4

An Introduction to Multi-Objective Evolutionary Algorithms and Some of Their Potential Uses in Biology

Antonio López Jaimes and Carlos A. Coello Coello

CINVESTAV-IPN
Evolutionary Computation Group (EVOCINV)
Departamento de Computación
Av. IPN No. 2508
Col. San Pedro Zacatenco
México, D.F. 07360
MEXICO
tonio.jaimes@gmail.com, ccoello@cs.cinvestav.mx

Summary. This chapter provides a brief introduction to the use of evolutionary algorithms in the solution of problems with two or more (normally conflicting) objectives (called "multi-objective optimization problems"). The chapter provides some basic concepts related to multi-objective optimization as well as a short description of the main features of the multi-objective evolutionary algorithms most commonly used nowadays. In the last part of the chapter, some applications of multi-objective evolutionary algorithms in Biology (mainly within Bioinformatics) will be reviewed. The chapter will conclude with some promising paths for future research, aiming to identify areas of opportunity for those interested in the intersection of these two disciplines: multi-objective evolutionary algorithms and Biology.

4.1 Introduction

Many real-world problems in most disciplines have two or more objectives that we aim to optimize at the same time. Such problems are called "multi-objective", and their solution implies finding good trade-offs among the objectives. Traditionally, multi-objective optimization problems have been dealt with using a variety of mathematical programming techniques that have been developed over the years [34, 63]. However, in recent years, the use of

A.L. Jaimes and C.A. Coello Coello: *An Introduction to Multi-Objective Evolutionary Algorithms and Some of Their Potential Uses in Biology*, Studies in Computational Intelligence (SCI) **122**, 79–102 (2008)
www.springerlink.com

metaheuristics[1] to solve such problems has become increasingly popular (see for example [22, 28]).

Evolutionary algorithms (EAs) are a metaheuristic inspired on the "survival of the fittest" principle from Darwin's evolutionary theory [39]. EAs have become very popular as multi-objective optimizers because of their ease of use (and implementation) and flexibility (e.g., EAs are less sensitive than mathematical programming techniques to the initial search points and to the specific features of a problem). Additionally, the fact that EAs are population-based techniques makes it possible to simultaneously manage a set of solutions, instead of one at a time, as normally happens with mathematical programming techniques.

Multiobjective evolutionary algorithms (MOEAs) date back to the mid-1980s [47, 83], although they became popular in the mid-1990s. Today, it is possible to find applications of MOEAs in practically every discipline, including biology [18].

The rest of this chapter is organized as follows. In Section 4.2, we provide some basic multi-objective optimization concepts required to make the chapter self-contained. Section 4.3 contains a brief description of the main MOEAs in current use. Section 4.4 contains a survey of some of the most representative applications of MOEAs in biology. Section 4.5 indicates some potential paths for future research in this area. Finally, our conclusions are provided in Section 4.6.

4.2 Basic Concepts

This chapter deals with the solution of the **Multiobjective Optimization Problem** (MOP) (also called multicriteria optimization, multiperformance or vector optimization problem), which can then be defined (in words) as the problem of finding [69]:

> "a vector of decision variables which satisfies constraints and optimizes a vector function whose elements represent the objective functions. These functions form a mathematical description of performance criteria which are usually in conflict with each other. Hence, the term "optimize" means finding such a solution which would give the values of all the objective functions acceptable to the decision maker."

The **decision variables** are the numerical quantities for which values are to be chosen in an optimization problem. In most optimization problems there are always restrictions imposed by the particular characteristics of the

[1] A **metaheuristic** is a high level strategy for exploring search spaces by using different methods [7]. Metaheuristics have both a diversification (i.e., exploration of the search space) and an intensification (i.e., exploitation of the accumulated search experience) procedure.

environment or available resources (e.g., physical limitations, time restrictions, etc.). These restrictions must be satisfied in order to consider a certain solution acceptable. All these restrictions in general are called **constraints**, and they describe dependences among decision variables and constants (or parameters) involved in the problem.

In multiobjective optimization, the goal is to optimize a set of objective functions (i.e., more than two) simultaneously. Thus, in this context, the notion of "optimum" changes, because in MOPs, the aim is to find good compromises (or "trade-offs") rather than a single solution as in global optimization (in which we aim to optimize a single objective function). The notion of "optimum" most commonly adopted is that originally proposed by Francis Ysidro Edgeworth [33] and later generalized by Vilfredo Pareto [72]. Although some authors call this notion the *Edgeworth-Pareto optimum*, the most commonly accepted term is *Pareto optimum*.

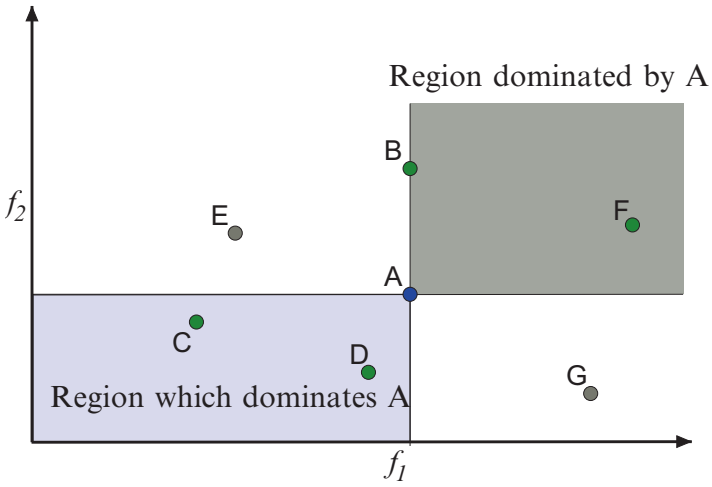

Fig. 4.1. Graphical representation of some solutions that are dominated and others that dominate a reference point (**A** in this case). Note that both **E** and **G** are **nondominated** with respect to **A**. So, **A** is better than **B** or **F**, but it is equally good than **E** or **G**.

A solution is Pareto optimal if there exists no other feasible solution (i.e., one which satisfies all the constraints of the problem) which would decrease some criterion without causing a simultaneous increase in at least one other criterion (assuming minimization). The vectors corresponding to these Pareto optimal solutions are called **nondominated**. Figure 4.1 provides a graphical representation of solutions that dominate and solutions that are dominated by a reference point for a problem with two objective functions. When plotted in objective function space, these nondominated vectors are collectively known as

the **Pareto front** (Figure 4.2 shows the graphical representation of a Pareto front).

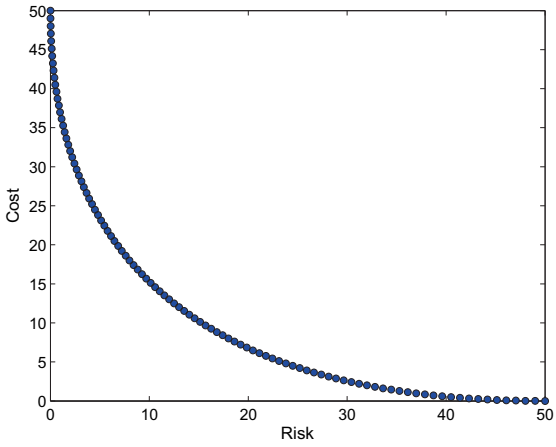

Fig. 4.2. Pareto front of a hypothetical problem with two objectives: risk and cost.

Although it is normally assumed that a MOEA will generate the entire Pareto front (or as many elements of it, as possible), in practice the entire front is rarely needed. This can be easily understood with an example. In Figure 4.2 the solutions lying at the extreme right of the Pareto front represent the lowest possible cost, but with the highest risk. Conversely, solutions lying at the top left of the Pareto front, represent the lowest possible risk, but with the highest cost. Normally, solutions that represent the best possible trade-offs among the objectives are the aim of the search (in the case of Figure 4.2, solutions lying on the "knee" of the Pareto curve).

4.3 MOEAs in Current Use

Although the first reference on the use of EAs for solving multi-objective problems dates back to the late 1960s [80], the first actual implementations was introduced in the mid-1980s [47, 82, 83]. For several years (up to the first half of the 1990s), most of the MOEAs developed had a relatively simple design and were mostly based on linear aggregating functions [88], lexicographic ordering [38], and target-vector approaches [15, 98].

However, four MOEAs are considered the most representative of this early period: the Vector Evaluated Genetic Algorithm (VEGA) [83], the Multi-Objective Genetic Algorithm (MOGA) [37], the Niched-Pareto Genetic Algorithm (NPGA) [42], and the Nondominated Sorting Genetic Algorithm

(NSGA) [86]. Details of these algorithms can be found in their original publications and in other sources (see for example [17]).

During the mid-1990s, elitism was formally introduced in MOEAs [104], and became a standard mechanism for the algorithms developed since then. In single-objective EAs, elitism simply consists of retaining the best individual from the current generation, and passing it without any changes to the next generation. In contrast, in multi-objective optimization, elitism involves retaining the solutions that are nondominated with respect to all the individuals that have been evaluated so far. Thus, instead of retaining only one individual, several must be kept. This introduces additional issues that need to be taken into account (e.g., should we bound the number of individuals to be retained? If so, how do we decide which individuals must be removed?). Elitism is an important mechanism, not only because it allows to keep the globally nondominated individuals (as opposed to handling only the locally nondominated individuals, as done with early MOEAs), but also because it is a requirement to prove convergence [96].

Despite the high number of elitist MOEAs developed from the mid-1990s to date (see for example [5, 21, 23, 24, 94, 100, 102]), three of them are normally considered to be the most representative in the current literature[2]:

1. The **Strength Pareto Evolutionary Algorithm** (SPEA): Developed by Zitzler and Thiele [104], this approach integrates ideas from the different MOEAs previously mentioned (i.e., MOGA [37], NPGA [42] and NSGA [86]). SPEA incorporates elitism through the usage of an external archive containing nondominated solutions previously found (the so-called external nondominated set). At each generation, nondominated individuals are copied to this external nondominated set, and are retained only if they are nondominated with respect to the contents of the set. If they dominated any individuals previously stored in the external set, such dominated individuals are deleted. For each individual in this external set, a *strength* value is computed. This strength is similar to the ranking value of MOGA [37], since it is proportional to the number of solutions to which a certain individual dominates. The fitness of each member of the current population is computed according to the strengths of all external nondominated solutions that dominate it (i.e., the external set plays a role in the selection process). The fitness assignment process of SPEA considers both closeness to the true Pareto front and even distribution of solutions at the same time. However, instead of using niches based on distance (as done in earlier MOEAs such as MOGA [37]), Pareto dominance is used to ensure that the solutions are properly distributed along the Pareto front. SPEA does not require a niche radius, but its effectiveness relies on the size of the external nondominated set. In fact, since the external nondominated set

[2] For more information on MOEAs, interested readers can refer to the EMOO repository, which is located at: `http://delta.cs.cinvestav.mx/~ccoello/EMOO/`

participates in the selection process of SPEA, if its size grows too large, it might reduce the selection pressure, thus slowing down the search. Because of this, the authors decided to adopt a technique that prunes the contents of the external nondominated set so that its size remains below a certain threshold. In 2001, a revised version of SPEA (called **SPEA2**) was introduced. SPEA2 has three main differences with respect to its predecessor [103]: (1) it incorporates a fine-grained fitness assignment strategy which takes into account for each individual the number of individuals that dominate it and the number of individuals by which it is dominated; (2) it uses a nearest neighbor density estimation technique which guides the search more efficiently, and (3) it has an enhanced archive truncation method that guarantees the preservation of boundary solutions.

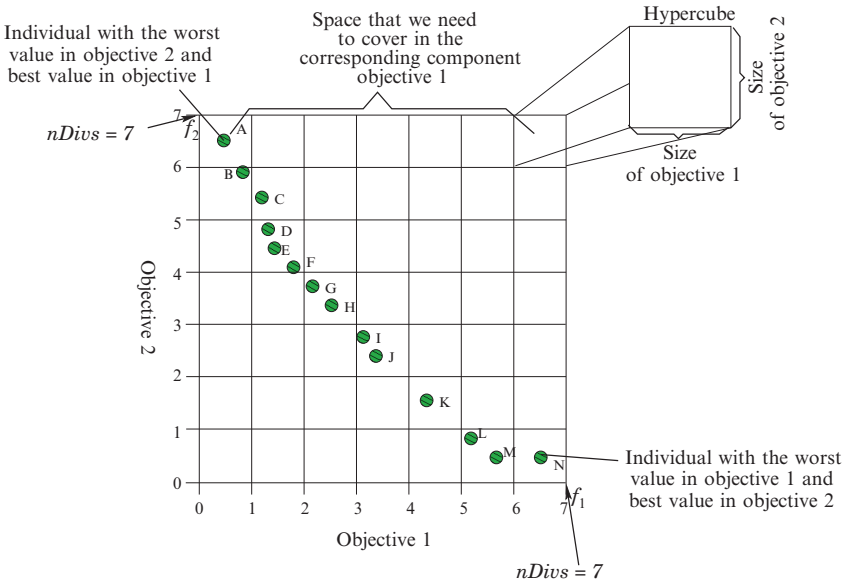

Fig. 4.3. Graphical illustration of the adaptive grid used by PAES.

2. The **Pareto Archived Evolution Strategy** (PAES): Developed by Knowles and Corne [52, 53], this is probably the most simple MOEA that can be conceived. It consists of a (1+1) evolution strategy (i.e., a single parent that generates a single offspring), combined with an external archive that records the nondominated solutions found along the search. As in SPEA, this external archive is used to compare each new individual produced. An interesting aspect of PAES is its procedure to maintain diversity which consists of a crowding mechanism that divides objective space in a recursive manner. Each solution is placed in a certain

grid location based on the values of its objectives (which are used as its "coordinates" or "geographical location") as indicated in Figure 4.3. A map of such grid is maintained, indicating the number of solutions that reside in each grid location.

3. The **Nondominated Sorting Genetic Algorithm II** (NSGA-II): This MOEA is described in Deb et al. [29, 31], and it consists of a considerably improved version of the NSGA [86]. The NSGA-II estimates the density of solutions surrounding a particular solution in the population by computing the average distance of two points on either side of this point along each of the objectives of the problem. This value is called *crowding distance* and its computation is not only efficient, but requires no extra parameters. During selection, the NSGA-II uses a crowded-comparison operator which takes into consideration both the nondomination rank of an individual in the population and its crowding distance (i.e., nondominated solutions are preferred over dominated solutions, but between two solutions with the same nondomination rank, the one that resides in the less crowded region is preferred). This introduces a total ordering (instead of the partial ordering that traditional Pareto ranking generates), and facilitates the selection process. That is the reason why the NSGA-II combines the population of parents with the population of offspring and selects the best half of them. This sort of selection scheme is implicitly elitist and, therefore, no external archive is required in this case. Due to its ease of use, efficacy, and efficiency, the NSGA-II has become a landmark against which other MOEAs are often compared.

4. **Coevolutionary MOEAs**: In evolutionary computation, the term **coevolution** is used to refer to a change in the genetic composition of a species (or group of species) as a response to a genetic change of another one. In a more general sense, coevolution refers to a reciprocal evolutionary change between species that interact with each other. The term "coevolution" is usually attributed to Ehrlich and Raven who published a paper on their studies performed with butterflies and plants in the mid-1960s [35]. The relationships between the populations of two different species can be described considering all their possible types of interactions. Such interaction can be positive or negative depending on the consequences that such interaction produces on the population. Evolutionary computation researchers have developed several coevolutionary approaches in which normally two or more species relate to each other in different forms [70]. The key issue in these coevolutionary algorithms is that the fitness of an individual in a population depends on the individuals of a different population. In fact, we can say that an algorithm is coevolutionary if it has such property.

There are two main classes of coevolutionary algorithms in the evolutionary computation literature:

a) Those based on competition relationships (called **competitive coevolution**): In this case, the fitness of an individual is the result of a series of "encounters" with other individuals [71, 81]. This sort of coevolutionary scheme has been normally adopted for games.

b) Those based on cooperation relationships (called **cooperative coevolution**): In this case, the fitness of an individual is the result of a collaboration with individuals of other species (or populations) [74, 77]. This sort of coevolutionary scheme has been normally adopted for solving optimization problems.

A variety of coevolutionary MOEAs have been proposed in the specialized literature (see for example [3, 20, 46, 50, 51, 58, 61, 73, 89, 90]), but a detailed description of them is beyond the scope of this chapter. Interested readers may refer to Chapter 3 in [19] for more information on this topic.

4.4 Applications of MOEAs in Biology

The use of MOEAs in Biology has raised an increasing interest in the last few years, mainly within Bioinformatics [40, 65]. An analysis of the literature shows five main types of applications of MOEAs in Biology:

1. **System optimization:** This refers to some applications in which it is of interest to determine the degree of optimality of a certain biological system.

2. **Classification:** A wide variety of problems in bioinformatics rely on performing classification tasks (either supervised, unsupervised or combinations of both).

3. **Sequence and structure alignment:** Here, the aim is to assess the structural similarities between a certain macromolecule and a sequence available from a database. The search is done through a series of alignments.

4. **Structure prediction and design:** In this case, the goal is to predict the structure of a macromolecule, given that the function properties of macromolecules derive from their three-dimensional shape. This three-dimensional shape is, in turn, mainly determined by the sequence of bases or amino acids.

5. **Inverse problems:** These are problems in which we have certain information that was generated by a biological process and our goal is to infer the original system using such available information.

Next, we will briefly review some of the most representative work within each of these types of applications.

4.4.1 System optimization

A single nucleotide polymorphism (SNP) is a variation that occurs at only one single nucleotide of two deoxyribonucleic acid (DNA) sequences (e.g., GA**A**CCT and GA**G**CCT). Geneticists carry out projects using a set of SNPs in order to, for example, search for genes responsible for a disease. Thus, prior to project initiation, geneticists need to select a subset of SNPs from large databases. Hubley et al. [44] formulated this task as a bi-objective optimization problem and proposed an algorithm called Multiobjective Analyzer for Genetic Marker Acquisition (MAGMA). The desired goals of a mapping project are to maximize the probability of locating a disease gene while minimizing the total project cost. However, as the authors point out, these goals may be subjective, difficult to describe, or may require excessive computation. To overcome this problem, the authors make use of the so called proxy objectives. That is to say, objectives that only capture certain aspects of the actual objectives [40]. In this case, the first proxy objective is to search for evenly spaced high-quality SNPs with an average spacing. Thus, the objectives are: minimize the average deviation from the ideal gap length between two SNPs and maximize the average quality of the SNPs. A solution in this problem is represented by a binary string where a bit is set to 1 only if the corresponding SNP is in the solution. The proposed algorithm was tested using two real SNP selection problems with a relatively small library of SNPs, and a constructed problem with a large library containing a vast number of SNPs. The Pareto front in all cases had a concave shape, and MAGMA was able to discover the true Pareto front in the three problems.

In a later study, Hubley et al. [43] proposed two new proxy objectives that reflect more precisely the actual goals of a project. Here the cost is modeled in a straightforward manner, as the sum of the cost associated with each SNP. The probability of project success is treated as the quality of a SNP, which is a heuristic combination of (i) allele frequency, (ii) database reliability, and (iii) biochemical suitability.

Lee et al. [55] formulate the probe design for DNA microarrays as a multiobjective problem, which is then solved by the NSGA-II [31]. The resulting problem has four objectives and one constraint. The authors use a thermodynamic criteria to assist the decision maker to choose a solution from the generated Pareto front. The melting temperature can be used to determine if a candidate probe hybridizes to the wrong target gene or not. This way, one can choose the set of probes which have the least mis-hybridizing probes from the

obtained Pareto front. Based on the specificity of hybridization of each probe, the proposed method achieved more reliable probe sets than those pre-existing oligonucleotide microarray for HPV (human papilloma virus) detection.

4.4.2 Classification

Deb and Reddy [32] address the classification of two-class cancer data using the NSGA-II [31]. Here, the authors formulate a two-objective and a three-objective problem. The first problem consists of minimizing the size of the gene-subset and minimizing the sum of misclassifications in the training and test samples. In the second problem, the misclassifications in the training and in test samples are considered as two different objectives. As some solutions with desirable subset sizes do not belong to the Pareto front, when using the standard Pareto domination concept, the authors introduce a variant called biased dominance. This modified concept allows that multiple solutions lying in parallel to a f_i axis do not dominate each other. An interesting finding is that in this problem a vector in the objective space can be produced by more than one solution in the decision variable space. The authors modified the NSGA-II so that it could take into account these types of solutions.

Usually, microarray data contain a large number of features (genes) from which most of them are non essential to carry out data classification. Banerjee et al. [1] proposed a MOEA that employs rough sets to reduce the number of features in order to ease the classification of gene expression patterns in a microarray. The set of genes is modeled as a rough set in such a way that the essential features are represented by the reduct of the information system. Thus, the objectives of this feature selection problem are: (i) to obtain a reduct of small cardinality and simultaneously (ii) to still classify all the elements of the universe with the same accuracy as with the entire attribute set. The feature reduction is carried out in a two stage process. The first stage generates an initial crude redundancy reduction among features by normalizing the expression values (attributes) and eliminating constantly expressed genes and ambiguously expressed genes (i.e., those with average expression value). In the second stage, the crude reduced data set is optimized by the NSGA-II [31] to achieve a refined minimal feature set. The formulation of this multi-objective problem includes two objectives: minimization of the number of attributes in the reduct set and maximization of the capacity to distinguish objects in order to achieve an acceptable classification. The only variable considered is the reduct which is represented by a binary string of length m (where m is the number of attributes). In this string, 1 indicates that the corresponding attribute is present in the reduct while 0 indicates the contrary. The proposed method was validated using microarray data sets consisting of three different cancer samples, namely colon cancer, lymphoma and leukemia. The Pareto front obtained in the three data sets is a typical convex front where the reduct cardinality decreases as the number of misclassifications increases. The proposed MOEA was compared against other approaches which include

a probabilistic neural network, a *t*-test-based feature selection with a fuzzy neural network, a saliency analysis to support vector machines and a linear aggregating function approach. Considering the available results, the MOEA achieved a better correct classification percentage than the other approaches using the three datasets.

Liu et al. [57] have proposed an entropy-based method to select genes related to the different cancer classes, simultaneously reducing the redundancy among the genes. This bi-objective problem is solved using an aggregation approach solved by a greedy algorithm.

Bleuler et al. [6] proposed an evolutionary framework for bi-clustering of gene expression data in a single-objective context. The main idea of the framework is to explore the search space by an EA and refine the solutions found by using a local search bi-clustering method. The framework was implemented using the bi-clustering method proposed by Cheng and Church [14]. The results showed that the EA coupled with a local search performs significantly better than the Cheng and Church's bi-clustering algorithm alone.

Recently, Mitra and co-workers [2, 64, 65] proposed a similar framework to that of Bleuler et al. [6] but in a multi-objective context. The two objectives considered were the maximization of the bi-cluster size and the maximization of the homogeneity. According to the results, this framework achieves better results than some other methods available in the literature [6, 14, 99, 101].

Prelic et al. [78] carried out a systematic comparison of five salient bi-clustering methods based on greedy search techniques, namely: the algorithm of Cheng and Church [14], Samba [92], the Order Preserving Submatrix Algorithm [4], the Iterative Signature Algorithm [45] and *xMotif*, [67]. Madeira and Oliveira [59] provides a survey on bi-clustering methods that, besides greedy search techniques, includes clustering methods based on strategies such as divide-and-conquer, and exhaustive enumeration to mention a few. The authors adopted only external indices to assess the performance of the algorithms. External indices are based on additional data in order to validate the achieved results. Moreover, the comparison study considered both synthetic and real datasets. The former has the advantage that the true optimal solutions are known *a priori*. As a reference algorithm, the authors proposed a fast and exact algorithm that uses a simple data model yet reflecting the fundamental idea of bi-clustering.

4.4.3 Sequence and Structure alignment

Malard et al. [60] formulate the *de novo* peptide identification as a constrained multi-objective optimization problem. The objectives considered in the study are the maximization of the similarity between portions of two peptides, and the maximization of the likelihood ratio between the null hypothesis and the alternative hypothesis. The former is that spectral peaks match ion fragments only by chance, whereas the latter is that spectral peaks match ion fragments because the candidate solution is in the sample. Constraints are

treated as an objective function in a similar way as the Constrained Multi-objective Optimization by Genetic Algorithm (COMOGA) proposed by Surry and Radcliffe [87]. The algorithm was implemented using the island parallel model [22, 68, 95], in which some subpopulations evolve independently of each other, although individuals periodically migrate between neighboring islands.

Boisson et al. [8] also studied the protein sequencing using the *de novo* peptide sequencing approach, although using a single-objective genetic algorithm. As the evaluations of the objective function involved in the problem are time consuming, Boisson et al. [9] decided to use a parallel genetic algorithm to discover the sequence of an experimental protein. The algorithm was implemented on a grid of computers.

Calonder et al. [12] address the problem of identifying gene modules on the basis of different types of biological data such as gene expression (GE), protein-protein interactions (PPI) and metabolic pathways (MP). Module identification refers to the identification of groups of genes similar with respect to its function or regulation mechanism. The particular problem addressed in this work is to identify the best module containing some user defined query genes with respect to n biological data sets. Some single-objective approaches for the identification of modules have been proposed including a co-clustering approach where a combined distance function is used as the objective function. Another approach combines distances on the Gene Ontology graph with gene expression data and applies a memetic algorithm[3] for identifying high scoring clusters.

The proposed multi-objective approach has some advantages over a single-objective aggregation approach. First, it is not required to define an overall similarity measure, which is often difficult since we need to aggregate measures (i.e., objective function values) with different scales and interpretations. With a multi-objective approach each similarity measure can be treated as an independent objective. Also, it offers a way to study the interactions and conflicts between the data sets. That is, the visual inspection of the trade-offs in the Pareto front allow us to determine, for instance, if accepting a slightly worse similarity on one data type could increase the similarity on the other data types substantially. Finally, as the objectives are treated independently, it is possible to easily integrate arbitrary data types and similarity measures.

In this formulation, each data type is associated with a distinct objective which is defined as the mean distance from all genes to the query genes on the corresponding data set. For each objective, a suitable measure of distance is computed. Each solution (module) is represented by a binary string of length m (where m is the number of genes) where a value of 1 indicates that the corresponding gene is included in the module. The MOEA employed in this work is the indicator-based evolutionary algorithm (IBEA) [102].

[3] Pablo Moscato [66] introduced the concept of "memetic algorithm" to denote the use of local search heuristics with a population-based strategy.

In the experimental study, the authors considered three bi-objective problems using different data types: GE-GE data on Arabidopsis, GE-MP data on Arabidopsis, and a yeast GE-PPI data set. In the experimental study, a local search heuristic was added to the evolutionary algorithm. However, the results revealed that the local search imposed a noticeable bias toward one of the objectives. The performance of the algorithm was compared against that of a single-objective aggregation approach and that of a k-means algorithm. In the first case, to generate the Pareto front, the single-objective optimizer was run repeatedly with 21 different weight vectors. The comparison of the resulting Pareto fronts using the ε-indicator [105] revealed that the multi-objective approach achieved approximation sets better than those obtained by the single-objective approach. In order to compare the multi-objective approach with the k-means algorithm, the authors ran the k-means using only the GE data, and then they selected at random a query gene from one of the clusters. For this cluster, they calculated the value of the two objectives, GE and PPI, to get a Pareto front consisting of a single solution. The same query gene was used as input to the EA to get the approximation set. Again, the ε-indicator showed that the EA performs better than k-means.

Zwir et al. [106] presented a two-level methodology for the elicitation and summarization of qualitative features in DNA sequences of *Tripanosoma cruzi*. The first stage had the goal of recognizing instances of interesting features through a multi-objective genetic-based clustering method. Here, the clustering problem was formulated as a multi-objective problem that takes into account, independently, the multiple measures of cluster quality. At this stage, Pareto local dominance was adopted. That is, a solution is locally non-dominated if there does not exist a neighboring solution that dominates it. At the second stage, the Pareto front obtained in the first stage was summarized in order to obtain a compact description of the set of interesting features.

4.4.4 Structure Prediction and Design

Lee and co-workers [56, 84] used the controlled elitist NSGA-II [30] to generate a set of DNA sequences which can be used in microarray design or in DNA-based computing. The desirable properties of a DNA sequence are the quality measures achieved while satisfying certain constraints. The quality of a sequence can be achieved by minimizing four objectives: the similarity between two sequences in the set, the number of bases that can be hybridized between sequences in the set, the degree of successive occurrences of the same base and the probability to form a secondary structure. A good sequence should have similar physical and chemical properties. To guarantee these characteristics the authors use as constraints the number of bases 'G' and 'C' in the sequence and the melting temperature where more than half of the double strands start to break into single strands. These constraints are handled with a tournament selection that determine the winner using the following rules: a feasible solution is preferred over an infeasible solution; between two infeasible solutions

the solution with the smaller constraint violation is preferred; between two feasible solutions the solution that dominates the other is preferred. The proposed approach was compared against three similar algorithms [27, 36, 91] using an instance problem of a set of 7 DNA sequences of length 20. The comparison was based on the average values of each objective over the generated set of DNA sequences. The results showed that the proposed method achieved smaller average values in all the objectives than the other approaches considered.

Day et al. [26] employed the multi-objective fast messy genetic algorithm (MO fmGA) [107] to solve the protein structure prediction problem. This study is based on an energy minimization technique which uses the CHARMm energy function. This function is composed of 10 major terms and in order to utilize a multi-objective framework, it was decomposed in two minimization objectives: (i) the sum of the connected and (ii) the sum of the non-connected atom energies. The decision variables for this problem are the dihedral angles for the protein being solved. The algorithm was applied to two proteins, [Met]-Enkephelin and Polyalanine. For both problems, a convex Pareto front was obtained. The results were compared against those obtained in a previous study using a single-objective fmGA (SO fmGA) [62]. To do so, for each vector of the obtained Pareto front the two objective values were added to obtain a single value. Then, the best objective value found was compared with the single value achieved by the SO fmGA. For [Met]-Enkephelin, the MO fmGA found the best solution, while for Polyalanine, the MO-fmGA compared favorably with respect to the SO fmGA.

Chen et al. [13] proposed a method to solve the structure alignment problem for homologous proteins. This problem can be formulated as a multi-objective optimization problem where the objectives are: maximize the number of aligned atoms and minimize their distance. The proposed method relies on a bipartite matching algorithm whose convergence is numerically stable and is also theoretically ensured.

4.4.5 Inverse problems

Phylogenetic inference is the construction of trees that represent the genealogical relationships between different species. Contrary to other kinds of taxonomy, phylogenetic classification is based on common ancestry and not mere similarity of appearance or function [75]. The reconstruction of phylogenetic trees relies on various types of data sets, for example nucleotide and amino acid sequences, protein shapes, anatomical characters, or behavioral traits to name a few.

Poladian and Jermiin [76] proposed using a multi-objective evolutionary approach to infer phylogenetic trees integrating many types of available data. As pointed out by the authors, MOEAs are especially suitable to obtain phylogenetic inferences for three reasons: (i) the large combinatorial space associated

with all possible phylogenies, (ii) the conflicting results obtained by using different data sets and (iii) the fact that, a single best tree may not tell the whole story but a nearly-best trees may also reveal information about the relationship between two species.

One of the current problems in phylogenetic inference is how to assess, combine, modify or reject different types of data. *Total evidence* [75] is one of the two main lines of thought about how to integrate information from different data types, which advocates the use of all available data to infer a phylogenetic tree. Instead of combining all available information, a multi-objective approach allows us to manage each type of information as a different objective. Thus, the multi-objective approach of Poladian and Jermiin [76] yields a family of trees instead of the single tree obtained by a combined analysis. The authors employed a basic MOEA where each solution encodes the type of topology of a candidate tree and the length (inferred evolutionary distance between species) for each edge of the tree. In this formulation each objective of the problem corresponds to the maximization of the likelihood of the tree given a type of information. The method was applied to a simple four-species problem using two data sets. The authors concluded that the visual inspection of the resulting Pareto front will help the experienced biological practitioner to interpret the conflict between the data sets and decide a plan of action. Furthermore, with a multi-objective approach the practitioner does not need to determine *a priori* the relative importance of the data.

The inference of gene regulatory networks is other type of inverse problems. Some gene products determine where, when and how much another gene is expressed into proteins. Thus cellular processes like cell growth, differentiation and reproduction are a result of complex interactions between genes instead of an isolated reaction of few genes. Gene regulatory networks are used to represent these interactions between genes using a directed graph. The task of the bioinformatician is to model such networks from large amounts of microarray data.

Spieth et al. [85] address the problem of finding gene regulatory networks using an evolutionary algorithm combined with a local search method. The global optimizer is a genetic algorithm whereas an evolution strategy plays the role of the local optimizer. The performance assessment showed that the proposed memetic algorithm is superior to standard optimization approaches found in the literature.

Recently, Keedwell and Narayanan [49] combined a genetic algorithm with a neural network to elucidate gene regulatory networks. The genetic algorithm has the goal of evolving a population of genes, while the neural network is used to evaluate how well the expression of the set of genes affects the expression values of other genes.

4.5 Future Areas of Research

As we have seen, MOEAs have been applied to different problems in biology and bioinformatics. However, there are other possible paths for future research that may be worth exploring. For example:

- **Use of Hybrid Approaches:** The use of combinations of soft computing[4] techniques for solving multi-objective problems arising in biology may be an interest path of future research in the area. Currently, most applications of soft computing in areas such as bioinformatics, normally rely on the use of a single technique [65] (e.g., artificial neural networks for classification or evolutionary algorithms for optimization). However, the use of combinations of techniques may introduce greater benefits. For example, a MOEA can be used to evolve the topology of an artificial neural network which serves as a classifier, adopting accuracy and complexity as the optimization criteria.

- **Incorporation of User's Preferences:** Most MOEAs are commonly employed under the assumption that the entire Pareto optimal set is needed. However, in most practical applications, not all the solutions are required, since users normally identify regions of interest within the Pareto front [41]. There are several ways in which the user's preferences can be incorporated into a MOEA such that the search is narrowed to a certain portion of the Pareto front (see for example [19]). Although in recent years more MOEA researchers have become interested in this topic (see for example [10, 11, 16, 25, 79, 97]), it certainly requires much further work.

- **Use of Domain Knowledge:** The incorporation of domain knowledge may improve the performance of MOEAs adopted to solve complex problems. Such knowledge may be provided either *a priori* (when available) or can be extracted during the search [48, 54]. This knowledge may influence the operators of a MOEA or can be used to design heuristic procedures aimed to reduce the size of the search space.

4.6 Conclusions

In this chapter, we have explored the use of MOEAs in different biological and bioinformatics applications. First, the most popular MOEAs in current use were briefly described. Then, a simple taxonomy of applications was introduced and representative applications within each class were described. It is worth noting, however, that this review was presented from the perspective

[4] Soft computing refers to a collection of computational techniques in computer science which attempt to study, model, and analyze complex phenomena. Such techniques include evolutionary algorithms, neural networks and fuzzy logic [93].

of a computer scientist and not from a biologist's point of view. We hope, however, that biologists may find it useful in spite of its possible pitfalls.

Readers will also note that no attempt was made to be critical in the review, since the aim was to provide a wide view of the field rather than to introduce any potential bias in the current work being done in this area. Clearly, the interest from biologists for using MOEAs is increasing and we certainly hope that such trend is maintained in the years to come, since such has been the main goal of this chapter.

Acknowledgements

The authors thank the anonymous reviewers for their valuable comments, which greatly helped us to improve the contents of this chapter.

The first author acknowledges support from CONACyT through a scholarship to pursue graduate studies at the Computer Science Department from CINVESTAV-IPN. The second author acknowledges support from CONACyT project no. 45683-Y.

References

1. Banerjee M, Mitra S, Banka H (2007) Evolutionary Rough Feature Selection in Gene Expression Data. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 37:622–632
2. Banka H, Mitra S (2006) Evolutionary biclustering of gene expressions. *Ubiquity*, 7(42):1–12
3. Barbosa HJ, Barreto AM (2001) An interactive genetic algorithm with coevolution of weights for multiobjective problems. In: Spector L, Goodman ED, Wu A, Langdon W, Voigt HM, Gen M, Sen S, Dorigo M, Pezeshk S, Garzon MH, Burke E (eds.) *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2001)*, 203–210. Morgan Kaufmann Publishers, San Francisco, California
4. Ben-Dor A, Chor B, Karp R, Yakhini Z (2003) Discovering Local Structure in Gene Expression Data: The Order-Preserving Submatrix Problem. *Journal of Computational Biology*, 10(3-4):373–384
5. Beume N, Naujoks B, Emmerich M (2007) SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, 181(3):1653–1669
6. Bleuler S, Prelić A, Zitzler E (2004) An EA Framework for Biclustering of Gene Expression Data. In: *Congress on Evolutionary Computation (CEC 2004)*, 166–173. IEEE, Piscataway, NJ
7. Blum C, Roli A (2003) Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys*, 35(3):268–308
8. Boisson J, Jourdan L, Talbi E, Rolando C (2006) A Preliminary Work on Evolutionary Identification of Protein Variants and New Proteins on Grids. *20th International Conference on Advanced Information Networking and Applications - Volume 2 (AINA'06)*, 02:583–587

9. Boisson J, Jourdan L, Talbi E, Rolando C (2006) Protein Sequencing with an Adaptive Genetic Algorithm from Tandem Mass Spectrometry. In: *IEEE Congress on Evolutionary Computation (CEC), Vancouver, BC, Canada*, 1412–1419
10. Branke J, Deb K (2005) Integrating User Preferences into Evolutionary Multi-Objective Optimization. In: Jin Y (ed.) *Knowledge Incorporation in Evolutionary Computation*, 461–477. Springer, Berlin Heidelberg. ISBN 3-540-22902-7
11. Branke J, Kaußler T, Schmeck H (2001) Guidance in Evolutionary Multi-Objective Optimization. *Advances in Engineering Software*, 32:499–507
12. Calonder M, Bleuler S, Zitzler E (2006) Module Identification from Heterogeneous Biological Data Using Multiobjective Evolutionary Algorithms. In: *Parallel Problem Solving from Nature (PPSN IX)*, no. 4193 in LNCS, 573–582. Springer
13. Chen L, Wu L, Wang R, Wang Y, Zhang S, Zhang X (2005) Comparison of Protein Structures by Multi-Objective Optimization. *Genome Informatics*, 16(2):114–124
14. Cheng Y, Church G (2000) Biclustering of expression data. *Proc Int Conf Intell Syst Mol Biol*, 8:93–103
15. Coello Coello CA (1996) *An Empirical Study of Evolutionary Techniques for Multiobjective Optimization in Engineering Design*. Ph.D. thesis, Department of Computer Science, Tulane University, New Orleans, LA
16. Coello Coello CA (2000) Handling Preferences in Evolutionary Multiobjective Optimization: A Survey. In: *2000 Congress on Evolutionary Computation*, vol. 1, 30–37. IEEE Service Center, Piscataway, New Jersey
17. Coello Coello CA (2000) An Updated Survey of GA-Based Multiobjective Optimization Techniques. *ACM Computing Surveys*, 32(2):109–143
18. Coello Coello CA, Lamont GB (eds.) (2004) *Applications of Multi-Objective Evolutionary Algorithms*. World Scientific, Singapore. ISBN 981-256-106-4
19. Coello Coello CA, Lamont GB, Van Veldhuizen DA (2007) *Evolutionary Algorithms for Solving Multi-Objective Problems*. Kluwer Academic Publishers, New York, second edn. ISBN 978-0-387-33254-3
20. Coello Coello CA, Reyes Sierra M (2003) A Coevolutionary Multi-Objective Evolutionary Algorithm. In: *Proceedings of the 2003 Congress on Evolutionary Computation (CEC'2003)*, vol. 1, 482–489. IEEE Press, Canberra, Australia
21. Coello Coello CA, Toscano Pulido G (2001) Multiobjective Optimization using a Micro-Genetic Algorithm. In: Spector L, Goodman ED, Wu A, Langdon W, Voigt HM, Gen M, Sen S, Dorigo M, Pezeshk S, Garzon MH, Burke E (eds.) *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2001)*, 274–282. Morgan Kaufmann Publishers, San Francisco, California
22. Coello Coello CA, Van Veldhuizen DA, Lamont GB (2002) *Evolutionary Algorithms for Solving Multi-Objective Problems*. Kluwer Academic Publishers, New York. ISBN 0-3064-6762-3
23. Corne DW, Jerram NR, Knowles JD, Oates MJ (2001) PESA-II: Region-based Selection in Evolutionary Multiobjective Optimization. In: Spector L, Goodman ED, Wu A, Langdon W, Voigt HM, Gen M, Sen S, Dorigo M, Pezeshk S, Garzon MH, Burke E (eds.) *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2001)*, 283–290. Morgan Kaufmann Publishers, San Francisco, California

24. Corne DW, Knowles JD, Oates MJ (2000) The Pareto Envelope-based Selection Algorithm for Multiobjective Optimization. In: Schoenauer M, Deb K, Rudolph G, Yao X, Lutton E, Merelo JJ, Schwefel HP (eds.) *Proceedings of the Parallel Problem Solving from Nature VI Conference*, 839–848. Springer. Lecture Notes in Computer Science No. 1917, Paris, France
25. Cvetković D, Parmee IC (2002) Preferences and their Application in Evolutionary Multiobjective Optimisation. *IEEE Transactions on Evolutionary Computation*, 6(1):42–57
26. Day RO, Zydallis JB, Lamont GB (2002) Solving the Protein structure Prediction Problem through a Multi-Objective Genetic Algorithm. In: *Proceedings of IEEE/DARPA International Conference on Computational Nanoscience (ICCN'02)*, 32–35
27. Deaton R, Chen J, Bi H, Rose J (2002) A software tool for generating noncrosshybridizing libraries of DNA oligonucleotides. *Preliminary Proceedings of the 8th International Meeting on DNA Based Computers, June*, 10–13
28. Deb K (2001) *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley & Sons, Chichester, UK. ISBN 0-471-87339-X
29. Deb K, Agrawal S, Pratab A, Meyarivan T (2000) A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. In: Schoenauer M, Deb K, Rudolph G, Yao X, Lutton E, Merelo JJ, Schwefel HP (eds.) *Proceedings of the Parallel Problem Solving from Nature VI Conference*, 849–858. Springer. Lecture Notes in Computer Science No. 1917, Paris, France
30. Deb K, Goel T (2001) Controlled elitist non-dominated sorting genetic algorithms for better convergence. *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization*, 67–81
31. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A Fast and Elitist Multiobjective Genetic Algorithm: NSGA–II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197
32. Deb K, Reddy A (2003) Reliable classification of two-class cancer data using evolutionary algorithms. *BioSystems*, 72(1):111–129
33. Edgeworth FY (1881) *Mathematical Psychics*. P. Keagan, London, England
34. Ehrgott M (2005) *Multicriteria Optimization*. Springer, Berlin, second edn. ISBN 3-540-21398-8
35. Ehrlich P, Raven P (1964) Butterflies and Plants: A Study in Coevolution. *Evolution*, 18:586–608
36. Faulhammer D, Cukras AR, Lipton RJ, Landweber LF (2000) Molecular computation: RNA solutions to chess problems. *Annals of the New York Academy of Sciences*, 97(4):1385–1389
37. Fonseca CM, Fleming PJ (1993) Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In: Forrest S (ed.) *Proceedings of the Fifth International Conference on Genetic Algorithms*, 416–423. University of Illinois at Urbana-Champaign, Morgan Kaufmann Publishers, San Mateo, California
38. Fourman MP (1985) Compaction of Symbolic Layout using Genetic Algorithms. In: Grefenstette JJ (ed.) *Genetic Algorithms and their Applications: Proceedings of the First International Conference on Genetic Algorithms*, 141–153. Lawrence Erlbaum, Hillsdale, New Jersey
39. Goldberg DE (1989) *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Publishing Company, Reading, Massachusetts

40. Handl J, Kell DB, Knowles J (2007) Multiobjective optimization in bioinformatics and computational biology. *IEEE-ACM Transactions on Computational Biology and Bioinformatics*, 4(2):279–292
41. Handl J, Knowles J (2007) An Evolutionary Approach to Multiobjective Clustering. *IEEE Transactions on Evolutionary Computation*, 11(1):56–76
42. Horn J, Nafpliotis N, Goldberg DE (1994) A Niched Pareto Genetic Algorithm for Multiobjective Optimization. In: *Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, vol. 1, 82–87. IEEE Service Center, Piscataway, New Jersey
43. Hubley R, Zitzler E, Roach J (2003) Evolutionary algorithms for the selection of single nucleotide polymorphisms. *BMC Bioinformatics*, 4(30)
44. Hubley R, Zitzler E, Siegel A, Roach J (2002) Multiobjective Genetic Marker Selection. In: *Advances in Nature-Inspired Computation: The PPSN VII Workshops*, 32–33. University of Reading, UK
45. Ihmels J, Friedlander G, Bergmann S, Sarig O, Ziv Y, Barkai N (2002) Revealing modular organization in the yeast transcriptional network. *Nat Genet*, 31(4):370–7
46. Iorio AW, Li X (2004) A Cooperative Coevolutionary Multiobjective Algorithm Using Non-dominated Sorting. In: et al KD (ed.) *Genetic and Evolutionary Computation–GECCO 2004. Proceedings of the Genetic and Evolutionary Computation Conference. Part I*, 537–548. Springer-Verlag, Lecture Notes in Computer Science Vol. 3102, Seattle, Washington, USA
47. Ito K, Akagi S, Nishikawa M (1983) A Multiobjective Optimization Approach to a Design Problem of Heat Insulation for Thermal Distribution Piping Network Systems. *Journal of Mechanisms, Transmissions and Automation in Design (Transactions of the ASME)*, 105:206–213
48. Jin Y (ed.) (2005) *Knowledge Incorporation in Evolutionary Computation*. Springer, Berlin. ISBN 3-540-22902-7
49. Keedwell E, Narayanan A (2005) Discovering gene networks with a neural-genetic hybrid. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2(3):231–242
50. Keerativuttiumrong N, Chaiyaratana N, Varavithya V (2002) Multi-objective Co-operative Co-evolutionary Genetic Algorithm. In: Merelo Guervós JJ, Adamidis P, Beyer HG, nas JLFV, Schwefel HP (eds.) *Parallel Problem Solving from Nature—PPSN VII*, 288–297. Springer-Verlag. Lecture Notes in Computer Science No. 2439, Granada, Spain
51. Kleeman MP, Lamont GB (2006) Coevolutionary Multi-Objective EAs: The Next Frontier? In: *2006 IEEE Congress on Evolutionary Computation (CEC'2006)*, 6190–6199. IEEE, Vancouver, BC, Canada
52. Knowles JD, Corne DW (1999) The Pareto Archived Evolution Strategy: A New Baseline Algorithm for Multiobjective Optimisation. In: *1999 Congress on Evolutionary Computation*, 98–105. IEEE Service Center, Washington, D.C.
53. Knowles JD, Corne DW (2000) Approximating the Nondominated Front Using the Pareto Archived Evolution Strategy. *Evolutionary Computation*, 8(2):149–172
54. Landa Becerra R, Coello Coello CA (2006) Solving Hard Multiobjective Optimization Problems Using ε-Constraint with Cultured Differential Evolution. In: Runarsson TP, Beyer HG, Burke E, Merelo-Guervós JJ, Whitley LD, Yao X (eds.) *Parallel Problem Solving from Nature - PPSN IX, 9th International*

Conference, 543–552. Springer. Lecture Notes in Computer Science Vol. 4193, Reykjavik, Iceland
55. Lee I, Kim S, Zhang B (2004) Multi-objective Evolutionary Probe Design Based on Thermodynamic Criteria for HPV Detection. *Lecture Notes in Computer Science*, 3157:742–750
56. Lee I, Shin S, Zhang B (2003) DNA sequence optimization using constrained multi-objective evolutionary algorithm. *Evolutionary Computation, 2003. CEC'03. The 2003 Congress on*, 4
57. Liu X, Krishnan A, Mondry A (2005) An Entropy-based gene selection method for cancer classification using microarray data. *BMC Bioinformatics*, 6:76
58. Lohn JD, Kraus WF, Haith GL (2002) Comparing a Coevolutionary Genetic Algorithm for Multiobjective Optimization. In: *Congress on Evolutionary Computation (CEC'2002)*, vol. 2, 1157–1162. IEEE Service Center, Piscataway, New Jersey
59. Madeira S, Oliveira A (2004) Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1(1):24–45
60. Malard J, Heredia-Langner A, Baxter D, Jarman K, Cannon W (2004) Constrained de novo peptide identification via multi-objective optimization. *Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International*
61. Mao J, Hirasawa K, Hu J, Murata J (2001) Genetic Symbiosis Algorithm for Multiobjective Optimization Problems. In: *Proceedings of the 2001 Genetic and Evolutionary Computation Conference. Late-Breaking Papers*, 267–274. San Francisco, California
62. Michaud SR, Zydallis JB, Lamont GB, Pachter R (2001) Scaling a genetic algorithm to medium-sized peptides by detecting secondary structures with an analysis of building blocks. In: *Proceedings of the First International Conference on Computational Nanoscience*, 29–32
63. Miettinen KM (1999) *Nonlinear Multiobjective Optimization.* Kluwer Academic Publishers, Boston, Massachusetts
64. Mitra S, Banka H (2006) Multi-objective evolutionary biclustering of gene expression data. *Pattern Recognition*, 39(12):2464–2477
65. Mitra S, Banka H, Pal S (2006) A MOE framework for Biclustering of Microarray Data. *Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06)-Volume 01*, 1154–1157
66. Moscato P (1989) On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts. Towards Memetic Algorithms. Tech. Rep. 158–79, Caltech Concurrent Computation Program, California Institute of Technology, Pasadena, California
67. Murali T, Kasif S (2003) Extracting conserved gene expression motifs from gene expression data. *Proc. Pacific Symp. Biocomputing*, 8:77–88
68. Nedjah N, Alba E, de Macedo Mourelle L (2006) *Parallel Evolutionary Computations.* Springer-Verlag. ISBN 3-540-32837-8
69. Osyczka A (1985) Multicriteria optimization for engineering design. In: Gero JS (ed.) *Design Optimization*, 193–227. Academic Press
70. Paredis J (1995) Coevolutionary computation. *Artificial Life*, 2(4):355–375
71. Paredis J (1998) Coevolutionary algorithms. In: Bäck T, Fogel DB, Michalewicz Z (eds.) *The Handbook of Evolutionary Computation, 1st Supplement*, 225–238. Institute of Physics Publishing and Oxford University Press

72. Pareto V (1896) *Cours D'Economie Politique*, vol. I and II. F. Rouge, Lausanne
73. Parmee IC, Watson AH (1999) Preliminary Airframe Design Using Co-Evolutionary Multiobjective Genetic Algorithms. In: Banzhaf W, Daida J, Eiben AE, Garzon MH, Honavar V, Jakiela M, Smith RE (eds.) *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'99)*, vol. 2, 1657–1665. Morgan Kaufmann, San Francisco, California
74. Pimpawat C, Chaiyaratana N (2001) Using a co-operative co-evolutionary genetic algorithm to solve a three-dimensional container loading problem. In: *Proceedings of the Congress on Evolutionary Computation 2001 (CEC'2001)*, vol. 2, 1197–1204. IEEE Service Center, Piscataway, New Jersey
75. Poladian L, Jermiin L (2004) What might evolutionary algorithms (EA) and multi-objective optimization (MOO) contribute to phylogenetics and the total evidence debate? In: et al RP (ed.) *GECCO 2004 Workshop Proceedings*. Seattle, Washington, USA
76. Poladian L, Jermiin L (2006) Multi-objective evolutionary algorithms and phylogenetic inference with multiple data sets. *Soft Computing-A Fusion of Foundations, Methodologies and Applications*, 10(4):359–368
77. Potter MA, de Jong K (1994) A Cooperative Coevolutionary Approach to Function Optimization. In: Davidor Y, Schwefel HP, Männer R (eds.) *Parallel Problem Solving from Nature—PPSN III*, 249–257. Springer-Verlag. Lecture Notes in Computer Science Vol. 866, Jerusalem, Israel
78. Prelić A, Bleuler S, Zimmermann P, Wille A, Bühlmann P, Gruissem W, Hennig L, Thiele L, Zitzler E (2006) A Systematic Comparison and Evaluation of Biclustering Methods for Gene Expression Data. *Bioinformatics*, 22(9):1122–1129
79. Rachmawati L, Srinivasan D (2006) Preference Incorporation in Multi-objective Evolutionary Algorithms: A Survey. In: *2006 IEEE Congress on Evolutionary Computation (CEC'2006)*, 3385–3391. IEEE, Vancouver, BC, Canada
80. Rosenberg RS (1967) *Simulation of genetic populations with biochemical properties*. Ph.D. thesis, University of Michigan, Ann Arbor, Michigan, USA
81. Rosin C, Belew R (1996) New methods for competitive coevolution. *Evolutionary Computation*, 5(1):1–29
82. Schaffer JD (1984) *Multiple Objective Optimization with Vector Evaluated Genetic Algorithms*. Ph.D. thesis, Vanderbilt University, Nashville, Tennessee
83. Schaffer JD (1985) Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. In: *Genetic Algorithms and their Applications: Proceedings of the First International Conference on Genetic Algorithms*, 93–100. Lawrence Erlbaum, Hillsdale, New Jersey
84. Shin S, Lee I, Kim D, Zhang B (2005) Multiobjective Evolutionary Optimization of DNA Sequences for Reliable DNA Computing. *Evolutionary Computation, IEEE Transactions on*, 9(2):143–158
85. Spieth C, Streichert F, Supper J, Speer N, Zell A (2005) Feedback Memetic Algorithms for Modeling Gene Regulatory Networks. *Computational Intelligence in Bioinformatics and Computational Biology, 2005. CIBCB'05. Proceedings of the 2005 IEEE Symposium on*, 1–7
86. Srinivas N, Deb K (1994) Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation*, 2(3):221–248
87. Surry P, Radcliffe N (1997) The COMOGA Method: Constrained Optimisation by Multiobjective Genetic Algorithms. *Control and Cybernetics*, 26(3):391–412

88. Syswerda G, Palmucci J (1991) The Application of Genetic Algorithms to Resource Scheduling. In: Belew RK, Booker LB (eds.) *Proceedings of the Fourth International Conference on Genetic Algorithms*, 502–508. Morgan Kaufmann Publishers, San Mateo, California
89. Tan KC, Yang YJ, Goh CK (2006) A Distributed Cooperative Coevolutionary Algorithm for Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation*, 10(5):527–549
90. Tan KC, Yang YJ, Lee TH (2003) A Distributed Cooperative Coevolutionary Algorithm for Multiobjective Optimization. In: *Proceedings of the 2003 Congress on Evolutionary Computation (CEC'2003)*, vol. 4, 2513–2520. IEEE Press, Canberra, Australia
91. Tanaka F, Nakatsugawa M, Yamamoto M, Shiba T, Ohuchi A (2002) Towards a general-purpose sequence design system in DNA computing. *Evolutionary Computation, 2002. CEC'02. Proceedings of the 2002 Congress on*, 73–78
92. Tanay A, Sharan R, Shamir R (2002) Discovering statistically significant biclusters in gene expression data. *Bioinformatics*, 18(Suppl 1):S136–S144
93. Tettamanzi A, Tomassini M (2001) *Soft Computing: Integrating Evolutionary, Neural and Fuzzy Systems*. Springer-Verlag. ISBN 978-3540422044
94. Toscano Pulido G, Coello Coello CA (2003) The Micro Genetic Algorithm 2: Towards Online Adaptation in Evolutionary Multiobjective Optimization. In: Fonseca CM, Fleming PJ, Zitzler E, Deb K, Thiele L (eds.) *Evolutionary Multi-Criterion Optimization. Second International Conference, EMO 2003*, 252–266. Springer. Lecture Notes in Computer Science. Volume 2632, Faro, Portugal
95. Veldhuizen DAV, Zydallis JB, Lamont GB (2003) Considerations in engineering parallel multiobjective evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 7(3):144–173
96. Villalobos-Arias M, Coello Coello CA, Hernández-Lerma O (2006) Asymptotic Convergence of Metaheuristics for Multiobjective Optimization Problems. *Soft Computing*, 10(11):1001–1005
97. Wang J, Terpenny JP (2005) Interactive Preference Incorporation in Evolutionary Engineering Design. In: Jin Y (ed.) *Knowledge Incorporation in Evolutionary Computation*, 525–543. Springer, Berlin Heidelberg. ISBN 3-540-22902-7
98. Wienke PB, Lucasius C, Kateman G (1992) Multicriteria target optimization of analytical procedures using a genetic algorithm. *Analytical Chimica Acta*, 265(2):211–225
99. Yang J, Wang H, Wang W, Yu P (2003) Enhanced biclustering on expression data. *Bioinformatics and Bioengineering, 2003. Proceedings. Third IEEE Symposium on*, 00:321–327
100. Zeng SY, Kang LS, Ding LX (2004) An Orthogonal Multi-objective Evolutionary Algorithm for Multi-objective Optimization Problems with Constraints. *Evolutionary Computation*, 12(1):77–98
101. Zhang Z, Teo A, Ooi B, Tan K (2004) Mining deterministic biclusters in gene expression data. *Bioinformatics and Bioengineering, 2004. BIBE 2004. Proceedings. Fourth IEEE Symposium on*, 283–290
102. Zitzler E, Künzli S (2004) Indicator-based Selection in Multiobjective Search. In: et al XY (ed.) *Parallel Problem Solving from Nature - PPSN VIII*, 832–842. Springer-Verlag. Lecture Notes in Computer Science Vol. 3242, Birmingham, UK

103. Zitzler E, Laumanns M, Thiele L (2001) SPEA2: Improving the Strength Pareto Evolutionary Algorithm. In: Giannakoglou K, Tsahalis D, Periaux J, Papailou P, Fogarty T (eds.) *EUROGEN 2001. Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems*, 95–100. Athens, Greece
104. Zitzler E, Thiele L (1999) Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. *IEEE Transactions on Evolutionary Computation*, 3(4):257–271
105. Zitzler E, Thiele L, Laumanns M, Fonseca CM, da Fonseca VG (2003) Performance Assessment of Multiobjective Optimizers: An Analysis and Review. *IEEE Transactions on Evolutionary Computation*, 7(2):117–132
106. Zwir I, Zaliz R, Ruspini E (2002) Automated Biological Sequence Description by Genetic Multiobjective Generalized Clustering. *Annals of the New York Academy of Sciences*, 980(1):65–82
107. Zydallis JB, Veldhuizen DAV, Lamont GB (2001) A Statistical Comparison of Multiobjective Evolutionary Algorithms Including the MOMGA–II. In: Zitzler E, Deb K, Thiele L, Coello CAC, Corne D (eds.) *First International Conference on Evolutionary Multi-Criterion Optimization*, 226–240. Springer-Verlag. Lecture Notes in Computer Science No. 1993