Inferring Regulatory Systems with Noisy Pathway Information

C. Spieth, F. Streichert, N. Speer, and A. Zell
Centre for Bioinformatics Tübingen (ZBIT),
University of Tübingen,
Sand 1, 72076 Tübingen, Germany

Abstract: With increasing number of pathways available in public databases, the process of inferring gene regulatory networks becomes more and more feasible. The major problem of most of these pathways is that they are very often faulty or describe only parts of a regulatory system due to limitations of the experimental techniques or due to a focus specifically only on a subnetwork of a larger process. To address this issue, we propose a new multi-objective evolutionary algorithm in this paper, which infers gene regulatory systems from experimental microarray data by incorporating known pathways from publicly available databases. These pathways are used as an initial template for creating suitable models of the regulatory network and are then refined by the algorithm. With this approach, we were able to infer regulatory systems with incorporation of pathway information that is incomplete or even faulty.

1 Introduction

Systems biology has become one of the major research areas in biology in the past few years. Due to tremendous progress in experimental methods like DNA microarrays, several thousand expression levels of genes in an organism can be measured in parallel under specific environmental conditions. This enables researchers to examine intra-cellular processes on a systemic level. Here, the inference of gene regulatory networks from experimental data is one of the main unsolved problems in the post-genomic area. A gene regulatory network (GRN) is an abstract model representing dependencies between genes using a directed graph. In this graph, each node is a gene or component of the regulatory system and each edge represents a regulatory impact from one component to the other (e.g. activation or suppression of the transcription and translation of the dependent gene).

Several publications addressing the problem of inferring gene regulatory networks can be found in the literature. A good overview about related work can be found in [dJ02]. One major part of the work done in this field is using deterministic mathematical models to simulate regulatory networks. One kind of those deterministic models are S-Systems, which consist of a set of differential equations describing the changes in expression over time. So far, only very small networks have been successfully inferred by computational methods or larger networks have been reconstructed, where the participating genes show very similar time dynamical behavior to the target system, but the correctness of the connections in the graph cannot be verified [KTA$^+$03, TKO00, KIK$^+$05]. The main obstacle is the

ambiguity in the data and the resulting high number of possible network structures. This is caused by the limited number of microarrays compared to the number of variables in the network model, thus making the estimation of the underlying system a very difficult task. Only a small number of recent published methods are using additional biological data to infer regulatory dependencies from microarray data sets and thus suffer from the problems described above. Therefore, we think that algorithms for inferring gene regulatory systems have to include biological knowledge to successfully reconstruct the network from experimental data. An example for a combinatorial approach was introduced by [TKB$^+$03] where the authors combined a Bayesian network model with biological knowledge about promoter regions in the DNA sequence to find better solutions in the inference process. With the recently increased number of pathways available in public databases, we suggest to utilize this knowledge in the inference process to model network structures more efficiently. However, the pathways available are often faulty or they describe only parts of a regulatory system that is being examined. To overcome this problem, we introduce a new algorithm, which incorporates known pathway information and uses it as a first template for valid model structures. The advantage is that the algorithm is not using the pathway as a fixed structure but as a clue to find the correct pathway. For this, it modifies the template and is therefore able to find errors or unknown interactions in the given pathway. To do so, we combine two objectives into a multi-objective optimization problem in our algorithm. The first objective is the dissimilarity between the experimental and the simulated data. The second objective is the difference to a given pathway that was imported from other resources like public databases (e.g. KEGG [KG00], or TransFAC [WCF$^+$01]) or from biological knowledge at the researcher's site. All objectives are to be minimized to gain a system, which fits the data and at the same time is similar in its structure to the given pathway and therefore biologically plausible.

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 [SSSZ04, OYSOM04, Her98, PC04]. 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.

2 System and methods

The first step of the proposed method is to import pathway information to gain a basic knowledge about the pathway of interest. This imported pathway is then evolved and optimized through the algorithm to build possible alternative structures (or topologies) of the network, which are then evaluated. These topologies are optimized along with the parameters of the mathematical model representing the regulatory system of this topology.

The optimization algorithm is independent of the mathematical model, and several models can be used for simulating the dynamics of the regulatory system. Thus, we first introduce

the multi-objective EA (MOEA), and then discuss the mathematical model in detail. For this paper, we decided to use an S-System for modeling the dependencies of the genes.

2.1 Evolutionary Algorithms

Evolutionary Algorithms (EAs) have proven to be a powerful tool for solving complex optimization problems. Three main types of Evolutionary Algorithms have evolved during the last 30 years: Genetic Algorithms (GA), mainly developed by J.H. Holland [Hol75], Evolution Strategies (ES), developed by I. Rechenberg [Rec73] and H.-P. Schwefel [Sch81] and Genetic Programming (GP) by J.R. Koza [Koz92]. Each of these uses different representations of the data and different main operators working on them. They are, however, inspired by the same principles of natural evolution. Evolutionary Algorithms are a member of a family of stochastic search techniques that mimic the natural evolution of repeated mutation and selection as proposed by Charles Darwin.

2.2 Multi-Objective Evolutionary Algorithm

The mentioned basic evolutionary algorithms are referred to as single-objective optimization algorithms, since they employ only single-objective selection criteria. Thus, they are suitable to solve single-objective optimization problems, i.e. the result of the optimization is a single solution that minimizes or maximizes a fitness value, which is directly related to a single measure of quality. In many real world applications, the quality of a solution is not only depending on a single objective, but on several, possibly conflicting, objectives. For this kind of multi-objective problems, EAs have been extended by multi-objective selection criteria and elite or archiving strategies to address these extended requirements. Hence, they are referred to as multi-objective evolutionary algorithms (MOEA). The first approaches in this area have been published in [Gol89] and [SD94]. Beside their ability to handle multiple optimization objectives, MOEAs have several additional advantages. One advantage, which becomes important in our application, is that they preserve the diversity in the population of individuals in such a way that a whole set of solutions is maintained during the optimization process, representing niches of high fitness in respect to one optimization objective. This is especially important for the inference problem due to the highly multi-modal solution space. With a larger diversity, the algorithm is able to escape local optima and thus increasing the probability to find the global solution.

Objective Functions For examining the data fitness and the distance to the given pathway in parallel, we use a MOEA, which optimizes the parameters of a mathematical model in respect to the following two optimization objectives:

I.) **Data fitness**: For evaluating the RSE fitness of the individuals, i.e. the similarity of the time dynamics between the experimental and the simulated data, we used the following

equation for calculation of the fitness value, known as the relative squared error (RSE):

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

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 optimization problem is then to minimize the fitness values of objective function f_1.

II.) **Pathway fitness**: The second objective is to minimize the distance between the imported pathway and the topology found by the optimization algorithm. This is done by comparing the edges and their direction of the underlying directed graphs, which represent the pathways:

$$f_2 = \sum_{i=1}^{E} d_i^E \tag{2}$$

$$\text{with } d_i^E = \left\{ \begin{array}{lll} 1 & : & \text{sgn(pathway}_i) \neq \text{sgn(topology}_i) \\ 0 & : & \text{else} \end{array} \right. \tag{3}$$

where pathway is the imported pathway, topology is the topology of the current model, i denotes the ith edge of the directed graph, and E is the number of edges of the fully connected graph.

In the proposed method, we use an EA with hybrid encoding individuals to minimize the objectives, which was recently developed by the authors [SUZ04]. Here, each individual combines a binary and a real valued genotype that are evolved in parallel. The binary variables are used to code the topology or structure of the network and the double encoded optimization variables represent the corresponding model parameters. The real valued genotype gives the kinetic parameters of the mathematical model for the current topology. 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 in the optimization process and are subject to random walk and might be incorporated in the simulation, if the bitset at the corresponding position changes. This enables the optimization algorithm to escape local optima more efficiently.

2.3 Model

On an abstract level, the behavior of a cell is represented by a directed graph with N nodes representing N genes. Each gene g_i produces a certain amount of RNA x_i when expressed

and changes the concentration of the RNA level over time: $\vec{x}(t+1) = h(\vec{x}(t))$, $\vec{x}(t) = (x_1, \cdots, x_n)$. Here, function h represents the changes of the vector of expression levels from one state to the next.

To model and to simulate regulatory networks in the present work, we decided to use S-Systems, since they are well-documented and examined. S-Systems are a type of power-law formalism, which has been suggested in [Sav91] and can be described by a set of nonlinear differential equations:

$$\frac{dx_i(t)}{dt} = \alpha_i \prod_{j=1}^{N} x_j(t)^{\mathcal{G}_{i,j}} - \beta_i \prod_{j=1}^{N} x_j(t)^{\mathcal{H}_{i,j}} \tag{4}$$

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 4 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. With this, the MOEA has to optimize the parameters of $\mathcal{G}$, $\mathcal{H}$, α_i and β_i and the binary interaction matrix in respect to the given objective functions f_1 and f_2.

3   Results

As described before, we used MOEA with hybrid encoding individuals to fit the data resulting from the simulation of an artificial model to minimize the distance to the given pathway. In the present case, the algorithm had to optimize $2(N + N^2)$ real valued parameters for the S-System. Additionally, the bits of the binary genotype representing the topology had to be optimized as well. Here, we decided to model the underlying S-System in more details and used $2N^2$ bits for each entry $\mathcal{G}_{i,j}$ and $\mathcal{H}_{i,j}$ and not only N^2 bits for a simple quadratical interaction matrix. With this, the optimization algorithm had to evolve $2N + 4N^2$ variables in total.

3.1   Gene Network

To verify the algorithm, we applied the proposed method to model the dynamics of a regulatory system examined in [HS96].

Fig. 3.1 shows the network of the dependencies. This gene network consists of two genes (system component 1 and 4). X_1 and X_4 are mRNA concentrations produced by gene 1 and 4, respectively. X_2 is an enzyme translated from X_1, and X_3 is an inducer protein catalyzed by X_2. Component X_5 is a regulator protein translated from X_4. Note that
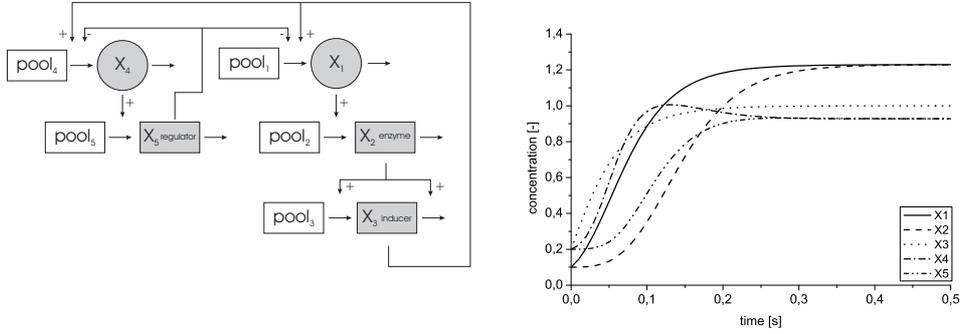
Figure 1: Genetic network and the corresponding time course dynamics of the simulated network introduced in [HS96].

X_3 and X_5 are assumed to supress/activate the mRNA transcription of genes 1 and 4 in negative or positive feed back loops, respectively. The time series data for each gene were created by simulating the S-System representing this GRN with $T = 50$ sampling points. The parameters of the model were also examined in [TKO00, KTA$^+$03] and the time dynamics are given in figure 3.1.

3.2   Settings

The multi-objective optimization runs were performed using an 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 $p_m = 0.1$ on the real-valued genotype and Uniform-crossover recombination with $p_c = 0.8$ and BitFlip-mutation with a mutation probability $p_m = 0.1$ on the binary genotype. Details on the implementation of NSGA-II are given in [DAPM00]. To keep track of the Pareto-front the multi-objective algorithm maintained an archive of (population size$/2 = 250$) individuals and used this archive as elite to achieve a faster convergence. Each multi-run experiment terminated after $1,000,000$ fitness evaluations and the whole setting was repeated 20 times.

Additionally, an extended GA was used as a benchmark algorithm. It represents the class of standard approaches to the inference problem of optimizing the parameters of the mathematical model without incorporation of any pathway information. The extension to the GA was suggested in [TKO00] and is referred to *skeletalizing*. 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, $|\mathcal{G}_{i,j}| < t_{skel} \rightarrow \mathcal{G}_{i,j} = 0.0$. The skeletalizing algorithm has the same total number of parameters to optimize as the MOEA described above. This

benchmark class was implemented as a standard real-coded GA with a population of 500 individuals, crowded tournament selection with a tournament group size of $t_{group} = 8$, Uniform-crossover with $p_c = 1.0$, a mutation probability of $p_m = 0.1$ and a threshold value $t_{skel} = 0.05$.

3.3 Inference

The proposed algorithm was then tested on the problem of inferring the gene regulatory system that was described in the previous section. As test cases we performed experiments with noisy/faulty topological information, to see, whether the algorithm is able to cope with noise and is still able to find the correct solution.

Noisy Pathway In the noisy test case we introduced noise to the imported pathway and thus simulated faulty or incomplete knowledge about the biological process of interest. This was done by randomly changing r entries of the correct topology. To study the ability of our algorithm to cope with noise, we stepwise increased the number of changes r from 1, which corresponds to a noise level of $\frac{1}{25} = 4\%$, with an overall number of interactions of 5 system components of $N^2 = 25$ parameters, to $\frac{5}{25}$, which corresponds to 20% noise in the system. For each noise level, we created 20 random topologies, which were then inferred by the proposed algorithm. The resulting models were then evaluated by comparing the dependencies between the system components.

Fig. 2 shows the fitness values of objective function f_1 (data matching) for the different noise levels $r = 1, \ldots, 5$. The resulting fitness values of the first two test cases ($r = 1$ and $r = 2$) are very close to the values of the initial test case described above. The RSE of the last test case with a high noise level (20% noise) are comparably high, thus not resembling the original time dynamics. However, the results of our algorithm still outperform the skeletalizing GA, which is shown in the figure too.
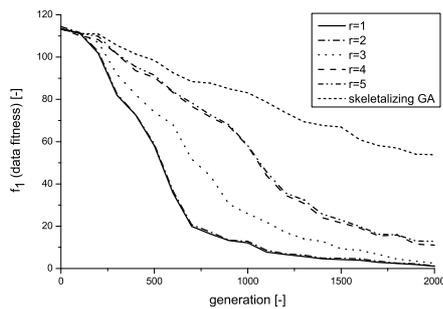
Figure 2: Fitness courses of the first test case with noisy pathway information.
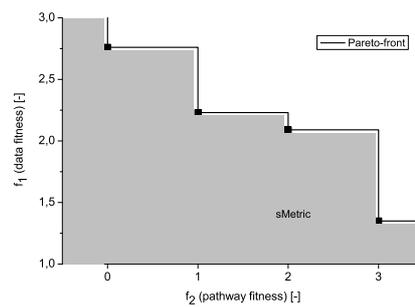
Figure 3: Pareto-front of one solution to the MOEA optimizing process in the test case with $r = 3$ (12% noise).

Table 1: Euclidian distance between the parameters of the true system and those of the model found by the MOEA. For the topology distance, the edges and their direction of the directed graphs representing the pathways were compared. The parameter distance gives the euclidian distance between the true system and the inferred model.

Noise	Topology distance	Parameter distance	Standard deviation
$r = 1$	0	0.15	0.21
$r = 2$	1	0.19	0.28
$r = 3$	2	1.88	1.07
$r = 4$	2	1.95	1.16
$r = 5$	2	3.11	1.87
skel. GA	6	12.49	10.12

To evaluate the quality of the models found in the inference process, we compared them to the true system that was used to create the experimental data. Since it is a multi-objective optimization problem, a biologist would have to choose a trade-off between fitting the current known pathway topology (low fitness of f_2) or a good data fit with an alternative topology (low fitness of f_1). Here, we used the best f_1 solutions for comparison.

The compared true pathway topology is not identical to the noisy pathway that was imported into the inference process. For assessing the differences we used the standard euclidian distance between the parameters p_i in the set of parameters P of the mathematical model:

$$d^P = \sum_{i=1}^{P} (\hat{p}_i - p_i)$$ (5)

Table 1 lists the distances between the true model and the model found by the algorithm resulting from Equation. 5. For the first noisy test cases ($r = 1$ and $r = 2$, 4% and 8% noise, respectively), the algorithm successfully finds the correct topology and is also able to correctly identify the parameters of the model in all multi-runs resulting in good values for the standard deviation. In the test cases with higher noise ($r = 3$ and $r = 4$) our algorithm still finds very good solutions in the multi-runs. But as expected, in the last test case ($r = 5$), the impact of the noise is too high, so that the algorithm is no longer reliably able to find the correct solution in either of the multi-runs. However, the quality of the solution is by far better than the solution of the *skeletalizing* GA, which is also listed in the table.

In the first two noisy test cases almost all multi-runs resulted in only one Pareto-optimal individual, i.e. only one global solution was found that was performing best in both optimization objectives. But as from the third test case, the MOEA resulted in multiple solutions, which were Pareto-optimal, due to the high level of noise. Fig. 3 shows exemplarily the Pareto-front of one run of the MOEA optimizing process in case of $r = 3$. The y-axis in the figure gives the fitness value with respect to objective f_1 (data fitness). The x-axis shows the values of the second objective f_2 (pathway fitness). The distance of this objective function was not calculated with the true system but with the noisy pathway information that was imported.

In this example, four Pareto-optimal individuals were found. Note that the lower fitness value with respect to the data fitness does not correspond to the examined topology indicated by the higher value with respect to the pathway fitness. On the first glance this seems to be contradictory, but as already mentioned, the algorithm is using a topology to evaluate objective function f_2, which is not the true topology due to the noise. In this example, the algorithm finds a good solution with respect to the first objective, but with a pathway mismatch of 3. This corresponds exactly to the level of noise of the test case. Thus, the algorithm was able to determine the correct topology although this topology differs from the noisy pathway information that was imported.

4 Conclusion

In this paper we proposed an algorithm to infer regulatory systems from experimental microarray data with incorporation of pathway information. The most important advantage is that researchers are able to import information for example from public databases like KEGG or from self-modeled research results. Public pathway databases are continuously increasing in size of their contents and thus more and more pathways become available. Unfortunately, the information within the pathway databases is often incomplete and in some cases even faulty. However, the proposed method is able to find correct pathways although the available information is noisy. We showed that the algorithm performs very well on small regulatory networks with noise levels up to 20%, making it very promising to work also on larger network structures. We are currently working on an extended solution representation to increase the performance of the simultaneous optimization of parameters and topologies. This will enable us to examine larger networks in the future. We showed that the proposed algorithm is able to identify network topologies correctly, even in the case of noisy pathway information. In the test cases with a comparably low level of noise, the algorithm identifies the topology and the kinetic parameters of the target system. And even in the test cases with higher level of noise, it is able to outperform state-of-the-art algorithms addressing the problem of sparseness of network connectivity.

Beside the ability of optimizing multiple and sometimes contradictory objectives another advantage of MOEAs is that they preserve the diversity of the population in the solution space. With this, the algorithm is able to escape local optima and thus the probability of finding the global optimum is increasing. The MOEA approach showed the success of combining multiple objectives into one optimization target, thus enabling to use a priori knowledge and constraints to gain better and biologically plausible results. Using several biological constraints also helps to decrease the number of valid network topologies, and thus to narrow the solution space of the inference algorithm. This is especially important because of the ambiguity in the experimental data, i.e. several contradictory network structures result in very similar time dynamics compared to the true system. This issue has already been discussed by the authors of this paper in previous publications.

In future work, we plan to extend the MOEA by adding new objectives like the sparseness of the dependency matrix, since biological networks are known to be only sparsely connected. Additionally, in future experiments we want to focus on different mathematical

models to simulate gene regulatory networks, which might also counteract the increasing number of valid network structures and to overcome the problems with a quadratic number of model parameters of the S-System.

Acknowledgement

References

[DAPM00] 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 Conference on Parallel Problem Solving from Nature*, number 1917 in Lecture Notes in Computer Science, pages 849–858, 2000.

[dJ02] Hidde de Jong. Modeling and Simulation of Genetic Regulatory Systems: A Literature Review. *Journal of Computational Biology*, 9(1):67–103, January 2002.

[Gol89] David E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, 1989.

[Her98] J. Herz. Statistical Issues in reverse engineering of genetic networks. In *Proceedings of the Pacific Symposium on Biocomputing*, 1998.

[Hol75] John H. Holland. *Adaption in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Systems*. University Press of Michigan, 1975.

[HS96] W.S. Hlavacek and M.A. Savageau. Rules for coupled expression of regulator and effector genes in inducible circuits. *Journal of Molecular Biology*, 255:121–139, 1996.

[KG00] Minoru Kanehisa and Susumu Goto. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research*, 28(1):27–30, 2000.

[KIK+05] Shuhei Kimura, Kaori Ide, Aiko Kashihara, Makoto Kano, Mariko Hatakeyama, Ryoji Masui, Noriko Nakagawa, Shigeyuki Yokoyama, Seiki Kuramitsu, and Akihiko Konagaya. Inference of S-system models of genetic networks using a cooperative coevolutionary algorithm. *Bioinformatics*, 21(7):1154–1163, 2005.

[Koz92] John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.

[KTA+03] Shinichi Kikuchi, Daisuke Tominaga, Masanori Arita, Katsutoshi Takahashi, and Masaru Tomita. Dynamic modeling of genetic networks using genetic algorithm and S-system. *Bioinformatics*, 19(5):643–650, 2003.

[OYSOM04] Isao Ono, Ryohei Yoshiaki Seike, Norihiko Ono, and Masahiko 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*, pages 2060–2067, 2004.

[PC04]   Carey Pridgeon and David Corne. Genetic Network Reverse-Engineering and Net-
            work Size; Can We Identify Large GRNs? In *Proceedings of the IEEE Symposium
            on Computational Intelligence in Bioinformatics and Computational Biology*, pages
            32–36, 2004.

[Rec73]   Ingo Rechenberg. *Evolutionsstrategie - Optimierung technischer Systeme nach
            Prinzipien der biologischen Evolution*. Frommann-Holzboog, 1973.

[Sav91]   Michael A. Savageau. 20 years of S-systems. In E.O. Voit, editor, *Canonical Nonlin-
            ear Modeling. S-systems Approach to Understand Complexity*, pages 1–44, 1991.

[Sch81]   Hans-Paul Schwefel. *Numerical optimization of computer models*. John Wiley &
            Sons, 1981.

[SD94]   N. Srinivas and Kalyanmoy Deb. Multiobjective Optimization Using Nondominated
            Sorting in Genetic Algorithms. *Evolutionary Computation*, 2(3):221–248, 1994.

[SSSZ04]   Christian Spieth, Felix Streichert, Nora Speer, and Andreas 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*, volume 1, pages
            146–151, 2004.

[SUZ04]   Felix Streichert, Holger Ulmer, and Andreas Zell. Evaluating a Hybrid Encoding
            and Three Crossover Operators on the Constrained Portfolio Selection Problem. In
            Proceedings of the IEEE Congress on Evolutionary Computation, pages 932–939,
            2004.

[TKB+03]   Yoshinori Tamada, SunYong Kim, Hideo Bannai, Seiya Imoto, Kousuke Tashiro,
            Satoru Kuhara, and Satoru Miyano. Estimating gene networks from gene expres-
            sion data by combining Bayesian network model with promoter element detection.
            Bioinformatics, 19(90002):227ii–236, 2003.

[TKO00]   Daisuke Tominaga, Nobuto Kog, and Masahiro Okamoto. Efficient Numeral Opti-
            mization Technique Based on Genetic Algorithm for Inverse Problem. In *Proceedings
            of the Genetic and Evolutionary Computation Conference*, pages 251–258, 2000.

[WCF+01]   E. Wingender, X. Chen, E. Fricke, R. Geffers, R. Hehl, I. Liebich, M. Krull, V. Matys,
            H. Michael, R. Ohnhauser, M. Pruss, F. Schacherer, S. Thiele, and S. Urbach.
            The TRANSFAC system on gene expression regulation. *Nucleic Acids Research*,
            29(1):281–283, 2001.