networks and related biological networks based on matrix multiplication and fingerprint function. In *Proceedings of the fourth annual international conference on Computational molecular biology*, pages 8 – 14, Tokyo, Japan, 2000. ACM Press New York, NY, USA.
3. S. Ando, E. Sakamoto, and H. Iba. Evolutionary modeling and inference of gene network. *Information Sciences*, 145(3-4):237–259, 2002.
4. K. Deb, S. Agrawal, A. Pratab, and T. Meyarivan. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii. In *Proceedings of the Parallel Problem Solving from Nature VI Conference*, number 1917 in LNCS, pages 849–858, 2000.
5. N. Hansen and A. Ostermeier. Adapting arbitrary normal mutation distributions in evolution strategies: the covariance matrix adaptation. In *Proceedings of the 1996 IEEE Int. Conf. on Evolutionary Computation*, pages 312–317, Piscataway, NJ, 1996. IEEE Service Center.
6. J. Herz. Statistical issues in reverse engineering of genetic networks. In *Proceedings of the Pacific Symposium on Biocomputing*, 1998.
7. S. Imoto, T. Higuchi, T. Goto, K. Tashiro, S. Kuhara, and S. Miyano. Combining microarrays and biological knowledge for estimating gene networks via bayesian networks. In *Proceedings of the IEEE Computer Society Bioinformatics Conference (CSB 03)*, pages 104 –113. IEEE, 2003.
8. S. A. Kauffman. *The Origins of Order.* Oxford University Press, New York, 1993.
9. E. Keedwell, A. Narayanan, and D. Savic. Modelling gene regulatory data using artificial neural networks. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN 02)*, volume 1, pages 183–188, 2002.
10. S. Kikuchi, D. Tominaga, M. Arita, K. Takahashi, and M. Tomita. Dynamic modeling of genetic netowrks using genetic algorithm and s-sytem. *Bioinformatics*, 19(5):643–650, 2003.

11. S. Liang, S. Fuhrman, and R. Somogyi. REVEAL, a general reverse engineering algorithm for inference of genetic network architectures. In *Proceedings of the Pacific Symposium on Biocomputing*, volume 3, pages 18–29, 1998.
12. Y. Maki, D. Tominaga, M. Okamoto, S. Watanabe, and Y. Eguchi. Development of a system for the inference of large scale genetic networks. In *Proceedings of the Pacific Symposium on Biocomputing*, volume 6, pages 446–458, 2001.
13. I. Ono, R. Yoshiaki Seike, N. Ono, and M. Matsui. An evolutionary algorithm taking account of mutual interactions among substances for inference of genetic networks. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2004)*, pages 2060–2067, 2004.
14. C. Pridgeon and D. Corne. Genetic network reverse-engineering and network size; can we identify large grns? In *Proceedings of the Computational Intelligence in Bioinformatics and Computational Biology (CIBCB 2004)*, pages 32–36, 2004.
15. M. A. 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.
16. C. Spieth, F. Streichert, N. Speer, and A. Zell. Iteratively inferring gene regulatory networks with virtual knockout experiments. In R. et al., editor, *Proceedings of the 2nd European Workshop on Evolutionary Bioinformatics (EvoWorkshops 2004)*, volume 3005 of *LNCS*, pages 102–111, 2004.
17. C. Spieth, F. Streichert, N. Speer, and A. Zell. Optimizing topology and parameters of gene regulatory network models from time-series experiments. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2004)*, volume 3102 (Part I) of *LNCS*, pages 461–470, 2004.
18. C. Spieth, F. Streichert, N. Speer, and A. Zell. Utilizing an island model for ea to preserve solution diversity for inferring gene regulatory networks. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2004)*, pages 146–151, 2004.
19. F. Streichert, H. Ulmer, and A. Zell. Evaluating a hybrid encoding and three crossover operators on the constrained portfolio selection problem. In *Proceedings of the 2004 Congress on Evolutionary Computation*, pages 932–939, 2004.
20. D. Thieffry and R. Thomas. Qualitative analysis of gene networks. In *Proceedings of the Pacific Symposium on Biocomputing*, pages 77–87, 1998.
21. D. Tominaga, N. Kog, and M. Okamoto. Efficient numeral optimization technique based on genetic algorithm for inverse problem. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 00)*, pages 251–258, 2000.
22. D. Weaver, C. Workman, and G. Stormo. Modeling regulatory networks with weight matrices. In *Proceedings of the Pacific Symposium on Biocomputing*, volume 4, pages 112–123, 1999.
23. M. K. S. Yeung, J. Tegner, and J. J. Collins. Reverse engineering gene networks using singular value decomposition and robust regression. In *Proceedings of the National Academy of Science USA*, volume 99, pages 6163–6168, 2002.