Multiobjective Evolutionary Algorithms:
a survey of the state-of-the-art

Francisco J. Palmero Moya
August 30, 2023

Abstract

In almost no other field of computer science, the idea of using bio-inspired search paradigms has been so useful as in solving multiobjective optimization problems. This document delves into the utilization of one such paradigm, namely evolutionary techniques, for tackling multiobjective optimization problems. In fact, many real-world applications involve several conflicting objectives that can be solved using multiobjective evolutionary algorithms (MOEAs). We begin by introducing these class of problems and the particularly important notion of Pareto optimality. We then look at some of the current state-of-the-art MOEAs and present a classification based on the ways in which they make use of concepts of different evolutionary spaces and techniques for preserving diversity within the population.

1 Introduction

In the realm of computer science, few areas have embraced the application of bio-inspired search paradigms as effectively as the resolution of multiobjective optimization problems (MOPs) [ED18]. In MOPs, the quality of a solution is defined by its performance in relation to several, possibly conflicting, objectives. In practice, it turns out that many applications that have traditionally been tackled by defining a single objective function have a multiobjective problem nature that has been transformed into a single-objective problem in order to simplify the optimization. This is done by assigning a numerical quality function to each objective, and then combine these scores into a single fitness score using some weighting. This approach, often called scalarisation, has been used for many years within the operations research and heuristic optimization fields [DK01]. However, scalarisation suffers from drawbacks related to how we define the weights [ES03, Chapter 12][ED18, Section 4].

The problem becomes challenging when the objectives are of conflict to each other, that is, the optimal solution of an objective function is different from that of the other. In solving such problems, with or without the presence of constraints, these problems give rise to a set of trade-off optimal solutions, popularly known as Pareto-optimal solutions. Mathematically, we can define MOP as follows

Definition (Multiobjective optimization problem[1])**.** *A type of problem where given a function* $\vec{f} : M \subseteq \Omega \to \mathbb{R}^n$, $M \neq 0$, *the objective is to find a solution* $\vec{x}^* = (x_1^*, ..., x_k^*) \in M$ *that minimize* $\vec{f}$ *such that* $\vec{f}^* := \vec{f}(\vec{x}^*) \leq \vec{f}(\vec{x})$ *for all* $\vec{x} \in M$ *subject to constraints* $g_i : M \to \mathbb{R}$ *for* $i = 1, ..., q$.

The region M where $g_i(\vec{x}) \geq 0$ for all $i = 1, ..., q$ is called the feasible region of the objective function, any $x \in M$ is called a feasible solution, and Ω is the search space. If $M := \{\vec{x} \in \mathbb{R}^N \mid g_i(\vec{x}) \geq 0,\ \forall i \in \{1, ..., q\}\} = \Omega$ the problem is called unconstrained. Thus, given this framework, the following Pareto definitions are

Definition (Pareto dominace)**.** *A solution* $\vec{u} \in M$ *is said to dominate another solution* $\vec{v} \in M$ *if*

$$\left[f_i(\vec{u}) \leq f_i(\vec{v}),\ \forall i \in \{1, ..., q\}\right] \wedge \left[\exists i \in \{1, ..., q\} : f_i(\vec{u}) < f_i(\vec{v})\right]; \tag{1}$$

The relation is denoted by $\vec{u} \prec \vec{v}$, *implying that* $\vec{f}(\vec{u})$ *is partially less than* $\vec{f}(\vec{v})$.

[1] In general, we would demand $n > 1$ when we talk about multiobjective optimization problems. Moreover, there is the convention to call problems with large n, not multiobjective optimization problems but many-objective optimization problems. Some examples are [Wan+15; WZY23]

Definition (Pareto Optimal)**.** *A solution $\vec{u}$ is Pareto optimal if there is no other solution $\vec{v} \in M$ such that $\vec{v} \prec \vec{u}$. A set $\mathcal{P}^*$ that contains all the Pareto solutions is called Pareto optimal set, i.e.,*

$$\mathcal{P}^* = \{\vec{u} \in M : \vec{u} \prec \vec{v}, \forall \vec{v} \in M\} \tag{2}$$

Definition (Pareto Front)**.** *A set $\mathcal{F}^*$ that contains all the objectives of Pareto optimal set $\mathcal{P}^*$ is called Pareto front, i.e.,*

$$\mathcal{F}^* = \left\{\vec{f}(\vec{u}) : \vec{u} \in \mathcal{P}^*\right\} \tag{3}$$

All the solutions in the Pareto front possess the attribute that their quality cannot be increased with respect to any of the objective functions without detrimentally affecting on of the others. In the presence of constraints, such solutions usually lie on the edge of the feasible regions of the search space.

The complex nature of search space of some objectives in MOOPs such as non-differentiability and discontinuity makes traditional gradient-based techniques impossible or at least ineffective to apply. As such, researchers have proposed many population-based heuristic optimization methods that are not dependent on the characteristics of the search space. Such methods target to find as many Pareto optimal solutions as possible.

Evolutionary Algorithm Approaches to Mutiobjective Optimization

Evolutionary Algorithms (EA) have a proven ability to identify high-quality solutions in high-dimensional search spaces containing difficult features such as discontinuity and multiple constraints. If we also take into account their population-based nature and their ability for finding and preserving diverse sets of good solutions, it is not surprising that EA-based methods are currently the state-of-the-art in many MOOPs.

Multiobjective evolutionary algorithms (MOEAs) are typically designed to gradually approach sets of Pareto optimal solutions that are well-distributed across the Pareto front. As there are in general no single best solutions in multiobjective optimization, the selection schemes of such algorithms differ from those used in single-objective optimization. Although one fundamental difference between single and multiple objective optimization lies in the cardinality in the optimal set, from a practical standpoint a user needs only one solution, no matter whether the associated optimization problem is single or multiobjective

With some exceptions [SG20], the distinction between different classes of evolutionary multiobjective optimization algorithms is mainly due to the differences in the paradigms used to define the selection operators [ED18; Liu+22], whereas the choice of the variation operators is generic and dependent on the problem[2]. A much more exhaustive description of existing MOEAs methods can be found in Zhou et al. [Zho+11]. There are currently three main paradigms for MOEA designs:

Pareto-based The framework of Pareto-based MOEAs employs a two-level scheme. The initial ranking is determined by the Pareto dominance relationship, while the subsequent ranking at the secondary level is influenced by the contributions of points to enhancing diversity. This secondary ranking is particularly relevant when dealing with points that occupy identical positions in the primary ranking.

Indicator-based In the indicator-based approach emphasis is placed on an indicator that quantifies the performance of a set, such as the hypervolume indicator[3] or the R2 indicator[4]. These algorithmic approaches are constructed with the principle that enhancements in relation to this specific indicator play a pivotal role in shaping the selection process or the prioritization of individuals.

Decomposition-based Here, the algorithm dissects the main problem into multiple subproblems, each strategically addressing distinct parts of the Pareto front. For every individual subproblem, a different parameterization or weighting of a scalarisation technique is applied.

The taxonomy presented by Suárez and Galán [SG20] is based on the different moments during the optimization problem where the user specify its preferred solutions. Thus, we have

[2] Notice that the choice of representation, and hence variation operators, are entirely problem dependent.

[3] Hypervolume indicator quantifies the volume of the objective space that is dominated by the solutions in a particular Pareto front.

[4] The R2 indicator is based on the idea of linear regression and measures how well the solutions obtained approximate the true Pareto front.

A priori These techniques involve assessing the significance of each objective before the search process begins. However, determining this significance is often complex for real-world optimization problems. The usual methods for determining objective importance are explicit ordering[5] of objectives and assigning weights to the objective functions as we saw before.

Progressive These techniques involve adding an iterative search process to the optimization procedure. This allows the user to intervene during the execution of the MOEA in order to enhance its effectiveness and efficiency.

A posteriori Ultimately, these techniques aim to identify the entire Pareto front. If only a portion of the front is discovered, it is important that the solutions are as diverse as possible and not concentrated in a limited area of the front.

As you may have notice, most *a posteriori* techniques are Pareto-based since the try to find uniformly distributed solutions around Pareto front, deriving the final decision to the user for choosing a solution. Decomposition-based approaches are related to *a priory* is the importance associated to each objective is set by the user, i.e., *a priori* or is also compute by the algorithm, i.e., *a posteriori*. However, most of the time in decomposition-based approaches is the algorithm how compute the weights since it suffers from fewer drawbacks.

In this document, the contents are structured as follows: Section 2 present an introduction to the state-of-the-art MOEAs with a special focus of the most relevant algorithm of each approach based on [ED18] taxonomy, Section 3 include a review of the different techniques showed in Section 2 where pros and cons are discussed. Finally, Section 4 brings some conclusions about the state-of-the-art of MOEAs and its future directions.

2 State-of-the-art

The first use of heuristic algorithms such as MOEAs dates to the introduction of Vector Evaluated Genetic Algorithm (VEGA) [Sch85] in the literature. VEGA is a decomposition-based approach, where a problem with n objectives is decomposed into n subproblems and the population is also split into n subpopulations. Hence, each decomposed subproblem is addressed only by the corresponding subpopulation. The biggest drawback of VEGA is that it leads the entire population to converge to only one optimum due to using a selection mechanism that often relies on best solutions. The lack of proper selection scheme in VEGA led to the emergence of the Pareto concept [Gol13]. MOEA\D [LZ09] is another decomposition-based approach that aims to minimize the maximum margin between each objective and its corresponding reference value. MOEA\D is a state-of-the-art decomposition-based method, that succeeded a couple of preceding algorithms based on the idea of combining decomposition, scalarisation and local search. In contrast to these early algorithm such as VEGA, in MOEA\D one population with interacting neighborhood individuals is applied, which reduces the complexity of the algorithm. MOEA\D evolves a population of individuals, each individual $\vec{x}_i \in P$ being associated with a weight vector $\vec{\alpha}_i$. The i-th subproblem is defined by a scalalization function[6] $g(\vec{x}|\vec{\alpha}_i)$. The main idea is that in the creation of a new candidate solution for the i-th individual the neighbors of this individual are considered. The main drawback of MOEA\D is that its performance is directly proportional to the decomposition method employed.

Contrary to these approaches, a general indicator-based evolutionary algorithm (IBEA) [ZK04] handles a MOOP in a single population. In the calculation of fitness value, it does not consider the objective directly, but rather uses a particular performance metric obtained from the objective and aims to increase the quality of solutions with respect to the chosen metric. The main idea is to formalize preferences in terms of continuous generalizations of the dominance relation, which leads to a simple algorithmic concept. Thereby, IBEA not only allows adaptation to arbitrary preference information and optimization scenarios, but also does not need any diversity preservation techniques. Furthermore, the outcome of an MOEA is defined as a set of incomparable solutions, i.e., no solutions dominates any other solution in the feasible region. Such

[5]The main disadvantage of this method is the fact that it does not explore the Pareto front uniformly, while it is in fact simple and efficient.

[6]Typically, with Chebychev scalarization, but other scalarization such as linear weighting are also possible.

a set will also be denoted as Pareto set approximation. A quality indicator $I : \mathcal{P}_1 \times ... \times \mathcal{P}_k \rightarrow \mathbb{R}$ is a function that maps k Pareto set approximations to a real number; most common are binary quality indicators. Binary quality indicators can be used to compare the quality of two Pareto set approximations relatively to each other. A common example is the binary additive ϵ-indicator I_{ϵ^+} which gives the minimum distance by which a Pareto set approximation needs to or can be translated in each dimension in objective space such that another approximation is weakly dominated[7] [Zit+03]

$$I_{\epsilon^+}(\mathcal{P}_1, \mathcal{P}_2) = \min_{\epsilon} \left\{ \forall \vec{x}_2 \in \mathcal{P}_2, \exists \vec{x}_1 \in \mathcal{P}_1 : f_i(\vec{x}_1) \leq \epsilon + f_i(\vec{x}_2), \forall i = 1, ..., n \right\} \tag{4}$$

for any two approximation sets $\mathcal{P}_1, \mathcal{P}_2 \in \Omega$. Please see Figure 1.

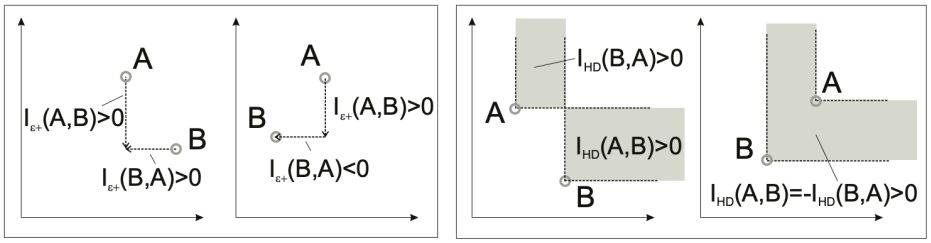

Figure 1: Illustration of the two binary quality indicators used in Zitzler and Künzli [ZK04] where A and B contain one solution vector each (left: I_{ϵ^+}-indicator; right: I_{HD}-indicator[3]). It corresponds to Figure 1 from Zitzler and Künzli [ZK04]

In IBEA, while parent and offspring individuals are constructing the next population, the poor solutions regarding the chosen metric are eliminated to maintain the size of the population. However, the use of a single-metric could lead to a poor performance with respect to other metrics.

Similar to IBEA, the Strength Pareto Evolutionary Algorithm[8] (SPEA-2) [ZLT05] does not make direct use of the approximate Pareto front set for fitness evaluation either, but instead uses a fitness value calculated based on the Pareto domination count of a solution and its distance to other solutions. One of the main concerns about SPEA-2 is that these two criteria used in fitness evaluation are treated evenly, which enables solutions that are completely dominated but farthest away from others to be kept in the population. NSGA-II [Deb+02] is the most popular Pareto-based algorithm in the literature and is based not only on the Pareto dominance but also on distances of solutions in the objective space. NGSA-II is considered a typical evolutionary MOOP [ED18]. Depending on the Pareto domination, NSGA-II splits the population into several fronts in which a distance, namely crowding distance, between the solutions is calculated. The algorithm proposed to compute the crowding distance is shown in Algorithm 1, while a visual representation for $n = 2$ is in Figure 2.

Algorithm 1 Crowding-distance assignment of non-dominated (Pareto) front $\mathcal{P}$

$l \leftarrow |\mathcal{P}|$ ▷ Count number of solution in $\mathcal{P}$
for all $\vec{x} \in \mathcal{P} = \{\vec{x}^1, ..., \vec{x}^l\}$ **do**
  $d(\vec{x}) \leftarrow 0$ ▷ Initialize distance
end for
for all objective $i = 1, ..., n$ **do**
  $\mathcal{P} \leftarrow$ sort $\mathcal{P}$ based on i-th values ▷ Sort using each objective value
  $d(\vec{x}^1), d(\vec{x}^l) \leftarrow +\infty$ ▷ Such that boundary points are always selected
  for $j = 2$ to $(l-1)$ **do** ▷ Update all other points
    $d(\vec{x}^j) \leftarrow d(\vec{x}^j) + \frac{x_i^{j+1} - x_i^{j-1}}{f_i^M - f_i^m}$ ▷ f_i^M and f_i^m represent maximum and minimum values, respectively
  end for
end for

[7] The "+" symbols indicates that it is an additive ϵ-indicator. Other option is

[8] Original work called SPEA was proposed by Zitzler and Thiele [ZT99], while SPEA-2 offers an improvement in strenght.

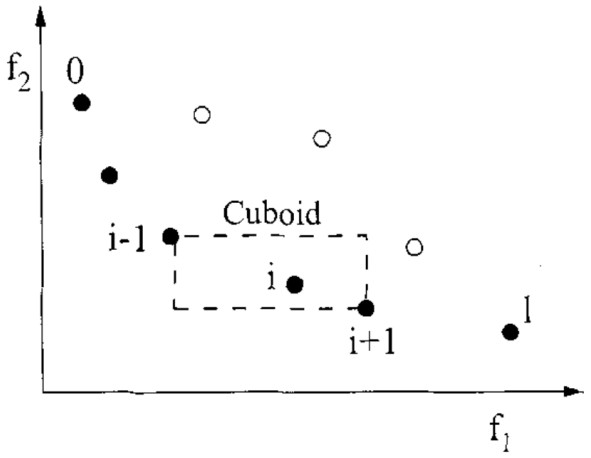

Figure 2: Crowding-distance calculation. Points marked in filled circles are solutions of the same non-dominated front. It corresponds to Figure 1 from Deb et al. [Deb+02]

One difference from SPEA2 is that non-dominated solutions that are in better fronts are allowed in NSGA-II to survive, i.e., it is an elitist approach in terms of Eiben and Smith [ES03] aproaches classification. Notice that if the size of next front exceeds the size of the population, the distance plays a key role in determining the fittest individuals to remain in the population.

NGSA-III [DJ14] is an improved version of NSGA-II which is proposed to primarily solve many-objective optimization problems[1]. The main difference is located in the selection phase. Unlike to the crowding distance, NSGA-III relies on a set of well-spread reference points to keep the population as diverse as possible throughout the optimization.

The multiobjective GDE3 [KL05] is very similar to NSGA-II, it employs Pareto front and crowding-distance-based survival scheme to keep a more diverse population in the objective space. Another algorithm similar to NSGA-II is MOCell [Neb+07], however, it applies basic genetic operators of GA to breed offspring in only a close neighborhood. This neighborhood is based on a predefined grid topology[9].

Multiobjective genetic algorithms (MOGAs) such as previously described NSGA-II, MOEA/D, SPEA2 have shown good performance in many engineering optimization problems. Nevertheless, in some concrete problems [Wan+23], these algorithms tend to converge on local optima rather than global optima. To deal with this challenge, some authors[WCJ20; Kra+21; Ede+20] have proposed an enhanced MOGA, neural network-based MOGA (NBMOGA). In this method, the data produced with the standard MOGA are used to train a neural network. The neural network is fast to produce a large pool of objective function estimates, with sufficiently high accuracy. A subset of the most competitive estimates is selected to form a population (matching MOGA population size), which is then evaluated with the MOGA evaluator. NBMOGA promises a faster convergence and a higher degree of diversity than that available with the standard MOGA. However, this kind of algorithms consisting on the hybridization of EAs with other technique, constitute a broad area, namely memetic algorithms (MA), and are exposed here just as a common practice to increase MOGAs performance.

Finally, here we introduce state-of-the-art algorithms where quantum phenomena such as superposition, entanglement, and interference are used as a source of inspiration to develop novel metaheuristics called quantum metaheuristics (QM). If their target system is a classical computer, they are called quantum-inspired metaheuristics[10] (QIMs), particularly those who draw inspiration from EA are Quantum-inspired evolutionary algorithms (QIEAs) [Zha11]. The rationale behind this is that quantum computers offer an

[9] In cellular GA (cEA), individuals cannot mate arbitrarily, but every one interacts with its closer neighbors on which a basic EA is applied

[10] Many of these quantum-inspired algorithms were thought to only operate on classical computers, such as those algorithms that use qubits' floating-point numerical representation. On the other hand, some other quantum-inspired algorithms were designed following the quantum mathematical representation that describes qubits and quantum operators capable of running in quantum simulators, which are homologous to the inner workings of real quantum devices. These kinds of metaheuristics may be closer to a future translation of a quantum computer.

exponential speed-up for complex problems when they take full advantage of quantum phenomena, and since it is widely known that quantum technology is advancing by leaps and bounds, it is reasonable to include then into this section. Like a quantum mechanical system, a quantum-inspired system can be regarded as a probabilistic system, in which the probabilities related to each state are utilized to describe the behavior of the system. QIEAs use quantum-inspired bits (Q-bit), quantum-inspired gates (Q-gates) and observation processes to specify their structure and steps. More specifically, Q-bits are applied to represent genotype individuals (Q-bit representation); Q-gates are employed to operate on Q-bits to generate offspring; and the genotypes and phenotypes are linked by a probabilistic observation process. In quantum mechanical systems, the act of observation causes a quantum particle to take on one and only one state in the measurement basis. Similarly, a superposition state represented by Q-bits in QIEAs will become a single state in the process of observation. A QIEA can exploit the search space for a global solution with a small number of individuals, even with one element [HK02]. Olvera, Montiel, and Rubio [OMR23] present the first reported statistical analysis that compares classical multiobjective evolutionary algorithms against their quantum versions when solving continuous functions. Based on their findings, the quantum-inspired algorithms executed on a quantum simulator proved efficiency when solving multiobjective optimization problems in continuous search spaces.

3 Approaches review

In the previous sections, we gave an introduction to the field of multiobjective optimization. We covered the topics of scalarisation approaches, and optimality conditions. As solution methods, we discussed evolutionary methods where state-of-the-art techniques were introduced. Furthermore, we have taken special interest in three algorithms, namely NSGA-II, IBEA, and MOEA/D, each representing the most relevant algorithm in their respective approach. NSGA-II served as a representative of Pareto-based approaches, IBEA as an example of indicator-based approaches, and MOEA/D as an example of decomposition based approaches. These algorithms have some advantages and disadvantages which we will uncover here.

Pareto-based approaches adhere to a direct and simple design principle rooted in Pareto dominance and the preservation of diversity, often utilizing metrics like crowding distance. These algorithms typically necessitate only a small number of parameters, remaining robust even when dealing with a higher count of objective functions. Nevertheless, ensuring and quantifying convergence, as well as achieving uniformly spaced solutions, can pose challenges.

Indicator-based approaches, on the other hand, employ indicators to gauge the performance of an approximation set, guiding the optimization process. These methods offer the advantage of real-time convergence assessment and the potential for more viable theoretical analysis. However, as the number of dimensions increases, their computational demands tend to escalate rapidly, and the arrangement of points in approximation sets could be greatly influenced by factors such as the reference point or other indicator-related parameters.

Decomposition-based approaches introduce a remarkably versatile framework for algorithmic design by integrating various scalarisation methodologies. Nonetheless, a drawback emerges in the form of requiring some prior knowledge about the Pareto front within the objective space. Additionally, the number of weight vectors might proliferate exponentially in tandem with the dimensionality of the objective space, even when the Pareto front exhibits low dimensionality.

Taking into account the insights presented earlier, the selection of an appropriate approach is notably contingent upon various factors. One crucial determinant is the dimensionality of the objective space under consideration. In instances where this dimensionality is relatively low, Pareto-based approach, owing to their simplicity and minimal parameter requirements, emerges as pragmatic options. These techniques exhibit a remarkable ability to handle a diverse range of objective functions without significant obstacle. However, it is worth noting that ensuring convergence and attaining a uniformly spaced distribution of solutions might be challenging in certain scenarios. Those scenarios where an online assessment of convergence is of paramount importance, indicator-based approach offers a viable alternative. These methods possess the advantage of dynamically evaluating the performance of approximation Pareto-sets during the optimization process, allowing for a more informed decision-making approach. Nevertheless, it is essential to acknowledge that as the dimensionality of the objective space increases, these approaches tend to

incur a notable rise in computational demands. Furthermore, the arrangement of solutions within approximation sets can be intricately influenced by parameter settings, such as the choice of reference point. Decomposition-based approach, with their adaptability to incorporate diverse scalarisation methods, introduces a versatile pathway for algorithmic design. However, their application requires a certain degree of prior knowledge regarding the position of the Pareto front within the objective space. This stipulation arises due to the inherent nature of these approaches, which necessitate a certain level of insight into the distribution of the Pareto-optimal solutions to define the weights. As a disadvantage, it is worth considering that the number of weight vectors utilized in decomposition-based methods can burgeon exponentially alongside the dimensionality of the objective space, even if the Pareto front itself exhibits lower dimensionality.

4 Conclusions

In conclusion, multiobjective optimization is a lively field of study with ongoing challenges. Evolutionary algorithms are popular approaches to generating solutions to a multiobjective optimization problems. Key areas for research include dealing with many objectives, refining theories, managing complexities, and adopting new strategies like diversity optimization. Currently, most evolutionary multiobjective optimization algorithms apply Pareto-based ranking schemes. Evolutionary algorithms such as the Non-dominated Sorting Genetic Algorithm-II (NSGA-II) or its extended version and Strength Pareto Evolutionary Algorithm 2 (SPEA-2) have become standard approaches. Looking ahead, Quantum-inspired Evolutionary Algorithms offer an exciting potential for advancing the field. The primary benefit of utilizing evolutionary algorithms for tackling multiobjective optimization problems lies in their tendency to produce collections of solutions. This facilitates the computation of an approximate representation of the complete Pareto front. Nonetheless, the main drawback of evolutionary algorithms pertains to their comparatively slower execution speed, coupled with the absence of a guarantee of Pareto optimality for the solutions they yield.

References

[Deb+02] K. Deb et al. “A fast and elitist multiobjective genetic algorithm: NSGA-II”. In: *IEEE Transactions on Evolutionary Computation* 6.2 (2002), pp. 182–197. DOI: 10.1109/4235.996017.

[DJ14] Kalyanmoy Deb and Himanshu Jain. “An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints”. In: *IEEE Transactions on Evolutionary Computation* 18.4 (2014), pp. 577–601. DOI: 10.1109/TEVC.2013.2281535.

[DK01] Kalyanmoy Deb and Deb Kalyanmoy. *Multi-Objective Optimization Using Evolutionary Algorithms*. USA: John Wiley & Sons, Inc., 2001. ISBN: 047187339X.

[ED18] Michael T. M. Emmerich and AndréH. Deutz. “A tutorial on multiobjective optimization: fundamentals and evolutionary methods”. In: *Natural Computing* 17.3 (2018), pp. 585–609. DOI: 10.1007/s11047-018-9685-y. URL: https://doi.org/10.1007/s11047-018-9685-y.

[Ede+20] Auralee Edelen et al. “Machine learning for orders of magnitude speedup in multiobjective optimization of particle accelerator systems”. In: *Phys. Rev. Accel. Beams* 23 (4 Apr. 2020), p. 044601. DOI: 10.1103/PhysRevAccelBeams.23.044601. URL: https://link.aps.org/doi/10.1103/PhysRevAccelBeams.23.044601.

[ES03] A. E. Eiben and J. E. Smith. *Introduction to Evolutionary Computing*. Natural computing series. Berlin, Heidelberg: Springer Berlin Heidelberg Berlin, Heidelberg, 2003. ISBN: 9783662050941; 3662050943. DOI: 10.1007/978-3-662-05094-1.

[Gol13] David E Goldberg. *Genetic algorithms*. pearson education India, 2013.

[HK02] Kuk-Hyun Han and Jong-Hwan Kim. “Quantum-inspired evolutionary algorithm for a class of combinatorial optimization”. In: *IEEE Transactions on Evolutionary Computation* 6.6 (2002), pp. 580–593. DOI: 10.1109/TEVC.2002.804320.

[KL05] S. Kukkonen and J. Lampinen. “GDE3: the third evolution step of generalized differential evolution”. In: *2005 IEEE Congress on Evolutionary Computation*. Vol. 1. 2005, 443–450 Vol.1. DOI: 10.1109/CEC.2005.1554717.

[Kra+21] M. Kranj čevi ć et al. “Multiobjective optimization of the dynamic aperture using surrogate models based on artificial neural networks”. In: *Phys. Rev. Accel. Beams* 24 (1 Jan. 2021), p. 014601. DOI: 10.1103/PhysRevAccelBeams.24.014601. URL: https://link.aps.org/doi/10.1103/PhysRevAccelBeams.24.014601.

[Liu+22] Qiuyue Liu et al. “A multiobjective evolutionary algorithm based on decision variable classification for many-objective optimization”. In: *Swarm and Evolutionary Computation* 73 (2022), p. 101108. ISSN: 2210-6502. DOI: https://doi.org/10.1016/j.swevo.2022.101108. URL: https://www.sciencedirect.com/science/article/pii/S2210650222000785.

[LZ09] Hui Li and Qingfu Zhang. “Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II”. In: *IEEE Transactions on Evolutionary Computation* 13.2 (2009), pp. 284–302. DOI: 10.1109/TEVC.2008.925798.

[Neb+07] Antonio J. Nebro et al. “Design Issues in a Multiobjective Cellular Genetic Algorithm”. In: *Evolutionary Multi-Criterion Optimization*. Ed. by Shigeru Obayashi et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 126–140. ISBN: 978-3-540-70928-2.

[OMR23] Cynthia Olvera, Oscar Montiel, and Yoshio Rubio. “Quantum-inspired evolutionary algorithms on continuous space multiobjective problems”. eng. In: *Soft computing (Berlin, Germany)* 27.18 (2023), pp. 13143–13164. ISSN: 1432-7643.

[Sch85] J D Schaffer. “Some experiments in machine learning using vector evaluated genetic algorithms”. In: (Jan. 1985). URL: https://www.osti.gov/biblio/5673304.

[SG20] E.J.C. Suárez and S.F. Galán. *Fundamentos de computación evolutiva*. Marcombo, 2020. ISBN: 9788426733344.

[Wan+15] Pu Wang et al. “Convex Hull-Based Multiobjective Genetic Programming for Maximizing Receiver Operating Characteristic Performance”. In: *IEEE Transactions on Evolutionary Computation* 19.2 (2015), pp. 188–200. DOI: 10.1109/TEVC.2014.2305671.

[Wan+23] Peilin Wang et al. “Combining multi-objective genetic algorithm and neural network dynamically for the complex optimization problems in physics”. In: *Scientific Reports* 13.1 (2023), p. 880. DOI: 10.1038/s41598-023-27478-7. URL: https://doi.org/10.1038/s41598-023-27478-7.

[WCJ20] Jinyu Wan, Paul Chu, and Yi Jiao. “Neural network-based multiobjective optimization algorithm for nonlinear beam dynamics”. In: *Phys. Rev. Accel. Beams* 23 (8 Aug. 2020), p. 081601. DOI: 10.1103/PhysRevAccelBeams.23.081601. URL: https://link.aps.org/doi/10.1103/PhysRevAccelBeams.23.081601.

[WZY23] Xujian Wang, Fenggan Zhang, and Minli Yao. “A many-objective evolutionary algorithm with estimating the convexity-concavity of Pareto fronts and clustering”. eng. In: *Information sciences* 644 (2023), p. 119289. ISSN: 0020-0255.

[Zha11] Gexiang Zhang. “Quantum-inspired evolutionary algorithms: a survey and empirical study”. In: *Journal of Heuristics* 17.3 (2011), pp. 303–351. DOI: 10.1007/s10732-010-9136-0. URL: https://doi.org/10.1007/s10732-010-9136-0.

[Zho+11] Aimin Zhou et al. “Multiobjective evolutionary algorithms: A survey of the state of the art”. In: *Swarm and Evolutionary Computation* 1.1 (2011), pp. 32–49. ISSN: 2210-6502. DOI: https://doi.org/10.1016/j.swevo.2011.03.001. URL: https://www.sciencedirect.com/science/article/pii/S2210650211000058.

[Zit+03] E. Zitzler et al. “Performance assessment of multiobjective optimizers: an analysis and review”. In: *IEEE Transactions on Evolutionary Computation* 7.2 (2003), pp. 117–132. DOI: 10.1109/TEVC.2003.810758.

[ZK04] Eckart Zitzler and Simon Künzli. “Indicator-Based Selection in Multiobjective Search”. In: *Parallel Problem Solving from Nature - PPSN VIII*. Ed. by Xin Yao et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 832–842. ISBN: 978-3-540-30217-9.

[ZLT05] Eckart Zitzler, Marco Laumanns, and Lothar Thiele. *SPEA2: Improving the strength pareto evolutionary algorithm*. en. Report. Zurich, 2001-05. DOI: 10.3929/ethz-a-004284029.

[ZT99] E. Zitzler and L. Thiele. “Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach”. In: *IEEE Transactions on Evolutionary Computation* 3.4 (1999), pp. 257–271. DOI: 10.1109/4235.797969.