REVIEW ARTICLE

A survey of multi-objective metaheuristics applied to structural optimization

**Gustavo R. Zavala · Antonio J. Nebro ·
Francisco Luna · Carlos A. Coello Coello**

Abstract In civil and industrial engineering, structural design optimization problems are usually characterized by the presence of multiple conflicting objectives, as to get the minimum investment cost and the maximum safety of the final design. This issue makes these problems to have not only one single solution, but a set them. Such solutions represent the possible trade-offs among the different objectives to be optimized. This paper reviews the latest developments in the field of multi-objective metaheuristics for solving design problems focusing on the optimization of the topology, shape, and sizing of civil engineering structures. We review both the algorithms and the applications, and the most relevant features of the solvers and the design optimization problems are analyzed. The paper ends by addressing a number of relevant and open issues that can be the subject of further research.

Keywords Multi-objective optimization · Metaheuristics · Structural optimization · Survey

G. R. Zavala
Universidad Nacional del Nordeste, Corrientes, Argentina
e-mail: grzavala@gmail.com

A. J. Nebro
Departamento de Lenguajes y Ciencias de la Computación,
Universidad de Málaga, Málaga, Spain
e-mail: antonio@lcc.uma.es

F. Luna (✉)
Departamento de Informática, Universidad Carlos III de Madrid,
Leganés, Spain
e-mail: fluna@inf.uc3m.es

C. A. Coello Coello
CINVESTAV-IPN Departamento de Computación,
Mexico City, México
e-mail: ccoello@cs.cinvestav.mx

1 Introduction

Civil engineering is a discipline that deals with the design, construction, and maintenance of the physical and naturally built environment, which includes works such as bridges, roads, canals, dams, and buildings (Institution of Civil Engineers 2012). By determining the stability of structures, it can materialize, among others, works for the shelter, protection, a livability of the society and its goods against inclement weather and environmental phenomena.

In this field, as in other disciplines (e.g, industrial engineering, telecommunications, economics, etc.), optimization problems appear everywhere and they bring a constant challenge, not only because of the appearance of new materials that may be used along with traditional ones, but also because this leads to cutting-edge methodologies that enable to realize complex designs in unexpected places.

When considering civil structures there are many issues to take into consideration, but there is always a common denominator for the concretion of a project: getting the minimum investment cost and the maximum safety of the final design. These are opposite goals indeed, because enhancing one of them involves worsening the other one. An example, is the design of a bridge, in which it is normally desirable to minimize cost, but with the aim of simultaneously maximizing safety. These are two conflicting goals, since a higher safety implies a higher economical cost. As a consequence, multi-objective optimization techniques are a highly valuable tool in this context. Compared to single-objective optimization, where a single objective function $f(x)$ has to be optimized and a single optimal solution x^* is searched for, in multi-objective optimization a set of so-called non-dominated solutions, known as the Pareto optimal set, is the goal of the search (Coello et al. 2007; Deb 2001). Non-dominated solutions represent the best possible trade-offs

among the objectives of the problem (i.e., these are solutions for which no improvement can be obtained for one objective without worsening another). Such non-dominated solutions are provided to the decision maker (e.g., the designer of a bridge) so that he/she can select a single one that best fits his/her particular needs and requirements (i.e., his/her user preferences). Among the available computational methods that can be used to solve multi-objective optimization problems, we focus on metaheuristics (Blum and Roli 2003), which are high level strategies governing a set of underlying techniques (usually, heuristics) in the search of optimal or quasi-optimal solutions of a given optimization problem. Metaheuristics are considered as particularly useful algorithms in structural engineering because, being randomized black box algorithms, they are able to handle problems with non-linear, non-differentiable, or noisy objectives, which are features normally found in structural engineering. Also, as opposed to traditional mathematical programming techniques used for solving multiobjective optimization problems (Miettinen 1999), metaheuristics are able to generate several elements of the Pareto optimal set in a single run.

The main goal of this paper is to provide both regular users and newcomers with a comprehensive survey of multi-objective metaheuristic techniques applied to structural design, along with a review of applications. To the best of our knowledge, this is the first attempt in the literature devoted to analyzing these two topics together. One can find an extensive study of evolutionary computing approaches in Kicinger et al. (2005), but no non-evolutionary metaheuristics are not considered in that study and the attention paid to multi-objective optimizations is limited to a section whose length is only of one page. A survey of multi-objective optimization methods for engineering was presented in Marler and Arora (2004), where only genetic algorithms, a particular type of metaheuristic, were included. However, since 2005 (the publication date of such survey), the multi-objective research community has developed numerous and major algorithmic advances, many of which have been applied to structural design problems. Thus, we consider that it is worth reviewing such research work and such is the motivation for this work. This paper aims to provide an overall view of the work conducted on the use of multi-objective metaheuristics for solving structural optimization problems, thus facilitating the exploration of the literature for those interested in pursuing research in this area. It is worth mentioning that we provide not only a fairly comprehensive list of the different metaheuristics that have been used to address structural design problems, but also details about each particular problem in which they have been applied (namely, type and domain, number of decision variables, objectives, and constraints, number of

nodes, elements and groups) as well as details of the specific type of algorithm adopted (i.e., algorithm type, population size, termination condition, operators, type of local search (if adopted)).

The remainder of this paper is structured as follows. Section 2 introduces basic concepts about multi-objective optimization. Basic concepts about metaheuristics, their main features, and the implications of dealing with multi-objective optimization problems are issues dealt with in Section 3. The next section is devoted to presenting a classification of the different types of structural design problems. The survey of solved problems and used techniques is provided in Section 5, and an analysis is included in Section 6. Finally, we present our conclusions as well as some open research lines in this area in Section 7.

2 Multi-objective optimization background

In this section, we provide some background on multi-objective optimization fundamentals. We define first basic concepts such as *multi-objective optimization problem*, *Pareto optimality*, *Pareto dominance*, *Pareto optimal set*, and *Pareto front*. Then, the goals of multi-objective optimization are discussed. We will assume, without loss of generality, that all the objective functions are to be minimized.

A general multi-objective optimization problem (MOP) can be formally defined as follows:

Definition 1 (MOP) Find a vector $\mathbf{x}^* = \left[x_1^*, x_2^*, \ldots, x_n^*\right]$ which satisfies the m inequality constraints $g_i(\mathbf{x}) \geq 0, i = 1, 2, \ldots, m$, the p equality constraints $h_i(\mathbf{x}) = 0, i = 1, 2, \ldots, p$, and minimizes the vector function $\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x})]^T$, where $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$ is the vector of decision variables.

The set of all the values satisfying the constraints defines the *feasible region* Ω and any point $\mathbf{x} \in \Omega$ is a *feasible solution*. In order to determine those points that are solutions to a given MOP, the concept of *Pareto optimality* has to be introduced:

Definition 2 (Pareto Optimality) A point $\mathbf{x}^* \in \Omega$ is Pareto Optimal if for every $\mathbf{x} \in \Omega$ and $I = \{1, 2, \ldots, k\}$ either $\forall_{i \in I}(f_i(\mathbf{x}) = f_i(\mathbf{x}^*))$ or there is at least one $i \in I$ such that $f_i(\mathbf{x}) > f_i(\mathbf{x}^*)$.

This definition states that $\mathbf{x}^*$ is Pareto optimal if no feasible vector $\mathbf{x}$ exists which would improve some criteria without causing a simultaneous worsening in at least one

other criterion. A related definition associated with Pareto optimality is *Pareto dominance*:

Definition 3 (Pareto Dominance) A vector $\mathbf{u} = (u_1, \ldots, u_k)$ is said to dominate $\mathbf{v} = (v_1, \ldots, v_k)$ (denoted by $\mathbf{u} \preccurlyeq \mathbf{v}$) if and only if $\mathbf{u}$ is partially less than $\mathbf{v}$, i.e., $\forall i \in \{1, \ldots, k\}, \ u_i \leq v_i \ \wedge \ \exists i \in \{1, \ldots, k\} : \ u_i < v_i$.

This way, a Pareto optimal point is that which is not dominated by any other point in Ω. Pareto dominance relates two solutions and it can be used as a binary operator. Thus, the application of this operator to two solutions in objective space returns that one solution *dominates* another or that the solutions do not dominate each other (i.e., they are *non-dominated* solutions).

The optimization of a MOP consequently will be aimed at finding the set of all the Pareto optimal solutions. This is the so-called *Pareto optimal set*, or simply Pareto set, and it is defined as follows:

Definition 4 (Pareto Optimal Set) For a given MOP $\mathbf{f}(\mathbf{x})$, the Pareto optimal set is defined as $\mathcal{P}^* = \{\mathbf{x} \in \Omega | \neg \exists \mathbf{x}' \in \Omega, \mathbf{f}(\mathbf{x}') \preccurlyeq \mathbf{f}(\mathbf{x})\}$.

Each vector in the Pareto optimal set has a correspondence in objective function space, leading to the so-called *Pareto front*:

Definition 5 Pareto Front For a given MOP $\mathbf{f}(\mathbf{x})$ and its Pareto optimal set $\mathcal{P}^*$, the Pareto front is defined as $\mathcal{PF}^* = \{\mathbf{f}(\mathbf{x}) | \mathbf{x} \in \mathcal{P}^*\}$.

Let us take into consideration as an example the four-bar plane truss design problem depicted in Fig. 1. It is a bi-objective optimization problem, in which two objectives are to 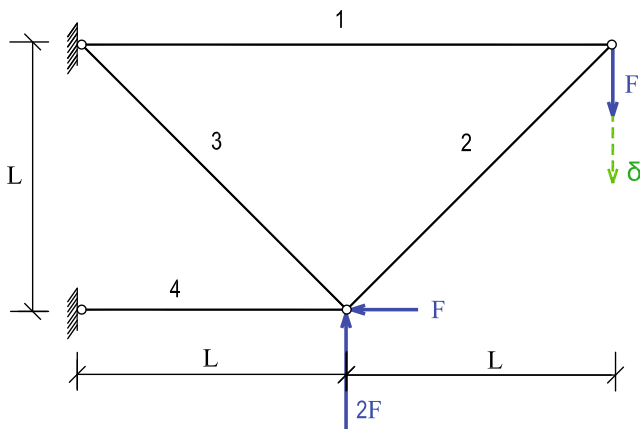be minimized: the volume of the truss (f_1) and its joint displacement Δ (f_2). This problem can be formulated as follows:

$$
\begin{aligned}
Min F &= (f_1(\mathbf{x}), f_2(\mathbf{x})) \\
f_1(\mathbf{x}) &= L(2x_1 + \sqrt{2}x_2 + \sqrt{x_3} + x_4) \\
f_2(\mathbf{x}) &= \frac{FL}{E}\left(\frac{2}{x_1} + \frac{2\sqrt{2}}{x_2} - \frac{2\sqrt{2}}{x_3}\frac{2}{x_4}\right)
\end{aligned}
$$

such that:

$$
\begin{aligned}
(F/\sigma) &\leq x_1, x_4 \leq 3(F/\sigma) \\
\sqrt{2}(F/\sigma) &\leq x_2, x_3 \leq 3(F/\sigma)
\end{aligned}
$$

where:

$$
\begin{aligned}
F &= 10kN, E = 2 \times 10^5 kN/cm^2, \\
L &= 200cm, \sigma = 10kN/cm^2
\end{aligned}
$$

This problem has four decision variables $\mathbf{x} = [x_1, x_2, x_3, x_4]$, and its true Pareto front is shown in Fig. 2.

The main goal of multi-objective optimization algorithms is to obtain an approximation of the true Pareto front of a given MOP. In general, multi-objective optimization problems can have a Pareto front composed by a huge (possibly infinite) number of solutions. When using stochastic techniques, such as metaheuristics, the goal is thus to obtain a *Pareto front approximation* (also called approximation set), i.e., a subset of solutions that represents the true Pareto front. We want to emphasize here that we are approaching multi-objective optimization using *a posteriori* techniques, that is, we produce first an approximation of the Pareto front and then, we pass the solutions obtained to the decision maker (the civil engineer in the case of structural design problems), so that he/she can choose the most appropriate solution to be implemented, based on his/her own preferences. There are multi-objective optimization approaches
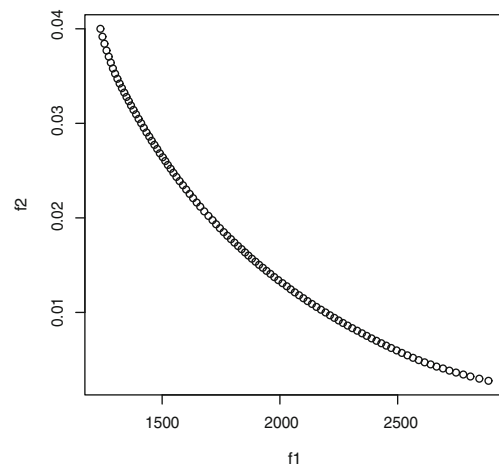

Fig. 1 Four-bar plane truss problem

Fig. 2 Pareto front of the four-bar plane truss

that can be used in an *a priori* or in *interactive* form (Coello et al. 2007), but they will not be discussed here.

In general, it does not make sense to search for a huge number of Pareto optimal solutions, since normally a reasonably low number of solutions is enough. Thus, multi-objective metaheuristics normally aim to obtain a limited set of Pareto optimal solutions (typically, around 100), although this is clearly a user-defined parameter and other values (e.g., 50, 200, or any other value) can be adopted as well, depending on the specific requirements from the user (Das 1999; Gobbi et al. 2013; di Pierro et al. 2007).

When producing an approximation of the Pareto optimal set using multi-objective metaheuristics, two main properties are normally aimed for: (1) *convergence* (i.e., we want to produce solutions as close as possible to the true Pareto front) and (2) *diversity* (i.e., we want to produce solutions that are spread along the entire Pareto set, and not only clustered around a few specific parts of it).

These concepts are illustrated in Fig. 3. The Pareto front approximation in the top left is on the true Pareto front, but there are some regions in which no solutions were found; this means that the decision maker would not know whether or not there are any useful solutions in this region. The example in the top right of Fig. 3 shows a Pareto front with an excellent distribution of points along the Pareto front, but they have not converged to the true Pareto front. This is evidently undesirable, since a good distribution of solutions is relevant only when having a good convergence. Finally, the approximation set shown in the bottom of the figure shows a desirable result which has both a good convergence and a good diversity.

It is important to indicate that, in practice, other goals may also be desirable. For example, it may be required to obtain more solutions only on a specific part of the Pareto front. This may be particularly relevant when dealing with computationally expensive objective functions (e.g., in aeronautical engineering problems). In such cases, it may happen, for example, that the solutions shown in the top left part of Fig. 3 are preferred if they can be reached in a significantly shorter time than that required to obtain the solutions shown at the bottom of the figure.

3 Metaheuristics and multi-objective optimization

In the previous section we provided a background of basic multi-objective optimization concepts. Here we include a complement of that information by presenting some fundamentals of multi-objective metaheuristics. First, we define the concept of metaheuristic and how this family of solvers can be classified. Second, we describe Evolutionary Algorithms, which are, by far the most well-known metaheuristic techniques. Third, a number of issues that are particular
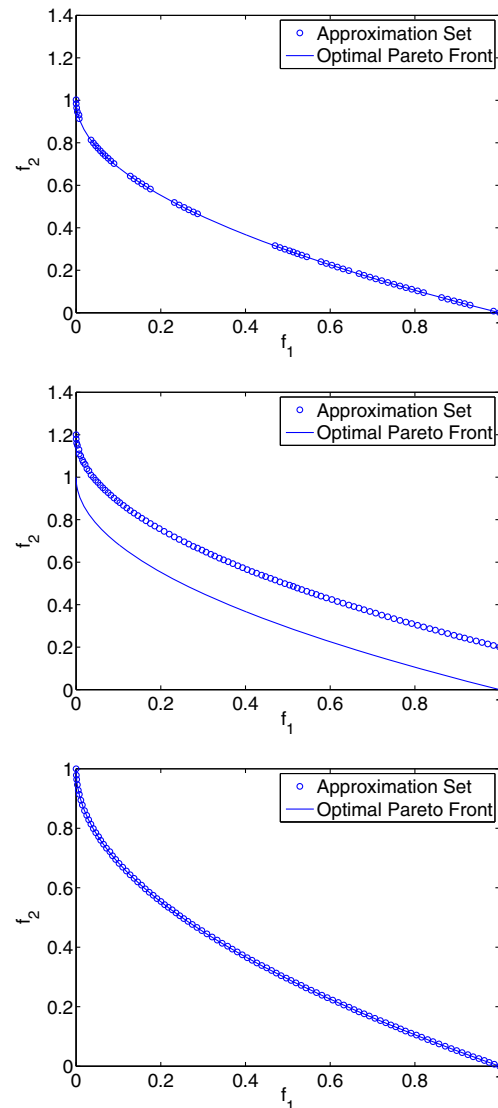

Fig. 3 Pareto front approximations: good convergence (*top left*), good diversity (*top right*), good convergence and diversity (*bottom*)

to multi-objective optimization algorithms are highlighted. Finally, we elaborate on the solution encoding, which is an important feature because of its influence in the algorithms than can be chosen to solve a specific problems.

3.1 Definition of metaheuristic and classification

Metaheuristics are a broad family of non-deterministic optimization methods aimed at finding accurate solutions to complex optimization problems when exact methods are not applicable. Metaheuristics cannot, in general, guarantee to find optimal solutions but they tend to produce near-optimal solutions with a reasonably low computational effort. This class of approaches includes, among others, Evolutionary Algorithms (by far, the most well-known

metaheuristic), Ant Colony Optimization, Scatter Search, Simulated Annealing, and Particle Swam Optimization. In fact, many metaheuristics have been designed in the last 20 years, and a number of approaches to classify them according to some criteria have been proposed (see for example Blum and Roli (2003)). One of the most accepted taxonomies is that distinguishing between *nature-inspired* and non-nature inspired metaheuristics. Table 1 contains a list of metaheuristic algorithms according to this classification.

Of the many definitions of metaheuristics that can be found in the literature, a number of basic properties of them can be stated (Blum and Roli 2003):

- They are general strategies or templates that guide the search process.
- Their goal is to provide an efficient exploration of the search space to find (near-) optimal solutions.
- They are not exact algorithms and their behavior is generally **nondeterministic** (stochastic).
- They may incorporate mechanisms to avoid visiting non promising regions of the search space.
- Their basic scheme has a predefined structure.
- They may use specific problem knowledge for the problem at hand, by using some specific heuristic controlled by the high level strategy.

Algorithm 1 Template of a metaheuristic

1: $A(0) \leftarrow$ GenerateInitialSolutions()
2: $t \leftarrow 0$
3: Evaluate($A(0)$)
4: **while not** StoppingCriterion( ) **do**
5: $\quad S(t) \leftarrow$ Generation($A(t)$)
6: $\quad$ Evaluate($S(t)$)
7: $\quad A(t+1) \leftarrow$ Update($A(t), S(t)$)
8: $\quad t \leftarrow t+1$
9: **end while**

This way, metaheuristics can be considered as high level strategies for exploring search spaces by using different low level search operators. From an algorithmic point of view, the provided templates can be instantiated or tuned to perform an efficient search of any given optimization problem. Algorithm 1 shows the pseudocode of a generic metaheuristic, where a set A of some initial solutions (A may be eventually initialized to Ø), is iteratively updated by generating a set S of new solutions from it until a stopping condition is met.

3.2 Evolutionary algorithms

As an example, we describe next an evolutionary algorithm (EA), the most popular and widely used member of the metaheuristics family. A typical EA follows the pseudocode included in Algorithm 2.

In EAs, candidate solutions are called *individuals*, which are composed of a *chromosome* (the representation of the variables of the problem) and a *fitness* (an indicator of the quality of the solution in the context of the problem being solved). Groups of individuals are referred to as *populations*. As in real life, some selected individuals from a population mate (i.e., are *selected*) for *reproduction*, generating new child or offspring individuals which, according to the natural selection process, can *replace* some other individuals. Whenever a new solution is created, it is evaluated to be assigned its corresponding fitness value. When the current population is replaced by a new one, a *generation* has taken place. The process of iterating through successive generations is called *evolution*, and ends when a termination condition is fulfilled.

Algorithm 2 Template of an evolutionary algorithm (EA) algorithm

---

1: $P(0) \leftarrow$ GenerateInitialPopulation()
2: $t \leftarrow 0$
3: Evaluate($P(0)$)
4: **while not** StoppingCriterion( ) **do**
5: $\quad P'(t) \leftarrow$ Selection($P(t)$)
6: $\quad P''(t) \leftarrow$ Reproduction($P'(t)$)
7: $\quad$ Evaluate($P''(t)$)
8: $\quad P(t+1) \leftarrow$ Replacement($P(t), P''(t)$)
9: $\quad t \leftarrow t+1$
10: **end while**

The pseudocode in Algorithm 2 can be instantiated also to yield particular EA variants. Thus, if the reproduction is based on crossover and mutation operators, the resulting EA is a Genetic Algorithm (GA), which, by the way, is the best-known EA. Other EA variants are Evolution Strategies (ESs) and Genetic Programming (GP). It is important to note that some parameters, such as the population size or the probability of applying the crossover and mutation operators, have to be carefully tuned so that the EA can have a good performance. In fact, this topic has been subject of a considerable amount of research in the evolutionary computation literature (see for example Lobo et al. (2007)).

Most metaheuristics, single and multi-objective, follow a pattern similar to the one used by EAs, i.e., the algorithms consist of a loop where a number of tentative solutions are combined and modified in a certain way, in order to produce new solutions, and at each iteration the loop tries to improve the managed solutions towards the optimum of the problem to be solved. All of these techniques have control parameters whose values have an important influence in the search capabilities of the algorithm.

Table 1 Classification of metaheuristics in Nature-inspired and Non-nature-inspired techniques

Nature-inspired	Non-nature-inspired
• Simulated annealing (SA)	• Tabu Search (TS)
• Swarm Intelligence	• Variable Neighborhood Search (VNS)
• Particle Swarm Optimization (PSO)	• Scatter Search (SS)
• Ant Colony Optimization (ACO)	• Iterated Local Search (ILS)
• Evolutionary Computation (EC)	• Guided Local Search (GLS)
• Genetic Algorithm (GA)	• Path Relinking (PR)
• Evolution Strategy (ES)	• Greedy Randomized Adaptive
• Genetic Programming (GP)	Search Procedure (GRASP)
• Estimation of Distribution	
Algorithm (EDA)	

3.3 Issues when solving multi-objective optimization problems

When the goal is to solve a multi-objective optimization problem (MOP) with metaheuristics, there are new issues to consider due to the fact that we aim to produce a set of solutions (i.e., our approximation of the Pareto set) rather than a single value. Comparisons between solutions are required in many selection operators used within metaheuristics (e.g., binary tournament within an EA) and in replacing policies. When dealing with single-objective optimization problems, the relationship "is better than" between two solutions is trivial: the one with the lower or higher fitness value, depending whether function has to be minimized or maximized, is better. In the case of MOPs, this relationship has to be redefined, because when comparing two solutions which are non-dominated there is no a way to assess which of them is better, unless we impose certain user's preferences. The consequence is that a new kind of fitness measurement is needed. On the other hand, as commented in Section 2, the set of solutions that we aim to produce has to have at the same time satisfactory convergence and diversity properties. This means that not only Pareto optimal solutions are sought for, which would guarantee a high degree of convergence, but also that such solutions must be evenly distributed along the Pareto front. Again, and in order to illustrate these issues, we will analyze next an example using the Non-dominated Sorting Genetic Algorithm-II (NSGA-II), which is the most popular multi-objective evolutionary algorithm (MOEA) in current use.

The NSGA-II (Deb et al. 2002) is a multi-objective genetic algorithm that is characterized by two features: the use of a *Pareto ranking* mechanism to classify solutions and a density estimator known as *crowding distance*. The ranking of solutions classifies a population in ranks (1, 2, ...) in such a way that the non-dominated solutions are assigned a rank equal to 1; then, they are removed and the procedure is successively applied yielding to solutions with ranks 2, 3, and so on. By selecting the solutions with best ranking, NSGA-II tries to converge towards the true Pareto front. However, when choosing the best ranked solutions it is possible that only a subset of solutions of a given rank be needed. In this case, it is necessary to carefully select the most promising solutions in order to promote diversity, and the approach taken in NSGA-II is to define a *density estimator*. The idea of a density estimator is to assign to a set of non-dominated solutions a value indicating in some way the degree of proximity (or density) of nearby solutions in the set. This way, solutions in sparse regions are preferred compared to those in most crowded regions. As indicated before, the density estimator in NSGA-II is called crowding distance, and it is detailed in Deb et al. (2002). NSGA-II has become de *de facto* multi-objective optimization metaheuristic, and it has influenced many other techniques proposed in the last ten years (Coello et al. 2007).

Algorithm 3 Pseudocode of the NSGA-II algorithm

1:   $P(0)$ GenerateInitialPopulation()
2:   $t \leftarrow 0$
3:   Evaluate($P(0)$)
4:   **while not** StoppingCriterion( ) **do**
5:     $P'(t) \leftarrow$ BinaryTournament($P(t)$)
6:     $P''(t) \leftarrow$ Crossover & Mutation($P'(t)$)
7:     Evaluate($P''(t)$)
8:     $P(t+1) \leftarrow$ Ranking($P''(t) \cup P(t)$)
9:     **if** $P(t+1)$ not full **then**
10:      $R \leftarrow$ CrowdingDistanceSelection($P''(t) \cup P(t)$)
11:      $P(t+1) \leftarrow P(t+1) \cup R$
12:     **end if**
13:     $t \leftarrow t+1$
14:   **end while**

Another popular MOEA is the Strength Pareto Evolutionary Algorithm 2 (SPEA2) (Zitzler et al. 2001). As the NSGA-II, SPEA2 is a genetic algorithm based on Pareto dominance. SPEA2 adopts a scheme known as strength, that takes into account, for each solution in the population, not

only the number of solutions that dominate it, but also the number of solutions by which it is dominated. This provides a finer grained estimation of the actual ranking of a solution. The density estimator adopted in SPEA2 is based on measuring the distance to the k-th nearest neighbor (i.e., it is a clustering approach).

NSGA-II, SPEA2 and most of the multi-objective metaheuristics developed in the last ten years conform the so-called second generation of multi-objective metaheuristics (characterized for adopting some form of elitism to retain the globally non-dominated solutions that they generate), leading to the period when research in this field grew in a very important manner, thus becoming a hot research topic. A set of representative multi-objective metaheuristics is depicted in Table 2.

A popular scheme to deal with multi-objective optimization problems before the raising of the second-generation algorithms was the use of aggregating functions. Under this sort of approach, a multi-objective optimization problem is transformed into a single-objective one, thus allowing to apply single-objective metaheuristics. However, this sort of approach has a number of drawbacks, from which the main one is that it cannot generate non-convex portions of the Pareto front (Das and Dennis 1997). Scalarizing functions are a better choice if one is interested in using single-objective optimizers for solving multi-objective optimization problems, since they don't have the same limitations of aggregating functions and can be very effective when dealing with complicated Pareto sets. When using scalarizing functions, the solution of a multi-objective optimization problem is decomposed into a set of single-objective optimizations (a set of weights are used for this sake). This idea is indeed the basis of the Multiobjective Evolutionary Algorithm Based on Decomposition (MOEAD, Zhang and Li (2008)), in which the several single-objective optimization problems generated are solved at the same time. By combining this idea with the definition of neighborhood relations among the subproblems that are being solved, MOEA/D (and, particularly, its MOEA/D-DE variant (Li and Zhang 2009)), is considered as one of the most powerful MOEAs currently available.

3.4 The influence of solution encoding

A key point when using metaheuristics (both for single- and multi-objective optimization) to solve a given problem is the encoding of the solutions. To illustrate this issue, the four-bar plane truss depicted in Fig. 2 will be used. This problem has four real variables, so a natural encoding in this case is to use an array of real values (i.e., real numbers encoding), but it is also possible to use a binary encoding, where each decision variable is encoded as a binary string. Choosing between these two representation has two consequences. First, many reproduction operators are applicable only to a given encoding; for example, single point crossover (SPX) and bit-flip mutation are used with binary strings, while simulated binary crossover (SBX) and polynomial-based mutation are operators intended for real numbers encoding. Given that each operator has a different behavior, the performance of the algorithm using it, gets affected. Second, some metaheuristics are designed to work with a given type of encoding. Differential Evolution (DE) and Particle Swarm Optimization (PSO) are typically adopted for solving continuous optimization problems, while other such as Ant Colony Optimization (ACO) or Tabu Search (TS) are intended mainly to work with combinatorial problems. Thus, encoding is an important feature that will be included in the survey (see Tables 6 and 7 in Section 5).

Table 2 Examples of representative multi-objective metaheuristics	Type	Algorithm	Citation
	Evolutionary algorithm	PAES	Knowles and Corne (1999)
		SPEA2	Zitzler et al. (2001)
		PESA-II	Corne et al. (2001)
		NSGA-II	Deb et al. (2002)
		IBEA	Zitzler and Künzli (2004)
		GDE3	Kukkonen and Lampinen (2005)
		SMS-EMOA	Beume et al. (2007)
		MOEA/D	Zhang and Li (2008)
	Particle swarm optimization	OMOPSO	Reyes and Coello Coello (2005)
		SMPSO	Nebro et al. (2009a)
	Simulated annealing	AMOSA	Bandyopadhyay et al. (2008)
	Scatter search	MOSS	Beausoleil (2006)
		AbYSS	Nebro et al. (2008b)

4 Structural design optimization problems

It is difficult to determine when was the first time that optimization was applied to structural design, but we can say as a historical review that in the ancient times, before the Roman aqueducts, there were stone arches and curved bar structures aimed at covering long spans by reducing the weight or the amount of material.

Ohsaki and Swan included a historical review in Ohsaki and Swan (2003) about a number of optimization techniques applied to discrete structural topology. There periods were stated as follows:

– Initial period where J.C. Maxwell (in 1894) and A.G.M. Michel (in 1904) made their pioneering studies in the field.
– Second period in the 1960s and 1970s, when the use of computers began to arise.
– Third period, during the 1980s and 1990s, characterized by the extremely dramatic grown in computer technologies.

The first applications of metaheuristics to structural optimization problems can be found in works developed in the 1970s and the begin of the 1980s, as it was pointed out in Kicinger et al. (2005). If we focus in multi-objective approaches, we can trace back works in the mid-1980s (Goldberg and Samtani 1986) and the beginning of the 1990s (Hajela 1990). Surveys about structural optimization including both single- and multi-objective optimization techniques were published in Andersson (2000) and in Kicinger et al. (2005). These techniques and others were discussed in Saitou et al. (2005).

The problems analyzed in these surveys can be divided in two main categories:

– **Bar or element design**: it is a local optimization process that affects the shapes and sizes of the elements of a structure. It is addressed by finding the measures of a pre-defined geometry transverse to the main axis of the structure, or determining the variables that lead to the optimum geometric shapes.
– **Topological design**: It is a global optimization process aimed at defining the entire topology (layout) of a structure. It is known as topological optimum design (TOD), and it can be applied to discrete and continuous structures. TOD takes into account the number of elements and the links or continuities among them in order to determine the optimum distribution of masses in the considered domain, with the goal of obtaining the stability of the structure. In this process, while new elements are incorporated or existing ones are removed, the shape and/or size of each element is also optimized.

4.1 Bar or element design

In this section, bar design problems (Chapman 1994) are classified. These problems are related to structures composed of flat and spatial bars, such as trusses and frames.

Truss structures are composed of bar elements that are linked by articulated joints. The calculation of the equilibrium requires to determine the areas of the cross section of each bar, with no information about shape and sizes. A frame structure is a more complex problem because, besides the areas, the flexural inertia must be calculated, so the cross shape and sizes must be previously defined in order to determine the stability of the structure. The problem complexity might increase in case of three dimensional structures, because there are four variables for the design: the area, two inertia moments, and a torsional inertia.

So, in this category, the cross geometry of each bar of the structure must be calculated, which defines the decision variables of the problem that is to be optimized. Four sub-categories can be distinguished, as illustrated in Fig. 4:

1. **Area optimization**: The decision variables represent the areas of the bars without considering the sizes nor the shapes of the cross sections (Fig. 4(1)).
2. **Size optimization**: As shown in Fig. 4(2), the shape is pre-defined, so the problem to optimize is related to finding the values that define the size of each part of the shape.
3. **Shape optimization**: The goal is to find the geometry and dimensions of a closed polygon inscribed within a known frame or border (Fig. 4(3)) by the discretization of the shape into lines or curves linked together by nodes.
4. **Topological optimization of cross-section**. The fourth sub-category is illustrated in Fig. 4(4). Starting from a known elemental section, it must be discretized into small elements having triangular or square sections. The optimization process is then based on removing pieces and re-arranging the linked elements with their neighbors by their sides. Empty spaces can appear inside.

4.2 Topological design

As commented before, TOD comprises the distribution of internal elements, the links, and the external shape of the structure to optimize, and it basically **relies** on distributing the resistant masses of material where required as well as
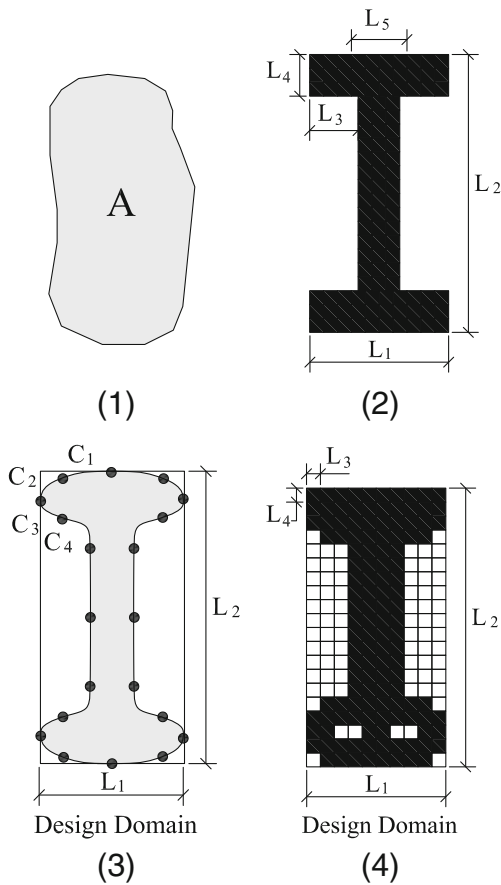
Fig. 4 Bar dimensioning or geometrical design problem variants: (1) Area without sizing, (2) sizing, (3) shape, and (4) topology cross-section

Fig. 5 Topological design classification: No topological optimization with two variants, (1a) bars and (1b) continuous structures; (2) bar topology optimization; (3) continuous topological optimization

defining their size. The optimization process starts from an initial state and the final shape, trying to find the optimum design satisfying the objective functions and constraints.

Three categories can be established to classify this type of problems. The issues to consider are the type of the used element or the way in which the subdivisions are made (bars or elements components interconnected by nodes, sides, or surfaces). The categories are the following:

1. **No topological optimization (pre-defined topological design)**: the topology of the structure is defined by the designer, and it is fixed and remains unchanged during the optimization process concerning the number of elements, links, restriction of movements, etc. This category is considered because it allows to classify those TOD problems in which only bar design issues and continuous structures are considered, as illustrated in Fig. 5(1a) and (1b) respectively. In the first case, the problems are composed of 2D or 3D bar structures where one of the measures stands of the other

two (length on the measures of the cross section), using a linear representation of the bar element that matches the centroid of the cross section describing a longitudinal axis; each element is linked to the others through its nodes. Examples are truss and frame structures. The second kind of problems include continuous

structures in which there is no variation of shape, and the optimization is limited to the thickness equal for all elements of the structure or the suitability of the main lengths of the structure.

2. **Discrete topological optimization**: this case is similar to the previous one, but the design of the structure is not defined in advance, i.e., it is unknown (see Fig. 5(2)). The starting point is the design domain, which is the zone of study, and consists of a figure or wire body, containing the boundary conditions, which can be in the 2D or 3D domain. It is divided internally by grids in which the coordinates of the vertices are known. The design of the structure shall be defined by the required unions of these points, allowing the links among the longitudinal elements to represent the bars.

3. **Continuous topological optimization**: Depicted in Fig. 5(3) it is applied to plate-like structures or continuous 2D and 3D structures. The thickness can be constant or not, so the decision variables of the optimization problem take values between the limits of the exploration space and boundary conditions. The starting point is a figure or basic body with pre-established peripheral measures and with excess material known as design domain. In both cases, 2D and 3D, the design domain is discretized in small elements. The final result will be a different figure depending on the grouping obtained with the optimization of the resistance and distribution of the material, discarding the decking material on the strength of the assembly. In general, these kinds of structures evolve to adopt bar shapes through the discrete elements linked among them by the adjacent sides between neighbors. When the problems are in 3D, the design is a spatial one and evolves by varying the sizes in the three dimensions, coupling and uncoupling volumetric finite elements attached to their neighbors through the contact surfaces. After the optimization, the required conditions of shape and size to meet the stability of the structure will be obtained.

5 The survey

After doing an in-depth analysis of the literature, we have selected more than 50 references related to multi-objective optimization applied to structural design problems, which we have found to be the most representative of the work that has been done in this area. Even though we have tried to encompass a wide and thorough review, it is important to clarify that the aim was not to be comprehensive, and there may be other relevant works that are not included here.

The analysis of this published material is divided in two parts. First, Section 5.1 provides all the details about the particular structural design problem tackled by the analyzed works. Then, an in-depth review based on the particular algorithmic details such as the algorithm used, the encoding of the tentative solutions, the search operators adopted, the number of solutions managed, etc., is provided.

5.1 Structural design problems addressed by multi-objective metaheuristics

Tables 3, 4 and 5 summarize the main features of the structural design problems that have been addressed with multi-objective metaheuristics in the analyzed papers. We use **n/a** for those cases in which we did not find the required information as well as in cases when that feature is not applicable. The entries in these tables include the following information:

- Cite: Bibliographic reference. The citations are arranged according to their publication year and, in a given year, they are presented alphabetically considering the first author name.
- Mat.: Material used in the structure, e.g., steel, wood, reinforced concrete (*RF*), prestressed concrete (*PC*), etc. If the material is not available, this field will include its mechanical feature such as (*Elastic*), when a linear relation exists between tension and distortion, or (*Plastic*), when this relation is not linear.
- Dom.: Domain of the problem. It can be either (*2D*) or (*3D*) if the analysis of the structure is carried out in the plane or the space, respectively.
- Calc.: Calculation method used to determine the effects of loads in the structure and stability. We have used the acronyms and names detailed next. SDC: Seismic Design Criteria (for problems related to seismic analysis of structures), DNV-CSR: Det Norske Veritas - Common Structural Rules (certification relating to calculus and quality of ships), DGV-CPH and NBE AE-88: Spanish construction regulations, Damage: identification of damaged members and estimation of severity, FEM: Finite Element Method, FEM-P: FEM and parallel processing, FEM-NRH: Nonlinear Response History FEM, SRSM: Stochastic Response Surface Method, CFDM: Constrained Force Density Method, Equ: Algebraic Equations, Ad Hoc: other method, different from the ones previously mentioned.
- Type: type of structure. We consider here Truss, Frame, SMRF: Special Moment-Resisting Frame, Cantilever plate, Tall building frame, etc.
- Nodes: number of nodes used in bar structures. Additionally, we use the △ (triangular cell) and □ (quadrilateral cell) symbols to represent the shape of the discrete structure forming a grid of continuous real problem 2D, and solid45 and solid95 to represent linear and quadratic hexahedral solid elements, respectively.

Table 3 Synthesis of the problems solved with MOPs

Cite	Optimized structure										
	Mat.	Dom.	Calc.	Type	Nodes	Elem.	Gr.	Design	Obj.	Con.	No Var.
Hajela and Lin (1992)	Steel	2D	FEM	Truss static load	6	10	3 to 5	1.1	W, δ	σ, δ	3 to 5
		3D		Truss dynamic load							
Coello and Christiansen (2000)	Elastic	3D	n/a	Truss wind-box	14	28	16	1.1	W, δ, σ	δ, σ	16
		2D		Truss	77	200	29				29
		3D			10	25	8				8
Deb et al. (2000)	Elastic	2D	Equ	Truss	3	2		1.2	V, σ	σ	3
Deb and Goel (2000)	Elastic	2D	FEM	Simply-supported plate	$\square$ cell	60		3	W, δ	δ, σ	n/a
				Cantilever plate							
				Hoister plate		48					
				Bicycle Frame		107					
Büche and Dornberger (2001)	Elastic	2D	FEM	Truss	6	10		1.1	W, δ	σ	10
		1D	ad hoc	Bricks tower	7	6		1.2	W, σ, Fr		6
Ray et al. (2001)	Elastic	2D	Equ	Truss	3	2		1.2	V, σ	V, σ	3
					4	4		1.1	V, δ	$A_{1\,to\,3}$	4
Hamda et al. (2002)	Elastic	2D	FEM	Plate cantilever	$\square$ cell	200		3	W, δ	σ	n/a
Ray and Liew (2002)	Elastic	2D	Equ	Truss	4	4		1.1	V, δ	$A_{1\,to\,3}$	4
Liu et al. (2003)	Steel	2D	SDC	Frame 5 storey 4 bay	30	45	25	2.1	Cost, LSDC DI	n/a	25
Kicinger and Arciszewski (2004)	Steel	2D	ad hoc	Tall building frame 36 storey 3 bay	148	252		2.2	W, δ	n/a	n/a
Luh and Chueh (2004)	Elastic	2D	Equ	Truss	6	10		1.1	W, δ	σ	10
		3D	FEM		10	25	8				25
Madeira et al. (2004)	Elastic	2D	FEM-P	Plate cantilever	$\square$ cell	768		3	F_1, F_2	V	n/a
Coello and Toscano (2005)	Elastic	2D	Equ	Truss	4	4		1.1	V, δ	$A_{1\,to\,3}$	4
					3	2		1.2	V, σ	V, σ	3
Greiner et al. (2005)	Steel	2D	n/a	Frame 5 store 5 bay	35	55		2.1	M, NCST	σ, λ, δ	55
Lagaros et al. (2005)	Steel	3D	SDC	Frame 6 storey	36	63	5	2.1	W, δ, FEV	q, δ	10
				Multi-layered Truss	3 614	12 974	6		W, δ	σ, δ	6
Liu et al. (2005)	Steel	2D	SDC	SMRF 5 storey 4 bay	30	45	11	2.1	W, NCST	n/a	n/a

Table 4 Synthesis of the problems solved with MOPs (cont)

Cite	Optimized structure										
	Mat.	Doma.	Calc.	Type	Nodes	Elem.	Gr.	Design	Obj.	Con.	No Var.
Tai and Prasad (2005)	Elastic Elastic	2D	PGCM	Target matching Compliant mechanism	□ cell	2 500		3	Dist, NEC δ_{ave}, GAM	NEC, PAS F_b, σ	n/a n/a
Valdez Peña et al. (2005)	Elastic	2D	FEM	Beam plate Beam plate	△ cell	36 144		3	W, δ	σ_{VM}	n/a
Greiner et al. (2007)	Steel	2D	FEM	Fame 2 bay Frame 5 storey 5 bay	5 36	4 55	8	2.1	W, NCST	σ, σ_B, δ	4 8
Herrero et al. (2007)	Elastic	2D	Equ	Truss	4	3		1.1	V, δ	σ	3
Kelesoglu (2007)	Elastic	3D	FEM	Truss	25	56	3	1.1	W, δ	σ, δ	3
Kicinger et al. (2007)	Steel	2D	FEM	Frame tall building 30 storey 5 bay Frame tall building 36 storey 3 bay	186 148	330 252		2.2	W, δ	n/a	n/a
Perera et al. (2007)	Elastic Aluminium	2D	Damage	Simply supported beam Beam of supported fixed	10 20	11 21		1.1	MF, MD	Da	n/a
Ohsaki et al. (2007)	Steel	2D	SDC	Frame 5 storey 4 bay	30	45	6	2.1	V, Ep	DR	6
Izui et al. (2008)	Elastic	2D 3D	n/a n/a	Truss	9 10	28 25	8	1.2 1.1	V, δ V, δ	$\pm\sigma$ $\pm\sigma, \delta$	28 8
Payá-Zaforteza et al. (2008)	RC	2D	DGV-CPH	Building frame 6 storey 2 bay	21	30	18	2.1	Cost, Env	n/a	115
Paya et al. (2008)	RC	2D	NBE AE-88	Building frame 4 storey 2 bay	15	20	12	2.1	Cost, N_s Cost, Sust Cost, γ	n/a	77
Perera and Ruiz (2008)	Elastic	2D	Damage	Continuous beam 3 bays Frame 2 storeys I-40 bridge	101 73 16	100 72 15		1.1	MF, MD	Da	n/a
Sharma et al. (2008)	Elastic	2D	PGCM	Tracing curvilinear path Tracing straight line path	□ cell	2 500		3	W, DS	σ, δ_{avg}	633
Shih and Kuan (2008)	Elastic	2D	Equ	Truss symmetric I-Beam design	4 2	3 1		1.1 2.1	W, δ A, δ	σ n/a	2 4
Wang et al. (2008)	Aluminium	2D	FEM	Cantilever plate	□ cell	1 250		3	W, δ	σ, δ, V_r	n/a
Noilublao and Bureerat (2009)	Elastic	2D 3D 2D	SDC	Truss bridge Walking tractor handlebar Truss bridge	12 16 12	21 27 25		1.1 2.1 1.2	M, $1/\omega_1$ M, $1/\sum_{i=1}^{3}\omega_i$ M, $FT(\omega_1)$ M, $\overline{FRF}$ M, $\overline{FT}$	σ	14 24 18

Table 5 Synthesis of the problems solved with MOPs (cont)

Cite	Optimized structure										
	Mat.	Doma.	Calc.	Type	Nodes	Elem.	Gr.	Design	Obj.	Con.	No Var.
Guedri et al. (2009)	Elastic	3D	SRSM	Two sub-structure assemblies of plates	□ cell	n/a		1	M, DyR	n/a	n/a
Sharma et al. (2009)	Elastic	2D	PGCM	Curvilinear path	□ cell	2 500		3	W, SIE	σ, δ_{avr}	637
Bin et al. (2010)	Stell	3D	FEM	Third flex arm of the HMAWV	solid45 solid95	715 000 670 000		1	V, L	σ_{max45} σ_{max95}	3
Su et al. (2010)	Elastic	2D	FEM	Truss	6 8	10 15		1.1 1.2	δ, W	σ	n/a n/a
Winslow et al. (2010)	Steel	3D	FEM unit cell	Dome grid Doubly curved surface	n/a	n/a		2.2	δ_1, δ_2	α, β	2/cell
Sharma et al. (2011)	Elastic	2D	PGCM	Curvilinear path Straight line generating	□ cell	625		3	W, Ei	σ, δ	637
Byrne et al. (2011)	Wood	2D	FEM	Bridge	n/a	n/a		2.2	σ, W	n/a	n/a
Descamps et al. (2011)	Elastic	3D	CFDM	Suspension bridge Arch bridge	n/a	n/a	2	2.2	M, σ_A M, L	GC	3 4
Greiner et al. (2011)	Steel	2D	FEM	Frame	5	4		2.1	CMA, SDCV	σ, λ, δ	n/a
Ghanmi et al. (2011)	Elastic	2D	FEM	Building frame 6 storey 2 bay	21	30	2	2.1	M, δ, f_M^v, f_δ^v	$\sigma, a_1 = e_1$	4/elem
Kunakote and Bureerat (2011)	Elastic	2D	FEM	Cantilever plate	□ cell	300		3	c, r $c_1, c_2,$ r c, r EVS, r	c, r $c_1, c_2,$ r $\sigma_{VM},$ r	108
Kaveh and Laknejadi (2011)	Elastic	3D	FEM	Truss	10 25	25 56	8 3	1.1	W, δ V, δ	$\pm\sigma$ $\pm\sigma, \delta$	8 3
Nafchi et al. (2011)	Elastic	2D 3D	Equ FEM	Truss	3 49	2 120	7	1.1	σ, δ W, δ	σ, δ $\pm\sigma, \delta$	2 7
Noilublao and Bureerat (2011)	Elastic	3D	SDC	Slender truss tower	4/level	16/level	6/level	1.2	M, $\sum_{i=1}^{3} F_i \delta_i$ M, $\omega_1 + \omega_2 + \omega_3$ M, $\sum_{i=1}^{3} FRF(\omega_i)$ M, $\sum_{i=1}^{3} FT(\omega_i)$	σ λ ad hoc	ad hoc
Stankovic et al. (2011)	Plastic	2D	DNV-CRS	Calculus simplified ship hull structure				2.1	M, n SDL	WS, t, σ A, p, t/h	4
Tang et al. (2011)	4 different material	2D	n/a	Square plate with a central hole	n/a	n/a	4	1	W, σ	n/a	7
	5 different material			Multilayer beam supported by springs			5			σ, δ	11

Table 5 (continued)

Cite	Optimized structure										
	Mat.	Doma.	Calc.	Type	Nodes	Elem.	Gr.	Design	Obj.	Con.	No Var.
Gong et al. (2012)	Steel	2D	FEM-NRH	Frame 3 storey 4 bay	20	27	2	2.1	W, E_i, E_{rh}	δ, ϑ	2
El Semelawy et al. (2012)	PC	3D	FEM	Flat slab	n/a	n/a		2.1	Cost, Dcb	Ec	11
Stanović et al. (2012)	Elastic	2D	FEM	Truss	6	9		2.2	M, σ	σ_B, δ, L, n^t	9
Thrall et al. (2012)	Steel	2D	BDS	Back-stay bascule	8	9		2.2	W, F	g_{BDS}	1
	Steel, piston			Piston bascule	5	5					
	Steel			Main-span operating rope	5	5					

- Elem.: number of finite elements that form the structure.
- Gr.: Number of groups of bars with similar mechanical and geometrical features. Grouping bars means simplifying the problem size with the goal of making it more affordable to be solved.
- Design: Size, shape and topology optimization problem. An $X.Y$ encoding of two variables has been used only for structures with bars, where X indicates the kind of structural component design (1: area, 2: size, 3: shape, 4: topology), as presented in Section 4.1, and Y represents the type of topology design of the structure (1: no optimization, 2: discrete, 3: continuous), as described in Section 4.2. For example, an 1.1 code in this column means optimizing the cross section area without taking into account shape or size, while the entire design of the structure remains the same (number of trusses and bars, external topology, etc.).

  In the case of continuous structures, it is enough to identify the type of optimization with one single value, which is pointed by the Y variable. Bearing in mind the classification of Section 4.2), $Y = 1$ means a size type design optimization and it corresponds only to a plate thickness optimization (not the geometry of the other two dimensions, which are fixed and predefined). The value 3 indicates a design change in the size and shape of the structure (topology design). In the 2D domain, the optimization is targeted to the main axis of the structure, being the constant the thickness or depth. In the 3D domain, the optimization is global, considering all the design variables.
- Obj.: Objective functions to be optimized.
- Const.: Constraints used to define a feasible structure.
- No Var.: Number of decision variables of the problem.

We have used the following symbols, names, and acronyms to refer to the different objectives and side constraints that are considered in the analyzed works:

- α, β: function of the geometrical parameters
- δ: displacement of node or point
- δ_{ave}: deviation between the actual output path of the mechanism and the desired path
- γ: safety
- λ: compressive slenderness or buckling factor
- ω_i: i^{th} natural frequency
- σ: axil stress limits (or allowable); in case the values of stress (+) and compression (-) are different, the corresponding sign will appear
- σ_A: stress in the anchorages
- σ_B: Euler buckling stress
- σ_{VM}: Von Misses stress
- ϑ: allowable values for rotation

- $a_1 = e_1$: restriction associated with the dimensions of each group
- A: cross sectional area
- c: structural compliance
- CMA: constrained mass average
- Cost: cost of material, design, analysis, execution, etc.
- Da: boundary constraints of the numerical values of damage that are acceptable
- Dcb: distance away from the constraint boundary
- DI: diversity index
- Dist: Euclidean distance considering the loading point, desired support point and actual support
- DR: interstory drift ratio among all stories corresponding to all the seismic motions for the initial Peak Ground Velocity (PGV) level
- DS: diversity of structure
- DyR: dynamic response
- Ei: input energy
- Ec: equations which take into account the allowable stresses of materials and other features
- Env: counts the environmental impact of the structure
- Ep: dissipated energy
- E_{rh}: hysteretic energy dissipation ratios
- EVS: first mode eigenvalue of the structure
- f^v: vulnerability function (VF) of robust multi-objective and multi-level optimization (RMOL)
- f_M^v: VF for mass of the structure
- f_δ^v VF to the maximum displacement at a specific point
- F: force or load
- Fr: frequency
- F_b: bound on the applied load needed to cause the mechanism to achieve the desired path
- FEV: first eigen period
- FRF(ω_i): frequency response function crest parameter with respect to the ω_i
- $\overline{FRF}$: mean value of FRF crest parameters at $\omega_1, \omega_2, \omega_3$
- FT(ω_i): force transmissibility crest parameter with respect to ω_i
- $\overline{FT}$: mean value of FT crest parameters at $\omega_1, \omega_2, \omega_3$
- g_{BDS}: constraint requirement that each member meets the AASHTO LRFD Bridge Design Specifications (BDS) throughout opening
- GAM: geometric advantage of the mechanism.
- GC: geometric constraints related to the position of nodes which support the distributed loads acting on the deck
- HMAWV: hybrid mode aerial working vehicle
- L: length
- LSDC: life time seismic damage cost
- M: mass
- MD: modal damage
- MF: modal flexibility

- n: number of transversely stiffening flat bars which expresses in a simple way the complexity of design or workmanship expenses
- n/a: not available
- n^t: dynamical and recursive constraint
- NCST: number of different cross-section type
- NEC: number of elements within this area that contains material
- N_s: constructability through the number of longitudinal bars
- p: lateral pressure
- PAS: prescribed area must be fully solid
- q: maximum non-dimensional ratio under a combination of axial force and bending moments
- r: normalized mass or the ratio of structural mass to the maximum mass
- SDCV: standard deviation of the constraints violation
- SDL: standard deviation of ultimate load carrying capacity taken as a measures of robustness of a structure
- SIE: supplied input energy
- Sust: environmental cost of the most sustainable solution
- t: plating of thickness
- t/h: ratio between flat bars of thickness t and height h
- V: volume
- V_r: volume ratio
- W: weight
- WS: allowable modulus section

5.2 Algorithms and parameter settings

This section aims at analyzing the literature from the point of view of the components of the multiobjective metaheuristic used for addressing structural design problems.

The columns in the tables are the following ones:

- Algorithms: name of the multiobjective algorithm used to solve the structural design problem. The algorihtms in parentheses are included for comparison purposes in the cited work.
- Fam.: metaheuristic family to which the algorithm belongs (GA: Genetic Algorithm, SA: Simulated Annealing, ES: Evolution Strategy, GP: Genetic Programming, AIS: Artificial Immune System, PSO: Particle Swarm Optimization, EDA: Estimation of Distribution Algorithm, TS: Tabu Search).
- PS: this column shows the number of solutions (PS = Population Size) used by the algorithm (if applicable).
- Encod.: representation used for the solutions manipulated by the multiobjective metaheuristics. The values that may appear in this column are: B: Binary encoding; BG: Binary Gray, IN: Integer, FP: Floating Point, DV: Discrete Values.

- Sel.: selection operator used (RW: Roulette Wheel, T: Tournament, BT: Binary Tournament, UST: Uniform Stochastic and Truncation, IS: Interval Selection, SS: Sequential Selection, RS: Random Selection). This method is the one in charge of providing (either explicitly or implicitly) an ordering for the solutions managed by the multiobjective metaheuristic.
- Rec.: recombination operator used (SBX: Simulated Binary Crossover, UX: Uniform crossover, IX: Intermediate Crossover, DPX: Two-points crossover). When more than one operator is used, we use the label "Mult.".
- Mut.: mutation operator (BF: Bit Flip, UM: Uniform Mutation, PM: Polynomial Mutation, DM: Displacement Mutation, SM: Shift Mutation, RM: Random Mutation). When more than one operator is used, we use the label "Mult.".
- FA-DP (Fitness Assignment and Diversity Preservation): these are the two most important design issues when proposing a new multiobjective metaheuristic. It is worth noting that in some cases, these two components are merged into a single measure that translates the vector of objective functions of a multiobjective problem into one single scalar value which is used to rank solutions properly. We consider here the following techniques: UIS: Uniform Interval Selection, St.: Strength, PR: Pareto Ranking, CD: Crowding Distance (density estimator of NSGA-II), WS: Weighted sum, G: Grid, AG: Adaptive Grid, SD: SPEA2's Density estimator, FS: Fitness Sharing, Cl.: Clustering

It must be noted that the columns including the selection, recombination, and mutation operators are mainly applicable to EAs (such as GA and ES). However, we can find in the literature that, for example, several PSO algorithms employ mutation as a perturbation operator. The choice of considering these three operators has its rationale in that
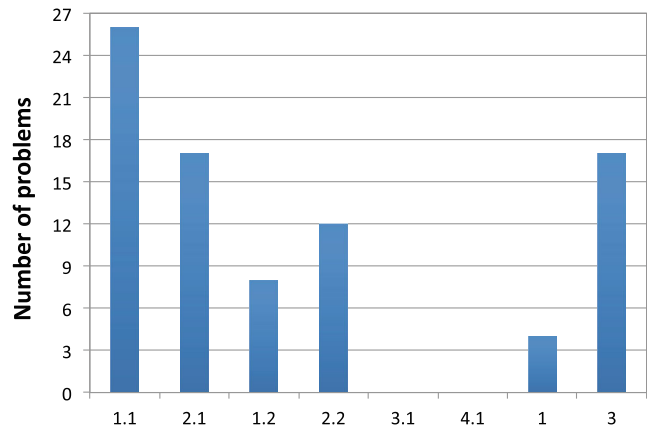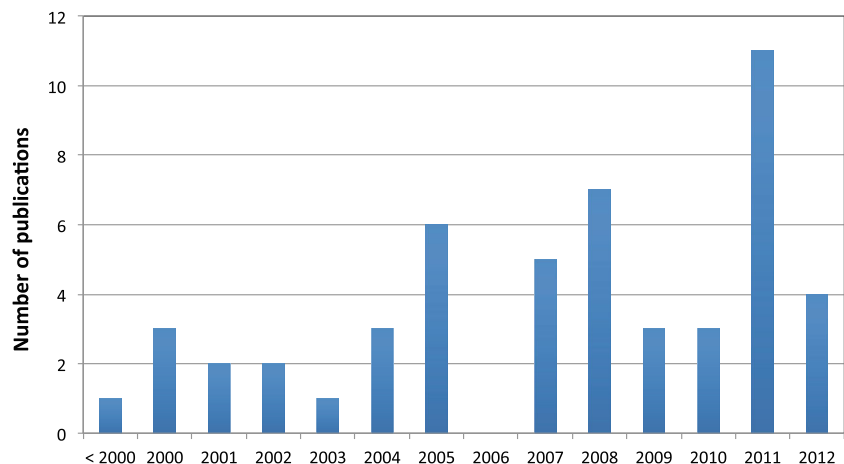

Fig. 7 Number of problems in each design category

more than the 90 % of the found works are about EAs, and the information about the used operators is, therefore, relevant. Detailing the operators used by non-EA algorithms (e.g., PSO, SA, AIS) would make the tables very complex to understand and, therefore, we omitted such details.

Whenever an item is not applicable to a given algorithm, or the paper does not report information to infer it, we write 'n/a' in the corresponding cell.

6 Analysis

In this section, we perform an analysis of the table contents presented in the previous section. We use bar plots to summarize the most relevant issues that can be extracted from all the collected data. We have considered: the number of publications per year, the type of problems addressed within each design category, and the multi-objective metaheuristics used to tackle these problems.

The total number of papers finally included in the survey is 51. Figure 6 includes the number of publications per

Fig. 6 Number of reviewed publications per year

year. The first conclusion that can be drawn from this figure is that, up to 2004, very few papers (oscillating between 1 and 3 per year, and only one before 2000) had been published. This makes sense because the use of multi-objective metaheuristics became popular at the beginning of the last decade, when the first two monographs on evolutionary multi-objective optimization were published (Deb (2001) and the first edition of Coello et al. (2007), which became available in 2002). From 2005 onwards (with the exception of 2006, in which no papers were found), the tendency increases up to 2011, where 11 papers appeared in the specialized literature. Regarding 2012, 4 papers have been considered for the survey. Our guess is that more papers are expected in the following years due to the evident benefits

that multi-objective metaheuristics have produced in structural optimization. We want to remark here that this survey is intended to serve to both expert practitioners and newcomers as the basis of cutting edge ideas and state-of-the-art algorithms to further improve this field.

We focus now on the 84 problems tackled according to the design categories established in Section 4, which are summarized in Fig. 7 and Table 6. The most studied problems fit into the categories with keys 1.1 (area optimization, no topological optimization) where 26 problems have been considered, and 2.1 (size optimization, no topological optimization), with 17 problems addressed. Continuous topological optimization (labelled as 3 in Table 7) has received also a lot interest, with 17 problems. A second group is

**Table 6** Algorithms and parameter settings

Cite	Metaheuristics							
	Algorithms	Fam.	PS	Encod.	Select.	Rec.	Mutat.	FA-DP
Hajela and Lin (1992)	Sharing GA, VEGA	GA	100	B	RW	DPX	BF	Sharing
Coello and Christiansen (2000)	GAminmax	GA	100	B, FP	BT	DPX	BF	Weighted Min-Max
Deb et al. (2000)	NSGA-II (NSGA)	GA	100	FP	BT	SBX	Pol.	PR+CD
Deb and Goel (2000)	NSGA-II + LS	GA	30	B	BT	FCX	BF	PR+CD+Cl.
Büche and Dornberger (2001)	SDM (SPEA, NSGA)	GA	60	B	IS	UX	UM	UIS
Ray et al. (2001)	MOEA	GA	n/a	n/a	Ad hoc	Ad hoc	No Mut.	PR
Hamda et al. (2002)	NSGA-II	GA	300	Voronoi	BT	Ad hoc	Mult.	PR+CD
Ray and Liew (2002)	PSO	PSO	100	FP	SS	n/a	n/a	PR+CD
Liu et al. (2003)	MOGA	GA	1000	IN	BT	DPX	UM	Weighted Min-Max
Kicinger and Arciszewski (2004)	ES	ES	12 + 60	B	UST	UX	UM	WS
Luh and Chueh (2004)	CMOIA (NSGA-II)	AIS	100, 160	B	T	Ad hoc	Ad hoc	PR
Madeira et al. (2004)	NSGA-II	GA	768	B	BT	Ad hoc	Ad hoc	PR+CD
Coello and Toscano (2005)	μGA (NSGA-II, PAES)	GA	4+100	B	BT	DPX	UM	PR+AG
Greiner et al. (2005)	NSGA-II, SPEA2	GA	100	BG	BT	UX	n/a	PR+CD/SD
Lagaros et al. (2005)	ESMO	ES	n/a	DV	Ad hoc	Ad hoc	Ad hoc	FS
Liu et al. (2005)	GA	GA	1000	IN	BT	DPX	UM	PR+CD
Tai and Prasad (2005)	GA	GA	100	B	Ad hoc	Ad hoc	Ad hoc	PR
Valdez Peña et al. (2005)	MASO	EDA		B	n/a	n/a	n/a	PR
Greiner et al. (2007)	DENSEA (NSGA-II, SPEA2)	GA	50	B, BG	RW	UX	n/a	PR+CD
Herrero et al. (2007)	ε-MOGA	GA	100	FP	RS	Ad hoc	RM	G
Kelesoglu (2007)	GA	GA	100	B	SS	DPX	BF	n/a
Kicinger et al. (2007)	SPEA2	GA	100∼500	IN	BT	UX	RM	St+SD
Perera et al. (2007)	NPGA	GA	50∼100	n/a	BT	n/a	n/a	FS
Ohsaki et al. (2007)	SA, TS	SA, TS	n/a	IN	n/a	n/a	n/a	FS
Izui et al. (2008)	MPOSO	PSO	100	FP	SS	n/a	Ad hoc	AG
Payá-Zaforteza et al. (2008)	SMOSA	SA	1	FP	n/a	n/a	n/a	n/a
Paya et al. (2008)	SMOSA	SA	1	FP	n/a	n/a	n/a	n/a
Perera and Ruiz (2008)	SPGA	GA	100	n/a	T	SPX	n/a	St.
Sharma et al. (2008)	NSGA-II	GA	240	B	BT	Ad hoc	1/n	PR+CD
Shih and Kuan (2008)	IMEA	AIS	200	FP	RS	Ad hoc	Ad hoc	WS
Wang et al. (2008)	GA+LS	GA	200	FP	n/a	n/a	n/a	PR

Table 7 Algorithms and parameter settings (cont)

Cite	Metaheuristics							
	Algorithms	Fam.	PS	Encod.	Select.	Rec.	Mutat.	FA-DP
Noilublao and Bureerat (2009)	PAES	ES	1	B	n/a	n/a	n/a	AG
	NSGA-II	GA	50	B, FP	BT	n/a	n/a	PR+CD
	SPEA2	GA	50	B, FP	BT	n/a	n/a	St.+SD
	OMPSO	PSO	50	FP	SS	n/a	n/a	AG
	PBIL	EDA	50	B	n/a	n/a	n/a	AG
Guedri et al. (2009)	NSGA	GA	20	n/a	n/a	n/a	n/a	FS
Sharma et al. (2009)	NSGA-II	GA	240	B	BT	Ad hoc	Ad hoc	PR+CD
Bin et al. (2010)	NSGA-II	GA	n/a	n/a	n/a	n/a	n/a	PR+CD
Su et al. (2010)	AMISS-MOP (NSGA-II)	GA	100	Ad hoc	BT	Ad hoc	Ad hoc	PR+CD
Winslow et al. (2010)	NSGA-II	GA	100	FP	BT	SBX, UX	Non-uniform	PR+CD
Sharma et al. (2011)	NSGA-II	GA	240	B	BT	Ad hoc	Ad hoc	PR+CD
Byrne et al. (2011)	GE	GP	100	n/a	T	n/a	n/a	PR
Descamps et al. (2011)	MOGA	GA	n/a	FP+IN	n/a	n/a	n/a	n/a
Greiner et al. (2011)	NSGA-II, SPEA2	GA	200	BG	BT	UX	n/a	PR/St.+CD/SD
Ghanmi et al. (2011)	GA	GA	30	FP	n/a	n/a	n/a	PR
Kunakote and Bureerat (2011)	PAES	ES	n/a	B	n/a	n/a	n/a	AG
	SPEA2, NSGA-II	GA	35	B, FP	n/a	n/a	n/a	PR/St.+CD/SD
	MPSO	PSO	35	FP	n/a	n/a	n/a	AG
	PBIL	EDA	35	B	n/a	n/a	n/a	AG
Kaveh and Laknejadi (2011)	CSS-MOPSO	PSO	50	FP	n/a	n/a	RM	Ad hoc
Nafchi et al. (2011)	Bees	Bees	80, 100, 150	n/a	n/a	n/a	n/a	n/a
Noilublao and Bureerat (2011)	SPEA2	GA	200	FP	BT	Ad hoc	Ad hoc	St.+SD
	AMOSA	SA	1	FP	n/a	n/a	n/a	SD
	PBIL	EDA	200	B	n/a	n/a	n/a	AG
Stankovic et al. (2011)	NSGA-II	GA	60	BG	BT	SPX	BF	PR+CD
Tang et al. (2011)	NSGA-II	GA	100	FP+B	n/a	n/a	n/a	PR+CD
Gong et al. (2012)	GA	GA	15	B	T	n/a	n/a	CD
El Semelawy et al. (2012)	SPEA	GA	n/a	n/a	n/a	n/a	n/a	St.+Cl
Stanović et al. (2012)	NSGA-II	GA	60	B	BT	Ad Hoc	BF	PR+CD
Thrall et al. (2012)	MOSA	SA	1	FP	n/a	n/a	n/a	Ad hoc

formed by the 8 and 12 problems belonging to area/size optimization plus discrete topological optimization (categories 1.2 and 2.2 respectively). Finally, no problems have been found related to the 3.1 and 4.1 types (shape/cross-section topology plus discrete no topological optimization), and 4 problems belongs to the class labelled as 1.

Front the point of view of the algorithms that have been applied, Fig. 8 clearly confirms that NSGA-II is the most widely used technique (it appears in 20 publications), as it could be expected *a priori* given the popularity of this algorithm. The second metaheuristic most frequently used is SPEA2 (7 occurrences), being the following ones different variants of MOPSO and MOSA algorithms. The rest of algorithms include a mix of techniques including PAES, PBIL, NSGA, SPEA, etc.
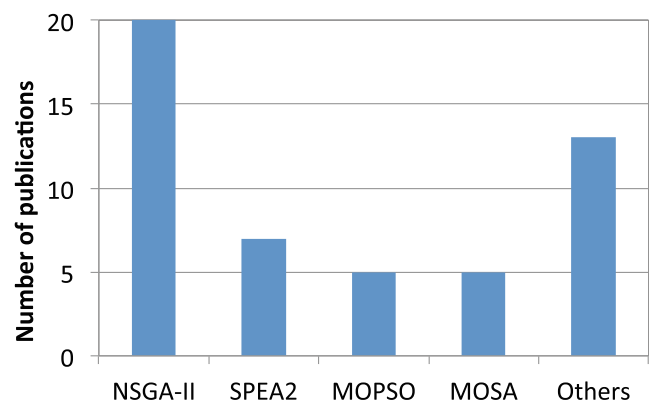
Fig. 8 Multi-objective algorithms used in the reviewed papers

7 Conclusions and perpectives

Multi-objective optimization with metaheuristics has become a hot research topic in the last decade mainly because metaheuristics have shown to be an easy and effective approach for solving real-world problems. The survey carried out in this paper confirms this fact in the field of structural design. Our purpose has been to provide the reader with a comprehensive review of relevant papers with the goal of offering a source of useful information about the most salient studies in this area.

From the analyzed information, a summary of issues and a number of open research lines can be indicated. We can subdivide them in two classes: (1) those related to structural design problems and (2) those having to do with the multi-objective techniques.

Design problems From the analyzed papers in Tables 3, 4, and 5 and Fig. 7, we can conclude that most of the studied engineering problems related to area dimensioning with no structural topology design (1.1) have been chosen because they are typically adopted as benchmarks to test multi-objective algorithms. In these problems, the use of static loads and truss structures is common (e.g., truss bridges). Somewhat more complex studies are those of class 2.2, e.g., tall building frames, which include the design of bar sizes as well as structural topological design. Similarly, problems of designs 1.2 (trusses), 2.1 (frames), 1 (slab, cantilever plate, etc.) and 3 (cantilever plate, beam, etc.) have been addressed in the specialized literature. Among the problems that have not been studied yet, we have the optimization of the cross sections of bars defining the shape of the boundary (design 3.1) and the transversal topology (design 4.1).

Multi-objective metaheuristics A rather quick look at the tables and figures suggests that, from an algorithmic point of view, there are many open research lines which can be largely exploited. For example, many of the problems reported in the specialized literature are encoded using floating point representation (i.e., real numbers are used for all the decision variables), and we found no study in which differential evolution was used to solve any of these problems. This is rather surprising if we consider that differential evolution is a very powerful approach for solving problems in which all the decision variables are real numbers. Some of the multi-objective metaheuristics based on differential evolution that could be used for these problems are GDE3 (Kukkonen and Lampinen 2005) and MOSADE (Huang et al. 2009).

Another interesting finding is that NSGA-II and SPEA2 are still the most commonly used multi-objective metaheuristics in the literature on structural optimization. This is a bit surprising considering that these two MOEAs were introduced more than 10 years ago. Remarkably, the use of other, more recent, MOEAs such as those based on decomposition (e.g., MOEA/D by Zhang and Li (2008) and Li and Zhang (2009)), MOEAs based on structuring the populations into a grid (Nebro et al. 2009b), indicator-based MOEAs (e.g., SMS-EMOA by Beume et al. (2007) and HyPE by Bader and Zitzler (2011)), algorithms based on local approximations (e.g., ANC by Gobbi et al. (2013), NBI by Das and Dennis (1998), UPS-MOEA by Aittokoski and Miettinen (2010)), and non-evolutionary metaheuristics such as scatter search (Nebro et al. 2008a), has not been reported yet in structural optimization problems. The use of coevolution and game theory (Sim and Kim 2004; Tan and Teo 2009) in structural optimization would also be an interesting choice. In fact these approaches may be quite useful in some applications (Barbosa 1997). For example, coevolutionary approaches are known to be very effective for dealing with large-scale optimization problems (i.e., problems having a very large number of decision variables) (see for example the work of Li and Yao (2012)). However, its use in large-scale multi-objective optimization has been very rare until now.

The use of parallel MOEAs (Nebro et al. 2005), surrogate methods (Ray et al. 2009), fitness approximation (Reyes Sierra and Coello Coello 2005) and incorporation of domain knowledge during the search (Landa-Becerra et al. 2008) could also bring important savings in terms of CPU time when solving multi-objective structural optimization problems. However, the use of these techniques has been very relatively scarce until now in multi-objective optimization problems (Ray and Smith 2006).

Finally, another aspect that has been only scarcely explored in multi-objective structural optimization using metaheuristics, is the incorporation of user's preferences in the search engine (Sanchis et al. 2008). These preferences allow, for example, to focus the search into a specific region of the Pareto front, and also helps the decision maker to choose one (or very few) solution from the many that a multi-objective metaheuristic normally generates (Thiele et al. 2009).

Evidently, more work in these directions is expected to appear in the next few years, as we expect that the use of multi-objective metaheuristics becomes more common not only in structural optimization, but also in engineering in general.

References

Aittokoski T, Miettinen K (2010) Efficient evolutionary approach to approximate the Pareto-optimal set in multiobjective optimization, ups-emoa. Optim Methods Softw 25(6):841–858

Andersson J (2000) A survey of multiobjective optimization in engineering design. Tech. Rep. LiTH-IKP-R-1097, Department of Mechanical Engineering, Linköping University, Linköping, Sweden

Beausoleil PR (2006) MOSS: multiobjective scatter search applied to non-linear multiple criteria optimization. Eur J Oper Res 169(2):426–449

Bader J, Zitzler E (2011) HypE: an algorithm for fast hypervolume-based many-objective optimization. Evol Comput 19(1):45–76

Bandyopadhyay S, Saha S, Maulik U, Deb K (2008) A simulated annealing-based multiobjective optimization algorithm: AMOSA. IEEE Trans Evol Comput 12(3):269–283

Barbosa HJC (1997) A coevolutionary genetic algorithm for a game approach to structural optimization. In: Bäck T (ed) Proceedings of the seventh international conference on genetic algorithms, michigan state university. Morgan Kaufmann Publishers, San Mateo, pp 545–552

Beume N, Naujoks B, Emmerich M (2007) SMS-EMOA: multiobjective selection based on dominated hypervolume. Eur J Oper Res 181(3):1653–1669

Bin X, Nan C, Huajun C (2010) An integrated method of multi-objective optimization for complex mechanical structure. Advances in Engineering Software 41(2):277–285

Blum C, Roli A (2003) Metaheuristics in combinatorial optimization: overview and conceptual comparison. ACM Comput Surv 35(3):268–308

Büche D, Dornberger R (2001) New evolutionary algorithm for multi-objective optimization and its application to engineering design problems. In: Fourth world congress of structural and multidisciplinary optimization

Byrne J, Jonathan F, Fenton M, Hemberg E, McDermott J, O'Neill M, Shotton E, Nally C (2011) Combining structural analysis and multi-objective criteria for evolutionary architectural design. In: Proceedings of the 2011 international conference on applications of evolutionary computation - Volume Part II, Springer-Verlag, Berlin, Heidelberg, EvoApplications'11, pp 204–213

Chapman C (1994) Structural topology optimization via the genetic algorithm. Massachusetts Institute of Technology, Department of Mechanical Engineering

Coello CA, Toscano G (2005) Multiobjective Structural Optimization using a Micro-Genetic Algorithm. Struct Multidiscip Optim 30(5):388–403

Coello CA, Lamont GB, Van Veldhuizen DA (2007) Evolutionary algorithms for solving multi-objective problems. Springer

Coello CAC, Christiansen AD (2000) Multiobjective optimization of trusses using genetic algorithms. Comput Struct 75(6):647–660

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). Morgan Kaufmann Publishers, San Francisco, pp 283–290

Das I (1999) A preference ordering among various Pareto optimal alternatives. Struct Optim 18(1):30–35

Das I, Dennis J (1997) A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems. Struct Optim 14(1):63–69

Das I, Dennis J (1998) Normal-boundary intersection: a new method for generating the Pareto surface in nonlinear multi-criteria optimization problems. SIAM J Optim 8(3):631–657. doi:10.1137/S1052623496307510

Deb K (2001) Multi-objective optimization using evolutionary algorithms. Wiley

Deb K, Goel T (2000) Multi-objective evolutionary algorithms for engineering shape design. KanGAL report 200003, Indian Institute of Technology, Kanpur, India

Deb K, Patrap A, Moitra S (2000) Mechanical component design for multi-objective using elitist non-dominated sorting GA. KanGAL report 200002, Indian Institute of Technology, Kanpur, India

Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 6(2):182–197

Descamps B, Coelho RF, Ney L, Bouillard P (2011) Multicriteria optimization of lightweight bridge structures with a constrained force density method. Comput Struct 89(3–4):277–284

di Pierro F, Khu ST, Savic DA (2007) An investigation on preference order ranking scheme for multiobjective evolutionary optimization. IEEE Trans Evol Comput 11(1):17–45

El Semelawy M, Nassef A, El Damatty A (2012) Design of prestressed concrete flat slab using modern heuristic optimization techniques. Expert Syst Appl 39(5):5758–5766

Ghanmi S, Guedri M, Bouazizi ML, Bouhaddi N (2011) Robust multi-objective and multi-level optimization of complex mechanical structures. Mechanical Systems and Signal Processing 25(7):2444–2461

Gobbi M, Guarneri P, Scala L, Scotti L (2013) A local approximation based multi-objective optimization algorithm with applications. Optim Eng:1–23. doi:10.1007/s11081-012-9211-5

Goldberg D, Samtani M (1986) Engineering optimization via genetic algorithm. In: Proceedings of the ninth conference on electronic computation, pp 471–482

Gong Y, Xue Y, Xu L, Grierson D (2012) Energy-based design optimization of steel building frameworks using nonlinear response history analysis. J Constr Steel Res 68(1):43–50

Greiner D, Winter G, Emperador JM, Galván B (2005) Gray coding in evolutionary multicriteria optimization: application in frame structural optimum design. In: Coello Coello CA, Hernández Aguirre A, Zitzler E (eds) Evolutionary multi-criterion optimization. Third international conference, EMO 2005, Springer. Lecture notes in computer science, vol 3410. Guanajuato, pp 576–591

Greiner D, Emperador JM, Winter G, Galván B (2007) Improving computational mechanics optimum design using helper objectives: an application in frame bar structures. In: Obayashi S, Deb K, Poloni C, Hiroyasu T, Murata T (eds) Evolutionary multi-criterion optimization, 4th international conference, EMO 2007, Springer. Lecture notes in computer science, vol 4403. Matshushima, pp 575–589

Greiner D, Galván B, Emperador J, Méndez M, Winter G (2011) Introducing reference point using g-dominance in optimum design considering uncertainties: an application in structural engineering. In: Takahashi R, Deb K, Wanner E, Greco S (eds) Evolutionary multi-criterion optimization, lecture notes in computer science, vol 6576. Springer, Berlin/Heidelberg, pp 389–403

Guedri M, Ghanmi S, Majed R, Bouhaddi N (2009) Robust tools for prediction of variability and optimization in structural dynamics. Mech Syst Signal Process 23(4):1123–1133

Hajela P (1990) Genetic search—an approach to the nonconvex optimization problem. AIAA J 26:1205–1212

Hajela P, Lin C (1992) Genetic search strategies in multicriterion optimal design. Struct Multidiscip Optim 4(2):99–107

Hamda H, Roudenko O, Schoenauer M (2002) Application of a multi-objective evolutionary algorithm to topological optimum design. In: Parmee I (ed) Fifth international conference on adaptive computing in design and manufacture. Springer-Verlag

Herrero J, Martínez M, Sanchis J, Blasco X (2007) Well-distributed Pareto front by using the ϵ-moga evolutionary algorithm. In: Proceedings of the 9th international work conference on artificial neural networks, IWANN'07. Springer-Verlag, Berlin, Heidelberg, pp 292–299

Huang VL, Zhao SZ, Mallipeddi R, Suganthan PN (2009) Multi-objective optimization using self-adaptive differential evolution algorithm. In: IEEE congress on evolutionary optimization (CEC), pp 190–194

Institution of Civil Engineers (2012) About civil engineering. http://www.ice.org.uk/About-civil-engineering

Izui K, Nishiwaki S, Yoshimura M, Nakamura M, Renaud J (2008) Enhanced multiobjective particle swarm optimization in combination with adaptive weighted gradient-based searching. Eng Optim 40(9):789–804

Kaveh A, Laknejadi K (2011) A hybrid multi-objective optimization and decision making procedure for optimal design of truss structures. IJST Trans Civ Eng 35(C2):137–154

Kelesoglu O (2007) Fuzzy multiobjective optimization of truss-structures using genetic algorithm. Adv Eng Softw 38:712–721. Elsevier

Kicinger R, Arciszewski T (2004) Multiobjective evolutionary design of steel structures in tall buildings. American institute of aeronautics and astronautics press reston, VA, AIAA 2004-6438, AIAA 1st intelligent systems technical conference chicago, IL

Kicinger R, Arciszewski T, De Jong K (2005) Evolutionary computation and structural design: a survey of the state-of-the-art. Comput Struct 83:1943–1978

Kicinger R, Obayashi S, Arciszewski T (2007) Evolutionary multiobjective optimization of steel structural systems in tall buildings. In: Obayashi S, Deb K, Poloni C, Hiroyasu T, Murata T (eds) Evolutionary multi-criterion optimization, 4th international conference, EMO 2007, Springer. Lecture notes in computer science, vol 4403. Matshushima, pp 604–618

Knowles J, Corne D (1999) The Pareto archived evolution strategy: a new baseline algorithm for multiobjective optimization. In: Proceedings of the 1999 congress on evolutionary computation. IEEE Press, Piscataway, pp 98–105

Kukkonen S, Lampinen J (2005) GDE3: the third evolution step of generalized differential evolution. In: IEEE congress on evolutionary computation (CEC'2005), pp 443–450

Kunakote T, Bureerat S (2011) Multi-objective topology optimization using evolutionary algorithms. Eng Optim 43(5):541–557

Lagaros N, Papadrakakis M, Plevris V (2005) Multiobjective optimization of space structures under static and seismic loading conditions. In: Abraham A, Jain L, Goldberg R (eds) Evolutionary multiobjective optimization, Advanced information and knowledge processing. Springer, London, ISBN: 978-1-85233-787-2, pp 273–300

Landa-Becerra R, Santana-Quintero LV, Coello Coello CA (2008) Knowledge incorporation in multi-objective evolutionary algorithms. In: Ghosh A, Dehuri S, Ghosh S (eds) Multi-objective evolutionary algorithms for knowledge discovery from data bases. Springer, Berlin, pp 23–46

Li H, Zhang Q (2009) Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II. IEEE Trans Evol Comput 2(12):284–302

Li X, Yao X (2012) Cooperatively coevolving particle swarms for large scale optimization. IEEE Trans Evol Comput 16(2):210–224

Liu M, Burns S, Wen Y (2003) Optimal seismic design of steel frame buildings based on life cycle cost considerations. Earthq Eng Struct Dyn 32(9):1313–1332

Liu M, Burns S, Wen Y (2005) Multiobjective optimization for performance-based seismic design of steel moment frame structures. Earthq Eng Struct Dyn 34(3):289–306

Lobo FG, Lima CF, Michalewicz Z (eds) (2007) Parameter setting in evolutionary algorithms. Springer, Berlin

Luh G, Chueh C (2004) Multi-objective optimal design of truss structure with immune algorithm. Comput Struct 82(11–12):829–844

Madeira JA, Rodrigues H, Pina H (2004) Multi-objective optimization of structures topology by genetic algorithms. Civil-Comp Ltd Elsevier Ltd 1(36):21–28

Marler R, Arora J (2004) Survey of multi-objective optimization methods for engineering. Struct Multidiscip Optim 26(6):369–395

Miettinen KM (1999) Nonlinear multiobjective optimization. Kluwer Academic, Boston

Nafchi A, Moradi A, Ghanbarzadeh A, Rezazadeh A, Soodmand E (2011) Solving engineering optimization problems using the bees algorithm. In: 2011 IEEE colloquium on humanities, science and engineering (CHUSER), pp 162–166

Nebro A, Luna F, Talbi EG, Alba E (2005) Parallel multiobjective optimization. In: Alba E (ed) Parallel metaheuristics. Wiley-Interscience, New Jersey, pp 371–394

Nebro A, Luna F, Alba E, Dorronsoro B, Durillo JJ, Beham A (2008a) AbYSS: adapting scatter search to multiobjective optimization. IEEE Trans Evol Comput 12(4):439–457

Nebro AJ, Luna F, Alba E, Dorronsoro B, Durillo JJ, Beham A (2008b) AbYSS: adapting scatter search to multiobjective optimization. IEEE Trans Evol Comput 12(4):439–457

Nebro A, Durillo J, Garcia-Nieto J, Coello Coello C, Luna F, Alba E (2009a) Smpso: a new pso-based metaheuristic for multi-objective optimization. In: IEEE symposium on computational intelligence in miulti-criteria decision-making, 2009. MCDM '09, pp 66–73

Nebro A, Durillo J, Luna F, Dorronsoro B, Alba E (2009b) Mocell: a cellular genetic algorithm for multiobjective optimization. Int J Intell Syst 24(7):726–746

Noilublao C, Bureerat S (2009) Simultaneous topology, shape and sizing optimisation of skeletal structures using multiobjective evolutionary algorithms. In: dos Santos WP (ed) Evolutionary Computation, ISBN: 978-953-307-008-7, InTech, pp 487–508

Noilublao N, Bureerat S (2011) Simultaneous topology, shape and sizing optimisation of a three-dimensional slender truss tower using multiobjective evolutionary algorithms. Comput Struct 89(23–24):2531–2538

Ohsaki M, Swan CC (2003) Topology and geometry optimization of trusses and frames. In: SAB (ed) Recent advances in optimal structural design, amer society of civil engineers, pp 107–133

Ohsaki M, Kinoshita T, Pan P (2007) Simulated annealing for multiobjective control system design. Earthq Eng Struct Dyn 36(11):1481–1495

Paya I, Yepes V, González-Vidosa F, Hospitaler A (2008) Multiobjective optimization of concrete frames by simulated annealing. Comput Aided Civ Infrastruct Eng 23(8):596–610

Payá-Zaforteza I, Yepes V, González-Vidosa F, Hospitaler A (2008) Cost versus sustainability of reinforced concrete building frames by multiobjective optimization. In: First international symposium on life cycle civil engineering (IALCCE08), pp 953–958

Perera R, Ruiz A (2008) A multistage fe updating procedure for damage identification in large-scale structures based on multiobjective evolutionary optimization. Mech Syst Signal Process 22(4):970–991

Perera R, Ruiz A, Manzano C (2007) An evolutionary multiobjective framework for structural damage localization and quantification. Eng Struct 29(10):2540–2550

Ray T, Liew K (2002) A swarm metaphor for multiobjective design optimization. Eng Optim 34(2):141–153

Ray T, Smith W (2006) A surrogate assisted parallel multiobjective evolutionary algorithm for robust engineering design. Eng Optim 38(8):997–1011

Ray T, Tai K, Seow K (2001) Multiobjective design optimization by an evolutionary algorithm. Eng Optim 33(4):399–424

Ray T, Isaacs A, Smith W (2009) Surrogate assisted evolutionary algorithm for multi-objective optimization. In: Pandu RG (ed) Multi-objective optimization techniques and applications in chemical engineering, chap 5. World Scientific, Singapore, pp 131–152

Reyes M, Coello Coello C (2005) Improving PSO-based multi-objective optimization using crowding, mutation and ϵ-dominance. In: Coello C, Hernández A, Zitler E (eds) Third international conference on evolutionary multicriterion optimization, EMO 2005, Springer, LNCS, vol 3410, pp 509–519

Reyes Sierra M, Coello Coello CA (2005) A study of fitness inheritance and approximation techniques for multi-objective particle swarm optimization. In: 2005 IEEE congress on evolutionary computation (CEC'2005), vol 1. IEEE Service Center, Edinburgh, pp 65–72

Saitou K, Izui K, Nishiwaki S, Papalambros P (2005) A survey of structural optimization in mechanical product development. Comput Inf Sci Eng ASME 5:214–226

Sanchis J, Martinez M, Blasco X (2008) Multi-objective engineering design using preferences. Eng Optim 40(3):253–269

Sharma D, Deb K, Kishore N (2008) Towards generating diverse topologies of path tracing compliant mechanisms using a local search based multi-objective genetic algorithm procedure. In: 2008 congress on evolutionary computation (CEC'2008). IEEE Service Center, Hong Kong, pp 2004–2011

Sharma D, Deb K, Kishore N (2009) Developing multiple topologies of path generating compliant mechanism (pgcm) using evolutionary optimization. Tech. rep., Kangal Report No. 2009002

Sharma D, Deb K, Kishore N (2011) Domain-specific initial population strategy for compliant mechanisms using customized genetic algorithm. Struct Multidiscip Optim 43:541–554

Shih CJ, Kuan TL (2008) Immune based hybrid evolutionary algorithm for Pareto engineering optimization. Tamkang J Sci Eng 11(4):395–402

Sim KB, Kim JY (2004) Solution of multiobjective optimization problems: coevolutionary algorithm based on evolutionary game theory. Artif Life Robot 8(2):174–185

Stankovic T, Ziha K, Marjanovic D (2011) Tracing engineering evolution with evolutionary algorithms. In: Kita E (ed) Evolutionary algorithms, ISBN: 978-953-307-171-8, InTech, pp 247–268

Stanović T, Storga M, Marjanović D (2012) Synthesis of truss structure designs by NSGA-II and NodeSort algorithm. J Mech Eng 58(3):203–212

Su R, Wang X, Gui L, Fan Z (2010) Multi-objective topology and sizing optimization of truss structures based on adaptive multi-island search strategy. Struct Multidiscip Optim 43(2):275–286

Tai K, Prasad J (2005) Multiobjective topology optimization using a genetic algorithm and a morphological representation of geometry. In: 6th world congress of structural and multidisciplinary optimization

Tan TG, Teo J (2009) Improving the performance of multiobjective evolutionary optimization algorithms using coevolutionary learning. In: Chiong R (ed) Nature-inspired algorithms for optimisation. Springer, Berlin, pp 457–487

Tang X, Bassir D, Zhang W (2011) Shape, sizing optimization and material selection based on mixed variables and genetic algorithm. Optim Eng 12:111–128

Thiele L, Pekka J, Korhonen KM, Molina J (2009) A preference-based evolutionary algorithm for multi-objective optimization. Evol Comput 17(3):411–436

Thrall A, Adriaenssens S, Paya-Zaforteza I, Zoli T (2012) Linkage-based movable bridges: design methodology and three novel forms. Eng Struct 37:214–223

Valdez Peña S, Botello Rionda S, Hernández Aguirre A (2005) Multiobjective shape optimization using estimation distribution algorithms and correlated information. In: Coello Coello CA, Hernández Aguirre A, Zitzler E (eds) Evolutionary multi-criterion optimization. Third international conference, EMO 2005, Springer. Lecture notes in computer science, vol 3410. Guanajuato, pp 664–676

Wang N, Yang Y, Tai K (2008) Hybrid genetic algorithm for designing structures subjected to uncertainty. In: IEEE international conference on systems, man and cybernetics, 2008. SMC 2008, pp 565–570

Winslow P, Pellegrino S, Sharma S (2010) Multi-objective optimization of free-form grid structures. Struct Multidiscip Optim 40:257–269

Zhang Q, Li H (2008) MOEA/D: a multiobjective evolutionary algorithm based on decomposition. IEEE Trans Evol Comput 8(11):712–731

Zitzler E, Künzli S (2004) Indicator-based selection in multiobjective search. In: Proceedings 8th international conference on parallel problem solving from nature (PPSN VIII). Springer, pp 832–842

Zitzler E, Laumanns M, Thiele L (2001) SPEA2: improving the strength Pareto evolutionary algorithm. Tech. Rep. 103, Computer engineering and networks laboratory (TIK), Swiss federal institute of technology (ETH) Zurich, Gloriastrasse 35, CH-8092 Zurich, Switzerland