A Multiobjective Phenomic Algorithm for Inference of Gene Networks

Rio G.L. D'Souza[1], K. Chandra Sekaran[2], and A. Kandasamy[2]

[1] St Joseph Engineering College, Mangalore, India
[2] National Institute of Technology Karnataka, Surathkal, Mangalore, India
{rio,kchnitk}@ieee.org, kandy@nitk.ac.in

Abstract. Reconstruction of gene networks has become an important activity in Systems Biology. The potential for better methods of drug discovery and of disease diagnosis hinge upon our understanding of the interaction networks between the genes. Evolutionary methods are proving to be successful in such problems and a number of such methods have been proposed. However, all these methods are based on processing of genotypic information. We have presented an evolutionary algorithm for reconstructing gene networks from expression data using phenotypic interactions, thereby avoiding the need for an explicit objective function. Specifically, we have also extended the basic phenomic algorithm to perform multiobjective optimization for gene network reconstruction. We have applied this novel algorithm to the yeast sporulation dataset and validated it by comparing the results to the links found between genes of the yeast genome at the SGD database.

Keywords: Gene networks, Phenomic algorithm, Multiobjective optimization, Evolutionary algorithms, Yeast Sporulation, Microarray data analysis.

1 Introduction

Advances in methods of gene expression measurement have heralded the advent of high throughput methods such as microarray technology. Biologists now can study hundreds of genes at a time, and such studies lead to the elucidation of relationships between genes which ultimately lead to a better understanding of the cellular processes that form the basis of life. However the datasets that result from such studies have high dimensionality. The challenge is to analyze such datasets without compromising their information content. Several researchers have developed methods of analysis which can determine useful patterns from the datasets without compromising the dimensionality [1].

Gene networks represent relationships between genes, based on observations of how the expression level of each gene affects the expression levels of the others [2]. The determination of these relationships from gene expression measurements is a reverse engineering or reconstruction activity [3]. Evolutionary methods have been found to be useful [4] to analyze and capture the relationships between hundreds of genes. The application of new ideas of evolutionary optimization to the inference of

J. Suzuki and T. Nakano (Eds.): BIONETICS 2010, LNICST 87, pp. 440–451, 2012.

gene networks is an ongoing process and many non-conventional methods have shown remarkable success [5]. The Phenomic Algorithm, introduced in [6], and further studied in [7], is one such method. It presents an evolutionary approach based on phenotypic interactions rather than genotypic mechanisms which are used in traditional evolutionary algorithms.

In this paper, we have modified the basic phenomic algorithm to handle multiple objectives. It is possible to employ multiobjective optimization to elucidate gene networks which are more biologically plausible [8]. We have used non-dominated sorting in order to determine the pareto-optimal solutions that best represent the balance between the objectives that we have chosen to optimize. We have applied the multiobjective phenomic algorithm to the yeast sporulation dataset [9] and results show a marked improvement in the quality of networks discovered.

The rest of this paper is organized as follows: In Section 2, we review the related work done by others. We devote Section 3 to a discussion about the methodology adopted by the basic phenomic algorithm and its implementation. We discuss the rationale for modification of the basic phenomic algorithm in Section 4 and its actual implementation in Section 5. Finally, Section 6 presents the results and validation, followed by Section 7 which concludes the paper.

2 Related Work

While early methods for reconstruction of gene networks focused on inferring Boolean networks [10] others have used differential equations [11], [12], [13] and Bayesian networks [14], [15] to infer qualitative, as well as quantitative models of gene networks. Given that gene networks are intrinsically nonlinear and dynamic systems, some researchers [8], [16] have used the S-system proposed by Savageau [17] in order to formulate an objective function for the evolutionary algorithm that they use to reverse engineer gene networks.

State space models [18] and information theoretic approaches [19], [20] have also been successfully applied to the problem of inferring gene networks from microarray data. In recent years machine intelligence based approaches [21], [22], [23] are becoming popular in this area due to their relative ease of application. A number of multiobjective evolutionary algorithms (MOEAs) have been applied to the problem of reconstructing gene networks from expression data [24], [25]. Notable among these algorithms is the non-dominated sorting genetic algorithm (NSGA) and its variations which have been applied to the problem of classification of cancer based on gene expression data [26], [27], [28].

The application of MOEAs to the elucidation of gene networks is an area which is receiving a large amount of focus from researchers due to the perceived benefits in applications such as drug discovery and the diagnosis of chronic diseases. This has been the motivation for the development of the phenomic algorithm [6], [7] which attempts to solve the problem of requiring an explicit fitness function for the optimization process. In this paper, we extend this algorithm to perform multiobjective optimization.

3 Basic Phenomic Algorithm

The basic phenomic algorithm is initialized with a population of individuals. Each individual has genetic information embedded within it. In the phenomic algorithm, we embed the expression of a gene within the individual. When constructing gene networks, we study the relationship between genes. If g_i and g_j are objects representing two such genes, their expression patterns across m samples may be written as $g_i = \{w_{ik} \mid 1 \le k \le m\}$ and $g_j = \{w_{jk} \mid 1 \le k \le m\}$.

When the microarray dataset contains records which represent the expression of each gene at m time-steps (instead of m samples) of an experiment, it is possible to verify whether the expression pattern of a gene g_i at a time-step (t-1) has any correlation with the expression pattern of a gene g_j at time t. For this, we define the Pearson correlation coefficient across time-steps (from gene g_i at time-step $t = (k\text{-}1)$, to gene g_j at time-step $t = k$), as given in Eqn. (1).

$$Pear\,(g_i, g_j) = \frac{\sum_{k=2}^{m} \left(w_{i(k-1)} - \mu_{g_i}\right)\left(w_{jk} - \mu_{g_j}\right)}{\sqrt{\sum_{k=2}^{m} \left(w_{i(k-1)} - \mu_{g_i}\right)^2} \sqrt{\sum_{k=2}^{m} \left(w_{jk} - \mu_{g_j}\right)^2}} \,. \tag{1}$$

In the basic phenomic algorithm random pairs of genes are considered at a time and the proximity measure between them is calculated. Once the proximity measure is calculated, typical gene interactions such as "meet", "know", "like", "dislike", etc. are defined as operations on genes g_i and g_j, as shown in Eqns. (2) to (4).

$$meet(g_i, g_j)\ returns\ TRUE\ \ iff\ \ g_i\ and\ g_j\ were\ partners, at\ least\ once\ . \tag{2}$$

$$know(g_i, g_j)\ returnsTRUE\ \ iff\ \ g_i\ and\ g_j\ are\ part\ of\ the\ same\ subnetwork\ \cdot \tag{3}$$

$$like(g_i, g_j)\ returns\ TRUE\ iff\ Pear(g_i,\ g_j) \le D\ \cdot \tag{4}$$

These operations determine the character of the phenotypic interactions that take place between gene objects. By storing links between genes that "like" each other it is possible to elucidate the relationships that are required for reconstructing the gene network. A brief description of the main features of the basic phenomic algorithm is given below:

1. **Modeling Genes as Individuals:** While modeling the genes as individuals, the expression profile of the gene is embedded within the object itself. Also the relationships with other genes which are discovered during the interaction phase are stored within the individual itself.
2. **Simulating Gene Interaction:** The stage is set for the survival-of-the-fittest by letting individuals to meet randomly. Eqns. (2) to (4) define the typical nature of these interactions between partners that meet.

Procedures for implementing the typical interaction criteria given in Eqns. (2) to (4).

```
meet($g_i$, $g_j$)
(
if($g_i$.MET[$g_j$])
          return TRUE;
else
          {
          if(!know($g_i$, $g_j$) and like($g_i$, $g_j$))
              link($g_i$, $g_j$);
          set $g_i$.MET[$g_j$] = TRUE;
            }
}

know($g_i$, $g_j$)
(
if($g_i$.LINK[$g_j$])
           return TRUE;
}

like($g_i$, $g_j$)
(
if(Pear($g_i$, $g_j$) $\leq$ $D$)
          return TRUE;
}
```

3. Enforcing Natural Processes: From time to time the population is consolidated by eliminating individuals which are replicates and have not acquired any links with other individuals. At the end of the process, the links between the genes, which are stored in the individuals, are used to construct the gene networks.

The structure of the basic phenomic algorithm is very similar to a genetic algorithm since phenotypic processing is encountered in every generation. Interested readers may refer to D'Souza et al. [6], [7] for further details of this algorithm. In the following sections, we have modified the basic phenomic algorithm to handle multiple objectives for optimizing the inference of gene networks.

4 Multiobjective Optimization

The inference of gene networks from microarray data is a problem where multiple, and often conflicting, objectives come into play. In this section, the fundamental concepts of MOEAs are presented and thereafter, multiple objectives are chosen for optimizing the inference of gene networks.

4.1 Multiobjective Evolutionary Algorithms

Early evolutionary algorithms were focused on optimizing single objectives. However, most optimization problems have multiple objectives. Optimization of multiple objectives requires that the relative importance of each objective be specified in advance which requires a prior knowledge of the possible solutions. But, by using the concept of Pareto-dominance, it is possible to avoid the need to know the possible solutions in advance. This is one of the reasons for the popularity of such Pareto-based approaches.

Before applying Pareto-based multiobjective optimization, some of the relevant concepts are defined and discussed. Consider the following *m*-objective minimization problem [29]:

$$\begin{aligned}&\min F(X),\\ &F=\{f_1(X),f_2(X),\ldots,f_m(X)\}\end{aligned}. \tag{5}$$

Where $f_1, f_2,\ldots, f_m$ are the *m* objectives. *F(X)* could as well have been a maximization problem, but it is arbitrarily chosen to discuss from a minimization perspective.

1. Dominance: A solution *X* is said to dominate a solution *Y* if $\forall j = 1, 2, \ldots, m$, $f_j(X) \le f_j(Y)$, and there exists $k \in \{1, 2, \ldots, m\}$ such that $f_k(X) < f_k(Y)$.

2. Pareto-Optimal Solutions: Solution *X* is called Pareto-optimal if it is not dominated by any other feasible solutions. Pareto-optimal, or non-dominated, solutions are those solutions which do not dominate each other, i.e., neither of them is better than the other in all the objective function evaluations.

3. Pareto-Front: The locus that is formed by a set of solutions that are equally good when compared to other solutions of that set is called as a Pareto-front. The solutions on each pareto-front are pareto-optimal with respect to each other.

In the next section, suitable objectives are selected based on the current knowledge of the biological properties of gene networks.

4.2 Multiple Objectives for Optimization

It is well known that most biological networks display the small-world network property that predicates sparseness between key nodes and dense local connections around each key node. This definition of small-world networks was offered by Spieth et al. [8]. In a conventional multiobjective evolutionary algorithm, the similarity of the target network to small-world networks could be used as an objective in order to determine the network that has the optimal number of links. Also, since the intention is to find as many links as possible, the number of links discovered could be used as the other objective. The two objectives are formally defined as Number of Links (NOL) in Eqn. (6) and Small-World Similarity Factor (SWSF) in Eqn. (7):

$$NOL=\sum_{i=1}^{N} l_{ij} \quad . \tag{6}$$

$$SWSF=\frac{1}{C}\sum_{k=1}^{C} k.n_k \quad . \tag{7}$$

where l_{ij} = 1 if gene g_i is linked to gene g_j, else l_{ij} = 0 (taken from the adjacency matrix of the network), N is the total number of genes in the target network, n_k is the number of nodes with out-degree of k, and C is the maximum cardinality of the set of genes that can influence any given gene. While optimizing, the objective NOL is maximized, whereas the objective SWSF is minimized. It should be noted here that the algorithms based on the phenomic approach do not directly evaluate solutions using Eqns. (6) and (7). These equations are given so that they can be used in other MOEAs whose results will be compared with the results of the phenomic algorithms. In the multiobjective phenomic approach the two objectives are realized indirectly as follows:

1. Multiobjective Screening: The objective NOL is implemented here by screening out duplicates without losing captured links. In multiobjective screening, the two individuals that meet check if their gene ID is the same. If so, the links captured by both the individuals up to that point are all copied into one of the individuals and the other is deleted. Thus the average number of links-per-gene will go on improving from one generation to the next.

The multiobjective screening procedure.

```
multiobjective_screening(g_i, g_j)
(
if(g_i.ID = g_j.ID)
          appendLinks(g_i, g_j);
          delete(g_j);
}
```

2. Multiobjective Phasing: The SWSF objective is implemented here by introducing a two-phased process. Initially, when two genes with dissimilar gene IDs meet, they are allowed to link without restriction, based on typical interaction criteria given in Eqns. (2) to (4). However, after a certain number of interactions (which is a parameter set at the beginning of the run) a pair of genes is allowed to link only if the first one has more links. Since, at any given time, all genes will not have undergone the same number of interactions; there will be many genes which have more links than the others. The net effect is that some key nodes will capture many more links than the others and most nodes will have very few links.

The multiobjective phasing procedure.

```
multiobjective_phasing($g_i$, $g_j$)
(
if(PHASE = 1)
         meet($g_i$, $g_j$);
else
         if(PHASE = 2)
             if($g_i$.LINKS > $g_j$.LINKS)
               meet($g_i$, $g_j$);
}
```

There is no need of fitness functions since there is no explicit fitness evaluation and only individuals that are fitter than others survive into the next generation.

5 The Multiobjective Phenomic Algorithm

The multiobjective phenomic algorithm is initialized in the same manner as the basic phenomic algorithm.

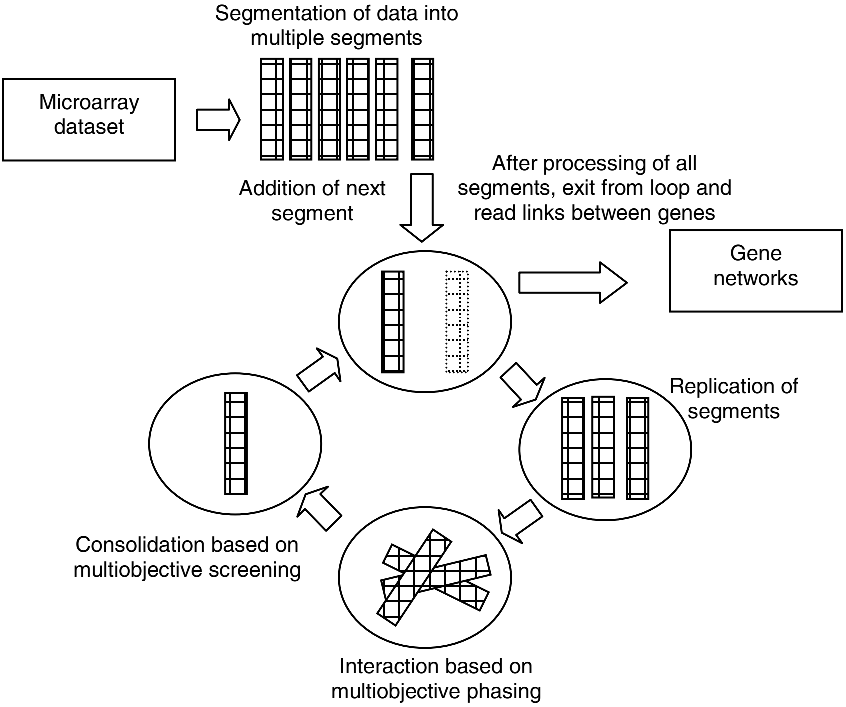
Segmentation of data into multiple segments
Microarray dataset
Addition of next segment
After processing of all segments, exit from loop and read links between genes
Gene networks
Replication of segments
Consolidation based on multiobjective screening
Interaction based on multiobjective phasing

Fig. 1. Sequence of processing in the Multiobjective Phenomic Algorithm

As shown in Fig. 1, most of the sequence of processing and functions are the same as in the basic version, except for the changes that introduce multiobjective considerations. As explained in the previous section, the objective SWSF is achieved by applying multiobjective phasing criteria at the interaction phase of the algorithm. Also, the objective NOL is achieved by introducing multiobjective screening (described in the previous section) in the consolidation phase of the algorithm. The features of the basic algorithm, such as segmentation, interaction and consolidation, which contribute to the scalability and robustness of the algorithm, are retained in this multiobjective version.

The consensus network that is formed in the consolidation phase is just the simple union of all the links in the two individuals being consolidated at that point. It should be pointed out here that in the basic version of the algorithm, the links in one of the individuals were lost when one of them was randomly deleted. Retaining all the links in that algorithm would have led to a proliferation of spurious links since there was no mechanism to restrict the growth of links. In the current version, the interaction phase employs multiobjective phasing which, as explained in the previous section, limits the growth of links after a certain stage.

The multiobjective phenomic algorithm and its main functions.

```
multiobjective_phenomic_algorithm( )
(
divide gene expression data into segments;
initialize population with first segment replicated;
set segment count to 0;

while population has not reduced to size of single
segment and there are more segments to process
   {
   interact_population;
   consolidate_population;
   replicate and add next segment;
   increment segment count;
   }

read gene-links stored in the final population;
display gene networks constructed from links;
}

interact_population( )
{
for a preset number of iterations
   {
 randomly select two individuals from population;
 apply multiobjective phasing interaction criteria;
   }
}

consolidate_population( )
{
for a preset number of iterations
    {
    randomly select two individuals from population;
    apply multiobjective screening criteria;
    }
}
```

The results obtained from this algorithm and its performance are discussed in the next section.

6 Results and Discussion

The Multiobjective Phenomic Algorithm (MPA) was run on the Yeast sporulation dataset [9]. The expression profiles of 6118 genes are included in this dataset. From these profiles, only those that show a 2.2-fold increase in mRNA levels were extracted by Chu et al. [9]. Among them, finally, only the 45 genes were found to be significant by Kupiec et al. [30].

In Fig. 2, a typical gene network inferred by the MPA is shown. It is only one of the networks that were inferred in that run. Each run of the MPA infers anywhere between 10 to 15 networks. In the network of Fig. 2, for example, node 17 is shown to have a large number of links. This node represents the gene RFA1, which indeed is a crucial yeast gene. As per the SGD database [31], RFA1 is a "subunit of hetero-trimeric Replication Protein A (RPA), which is a highly conserved single-stranded DNA binding protein involved in DNA replication, repair, and recombination." Upon verification with the 220 interactions given in the SGD database, it was found that all the links shown in the network are true links.

In Fig. 3, the value of D was set at 0.02. The number of links can be seen to be much higher. The links were verified by looking up the SGD database and most links were found to be valid. However, it was noticed that a few false positives had crept it at this stage. The comparison of inference methods developed by others was restricted to the following multiobjective evolutionary algorithms:

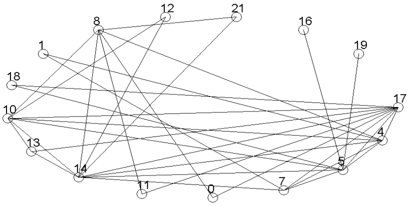

Fig. 2. A gene network inferred by MPA when D = 0.015

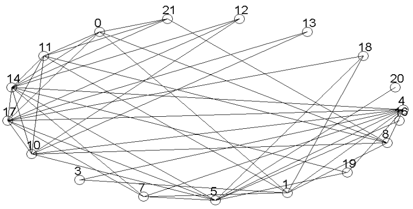

Fig. 3. A Gene Network Inferred by MPA when D = 0.02

1. Non-Dominated Sorting Genetic Algorithm Based Method (NSGA-Based): The NSGA due to Deb [25] was further improved by Deb et al. [26], [27]. In this algorithm, non-dominated sorting is performed on combined parent and offspring population to assign ranks to all the solutions. Based on these ranks solutions are copied over to the next generation. The NSGA is one of the standard MOEAs and therefore it was incorporated into a gene network inference algorithm developed by Spieth et al. [32]. This gene inference algorithm is based on the S-system model and uses Relative Squared Error (RSE) to evaluate the goodness-of-fit between model and the underlying data.

2. Memetic Algorithm (MA-Based): The memetic algorithm developed by Spieth et al. [33] uses the S-systems model and a memetic search procedure for inference of gene networks. A genetic algorithm evolves the topology of the gene networks, while the S-system parameters are evolved through the memetic search procedure.

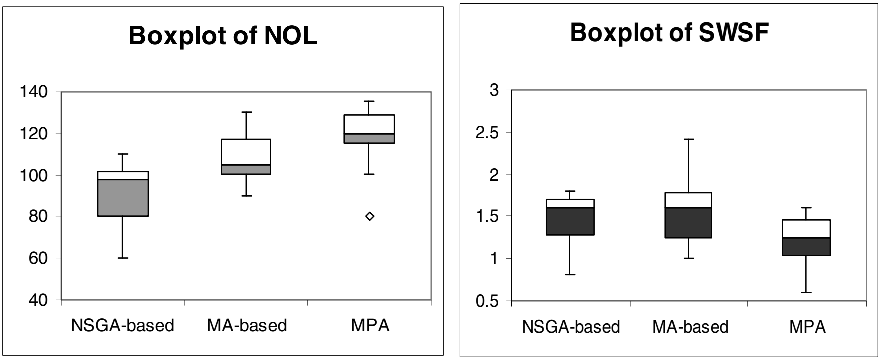

Fig. 4. Boxplots comparing NOL and SWSF of networks inferred by three algorithms being compared

In both the algorithms described above, in order to increase the relevance of gene networks inferred, the objectives defined in the previous section are used. The Number Of Links (NOL) and Small-World Similarity Factor (SWSF), which were defined in Eqn. (6) and Eqn. (7) are used as objectives, in addition to RSE, to perform multiobjective optimization. As seen from the boxplots in Fig. 4, MPA infers better networks than both the NSGA-based and MA-based methods.

7 Conclusion

We have presented the reconstruction of gene networks using the multiobjective phenomic algorithm and also presented results obtained when running the algorithm on the yeast sporulation dataset. The phenomic nature of the algorithm is manifested in its focus on the phenotypic, rather than genetic, information of an individual. Due to the implicit survival-of-the-fittest mechanisms the need for an explicit objective function was avoided.

Currently we are working on applying this algorithm to other datasets in order to study its effectiveness as optimization tool for inference of gene networks.

References

1. Schulze, A., Downward, J.: Navigating gene expression using microarrays - a technology review. Nature Cell Biology 3, E190–E195 (2001)
2. Soinov, L.A., Krestyaninova, M.A., Brazma, A.: Towards reconstruction of gene networks from expression data by supervised learning. Genome Biology 4(1), R6 (2003)
3. Bansal, M., Belcastro, V., Impiombato, A.A., di Bernardo, D.: How to infer gene networks from expression profiles. Mol. Syst. Biol. 3, 78 (2007), doi:10.1038/msb4100120
4. D'haeseleer, P., Liang, S., Somogyi, R.: Gene expression analysis and genetic network modelling: Tutorial. In: Pacific Symposium on Biocomputing (1999)
5. Siegal, M.L., Promislow, D.E.L., Bergman, A.: Functional and evolutionary inference in gene networks: does topology matter? Genetica 129(1), 83–103 (2007)
6. D'Souza, R.G.L., Chandra Sekaran, K., Kandasamy, A.: A phenomic algorithm for reconstruction of gene networks. In: IV International Conference on Computational Intelligence and Cognitive Informatics, CICI 2007, pp. 53–58. WASET, Venice (2007)
7. D'Souza, R.G.L., Chandra Sekaran, K., Kandasamy, A.: Reconstruction of gene networks using phenomic algorithms. Intl. Journal of Artificial Intelligence Applications (IJAIA) 1(2), 1–11 (2010), doi:10.5121/ijaia.2010.1201, ISSN: 0976-2191
8. Spieth, C., Streichert, F., Speer, N., Zell, A.: Optimizing Topology and Parameters of Gene Regulatory Network Models from Time-Series Experiments. In: Deb, K., Tari, Z. (eds.) GECCO 2004, Part I. LNCS, vol. 3102, pp. 461–470. Springer, Heidelberg (2004)
9. Chu, S., DeRisi, J., Eisen, M., et al.: The transcriptional program of sporulation in budding yeast. Science 282, 699–705 (1998)
10. Somogyi, R., Fuhrman, S., Askenazi, M., Wuensche, A.: The gene expression matrix: towards the extraction of genetic network architectures. In: Proc. of Second World Cong. of Nonlinear Analysts (WCNA 1996), vol. 30(3), pp. 1815–1824 (1997)
11. Christley, S., Nie, Q., Xie, X.: Incorporating existing network information into gene network inference. PLoS ONE 4(8), e6799 (2009), doi:10.1371/journal.pone.0006799
12. Liu, B., de la Fuente, A., Hoeschele, I.: Gene network inference via structural equation modeling in genetical genomics experiments. Genetics 178, 1763–1776 (2008)
13. Qian, L., Wang, H., Dougherty, E.R.: Inference of noisy nonlinear differential equation models for gene regulatory networks using genetic programming and Kalman filtering. IEEE Trans. on Signal Processing 56(7), 3327–3339 (2008)
14. Numata, K., Imoto, S., Miyano, S.: A structure learning algorithm for inference of gene networks from microarray gene expression data using Bayesian networks. In: Proc. of the 7th IEEE Intl. Conf. on Bioinfo. and Bioengg. 2007 (BIBE 2007), pp. 1280–1284 (2007)

15. Ko, Y., Zhai, C., Rodriguez-Zas, S.: Inference of gene pathways using mixture Bayesian networks. BMC Systems Biology 3, 54 (2009), doi:10.1186/1752-0509-3-54
16. Noman, N., Iba, H.: Reverse engineering genetic networks using evolutionary computation. Genome Informatics 16(2), 205–214 (2005)
17. Savageau, M.A.: Power-law formalism: a canonical nonlinear approach to modelling and analysis. In: Proc. of the World Congress of Nonlinear Analysts 1992, pp. 3323–3334 (1995)
18. Hirose, O., Yoshida, R., Imoto, S., Yamaguchi, R., Higuchi, T., Charnock-Jones, D.S., Print, C., Miyano, S.: Statistical inference of transcriptional module-based gene networks from time course gene expression profiles by using state space models. Bioinformatics 24(7), 932–942 (2008), doi:10.1093/bioinformatics/btm639
19. Dougherty, J., Tabus, I., Astola, J.: Inference of gene regulatory networks based on a universal minimum description length. EURASIP Journal on Bioinformatics and Systems Biology (2008), doi:10.1155/2008/482090
20. Chaitankar, V., Ghosh, P., Perkins, E.J., Gong, P., Deng, Y., Zhang, C.: A novel gene network inference algorithm using predictive minimum description length approach. BMC Syst. Biol. 4(suppl. 1) (2010), doi:10.1186/1752-0509-4-S1-S7
21. Kentzoglanakis, K., Poole, M.: Gene network inference using a swarm intelligence framework. In: Proc. of the 11th Annual Conf. Companion on Genetic and Evolutionary Computation Conference (GECCO 2009), pp. 2709–2712 (2009)
22. Xu, R., Wunsch, D.C., Frank, R.L.: Inference of genetic regulatory networks with recurrent neural network models using particle swarm optimization. IEEE/ACM Trans. on Computational Biology and Bioinformatics 4(4), 681–692 (2007)
23. Zarnegar, A., Vamplew, P., Stranieri, A.: Inference of gene expression networks using memetic gene expression programming. In: Mans, B. (ed.) Proc. of the 32nd Australasian Computer Science Conf. (ACSC 2009), Conferences in Research and Practice in Information Technology (CRPIT), vol. 91 (2009)
24. Van Veldhuizen, D.A., Lamont, G.B.: Multiobjective evolutionary algorithms: analyzing the state-of-the-art. Evolutionary Computation 8(2), 125–147 (2000)
25. Deb, K.: Multi-objective optimization using evolutionary algorithms. Wiley, Chichester (2001)
26. Deb, K., Reddy, A.R.: Classification of two-class cancer data reliably using evolutionary algorithms. Publ. of Kanpur Genetic Algorithms Lab., India, Report No. 2003001 (2003)
27. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans. Evol. Computation 6(2), 182–197 (2002)
28. Kumar, P.K., Sharath, S., D'Souza, R.G., Chandra Sekaran, K.: Memetic NSGA—A multi-objective genetic algorithm for classification of microarray data. In: 15th Intl. Conf. on Advanced Computing and Communications, ADCOM, pp. 75–80. IEEE (2007)
29. Jin, Y., Sendhoff, B.: Pareto-based multiobjective machine learning: An overview and case studies. IEEE Trans. on Systems, Man, and Cybernetics 38(3), 397–415 (2008)
30. Kupiec, M., Ayers, B., Esposito, R.E., Mitchell, A.P.: The molecular and cellular biology of the yeast Saccharomyces. Cold Spring Harbour, 889–1036 (1997)
31. SGD project: Saccharomyces genome database (2007), `http://www.yeastgenome.org/` (September 15, 2007)
32. Spieth, C., Streichert, F., Speer, N., Zell, A.: Multi-Objective Model Optimization for Inferring Gene Regulatory Networks. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 607–620. Springer, Heidelberg (2005)
33. Spieth, C., Streichert, F., Speer, N., Zell, A.: A memetic inference method for gene regulatory networks based on s-systems. In: Proc. of Congress on Evolutionary Computation (CEC 2004), Proc. Part I, pp. 152–157. IEEE Press (2004)