DOI: https://doi.org/10.33103/uot.ijccce.23.4.5

Analytical and Comparative Study for Optimization Problems

Sanaa Ali Jabber[1], Soukaena H hashem[2] ,Shatha H Jafer[3]
[1]Faculty of Administration and Economics, AL-Muthanna University, Iraq
[2,3]Department of Computer Science, University of Technology, Baghdad, Iraq
[1]sana.ali@mu.edu.iq, [2]Soukaena.h.hashem@uotechnology.edu.iq, [3]Shatha.h.jafer@uotechnology.edu.iq

Abstract— *Finding an optimal solution to some problem, like minimizing and maximizing the objective function, is the goal of Single-Objective Optimization (SOP). Real-world problems, on the other hand, are more complicated and involve a wider range of objectives, several objectives should be maximized in such problems. No single solution could be enhanced in all objectives without deteriorating at least one other goal, which is the definition of Pareto-optimality. Understanding the idea of Multi-Objective Optimization (MOP) is thus necessary to find the optimum solution. Multi-objective evolutionary algorithm (MOEA) are made to simultaneously assess many objectives and find Pareto-optimal solutions, MOEA can resolve multi-objective and single-objective optimization problems.*

This paper aims to introduce a survey study for optimization problem solutions by comparing techniques, advantages, and disadvantages of SOP and MOP with metaheuristics and evolutionary algorithms. From this study, we conduct that the efficiency of MOP lies in the present more than one SOP, but it takes a longer time to process and train and is not suitable for all applications, While SOP is faster and more useful in stock and profit maximization applications. And the posterior techniques are considered the dominant approach to solving multi-objective problems by the use of the field of metaheuristics.

Index Terms— *Multi-Objective Evolutionary Algorithm, Multi-Objective Optimization, Optimization problem, Objective Function, Single-Objective Optimization*.

I. INTRODUCTION

Nature has been utilizing evolution to solve difficult problems for a long time, Thus, drawing inspiration from nature for various difficult problems makes sense. When evolutionary notions in nature have been reproduced in the computers to address the subject of optimization problems in 1977, Holland made a ground-breaking suggestion in optimization [1][2]. There are several challenges with solving optimization problems, the properties of optimization problems vary and are not all the same. Uncertainty, dynamicity, multiple objectives, constraints, and various objectives are a few of such challenges or characteristics [3][4]. The position of optimum global shifts with time in dynamic problems. In order to follow changes and avoid losing the global optimum, a heuristic must be equipped with appropriate operators [5][6].

Each one of the components of real problems is subject to a variation of uncertainty. A heuristic must be able to locate fault-tolerant, robust solutions for dealing with this. Another challenge of an actual problem is constraints, which limit the search space. They categorize solutions as either infeasible or feasible. In order to eliminate impractical solutions throughout optimization and ultimately discover the best practicable solution, heuristics should be equipped with appropriate operators. Through the optimization process, the best

Received 05/April/2023; Accepted 27/April/2023
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.23.4.5

solution or value could be discovered [7][8]. Multi-objectives or one objective may be used and a minimum or maximum value in the optimization problems. Engineering, mathematics, economics, social sciences, aviation, agriculture, and the automobile industry are just a few of the fields where this problem can be encountered daily [9][10][11][12]. The SOP, MOP, and MOE will be examined, contrasted, and presented in this study, along with other works that have utilized such topics.

The study is divided into five components. Related work is covered in the section II, the theoretical foundation in section III, discussion and analysis are covered in the section IV, and the recommendations and conclusion are covered in the section V.

II. RELATED WORK

This section shows previous and related Works on Optimization Problems and Multi-objective evolutionary algorithm (MOEA) watch capable of resolving both multi-objective and single-objective optimization problems. Table I summarize the work of some researchers who have employed MOEA using metaheuristics to solve various life problems.

TABLE I. SUMMARIZE THE RELATED WORKS WHO HAVE EMPLOYED MOEA USING METAHEURISTICS TO SOLVE VARIOUS PROBLEMS.

Ref	Authors	Methods	Techniques	Comment
[13]	S. Mirjalili. etal.	Grasshopper optimization	Posteriori-metaheuristics	utilize the model to approximate the global optimum in a space with a single objective. The algorithm is then modified to incorporate an archive and target selection approach to estimate Pareto optimum front for MOPs.
[14]	K. B. Bey et al.	genetic algorithm some	The Min-Min heuristic	The optimal scheduling of numerous resources and tasks is a challenge when developing apps in cloud settings.
[15]	L. T. Rasheed	Ant Colony Optimization (ACO) and particle swarm optimization (PSO)	Priori- metaheuristics	presents the design of an optimal Linear Quadratic Regulator (LQR) controller for position control of a permanent magnet DC (PMDC) motor Ant Colony control and particle swarm control algorithms have been utilized to set the optimal elements of the weighting matrices subjected to a proposed cost function.
[16]	S. Mirjalili & Dong	the NSGA and NSGAII, the multi-objective versions of GA,	Posteriori-metaheuristics	The elitism, computational cost, and requirement to provide the sharing parameter were all improved over the previous version of nondominated sorting genetic algorithm II NSGAII. The NSGAII uses a quick, non-dominated sorting algorithm, an exclusive preservation mechanism, and a brand-new, parameter-free operator termed niching.
[17]	R. Enkhbat et at	weighted sum method	Posteriori-metaheuristics	apply the multi-objective optimization approach to Malfatti's problem
[18]	S.S. Jasim et al	Levy Flight- Chaotic Chen mapping on Wolf Pack Algorithm in Neural Network	Priori- metaheuristics	proposed a novel algorithm efficiently exploits the search regions to detect driving sleepiness and balance the exploration and exploitation operators, which are considered implied features of any stochastic search algorithm.

Received 05/April/2023; Accepted 27/April/2023
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.23.4.5

[19]	R. Stewart, et al.	Multi-agent blackboard system optimization algorithm (MABS)	Priori- metaheuristics	proposed a multi-objective optimization algorithm based on a multi-agent blackboard system (MABS). The MABS framework allows multiple agents to read and write relevant optimization problem data to a central blackboard agent. Agents can search the design space at random, use previously discovered solutions to explore local optima, or update and trim the Pareto front.
[20]	F. Wei. et al.	particle swarm optimization	Interactive-metaheuristics	provides a potential application to multi-objective PAS location problem
[21]	Ye Tian et al.	DOP algorithms	tri-objective OL-DOP framework	for reflecting the properties of real-world OL-DOPs, the study suggest a benchmark generator for multi-objective and single-objective online dynamic optimization problems (OL-DOPs).
[22]	P. Aspar. Stewart et al.	SOMOGSA+NM, which hybridizes the sophisticated multi-objective MOGSA solver with Nelder–Mead local search	noiseless BBOB benchmark set.	Demonstrates the value of artificially introducing a second objective to convert multimodal single-objective problems into their multi-objective counterparts.

III. THEORETICAL BACKGROUND

Problems with constraints make up almost all real-world problems. The overall structure of optimization problems is presented in the following section.

A. Objective Function and optimization problem

The objective function can be described as a mathematical representation of an aspect that is being evaluated and should be maximized (or minimized) [23]. Creating an objective function is the initial step in creating a process for solving either the inverse or the optimization problems. Mathematically, the objective function is expressed as follows:

$$S = S(P);\ P = \{P1, P2, ..., PN\} \quad (1)$$

Where P1, P 2,..., PN represent variables of issue under consideration that may be altered to find the minimal value of function S. Often, a mathematical or physical model could be used to express the relation between P and S. Rather than simply having a possible solution satisfy constraints or not, we commonly have preference relation compared to the possible solutions, and we want the optimal possible solution accordance with the preference. In a few cases, on the other hand, this relation is impossible or impractical, and variation of S concerning P should be found via experimentations. Most of the time, it is preferred to reduce some of the errors [24][7].

The optimization problem may be expressed as follows:

- a group of variables, each one with a related domain;
- an objective function mapping total assignments to the numbers;
- finding a total assignment that maximizes or minimizes objective function has been known as an optimality criterion.

Finding a total assignment that satisfies the optimality criterion is the goal. For concreteness, we presume that the objective function minimization is the optimality criterion [25]. An optimization problem with hard constraints defining the range of potential variable assignments is called a constrained optimization problem. The best assignment that complies with the hard requirements. On optimization, there is a wealth of literature. For

Received 05/April/2023; Accepted 27/April/2023
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.23.4.5

various types of constrained optimization problems, there are numerous methods. For instance, linear programming can be defined as a type of optimization in which variables are real valued, objective function represents a linear function of variables, and constrained constraints represent linear inequalities [1].

B. single-objective optimization

Specific techniques are only relevant to certain kinds of functions, such as uni-modal functions, with only one maximum (or minimum) within the parameter range under study. The function need not be continuous as a result [22]. An single-objective optimization (SOP) problem (SOPP) has an objective function of ($f(\vec{x})$) that should be maximized or minimized and a number of the constraints ($g(\vec{x})$). Eq. (2) exhibits SOPP formula in a generalized form [26].

$$\begin{cases} \text{minimize } f(\vec{x}) \\ s.t\ gj(\vec{x}) \geq 0\ (j = 1; \ldots\ldots; m) \\ \vec{x} \in X \subset R^n \end{cases} \quad (2)$$

The first two functions in *Fig. 1* are unimodal, as can be seen. The third function is uni-modal in an interval of $0 < P < 3\frac{\omega}{2}$ and the fourth function is multimodal. To discover the place of the minimum or maximum for uni-modal functions, it's incredibly simple to exclude parts of the domain that are being studied. Consider the first function in *Fig. 1* as an illustration. If we're attempting to locate the maximal value of a function and we are aware that S (P = 1) is smaller than S (P = 2), one could instantly rule out the part to the left of P = 1 because this function increases in its value monotonically. And that isn't true for the case of the multimodal functions, which had been shown as the fourth function in *Fig. 1*.

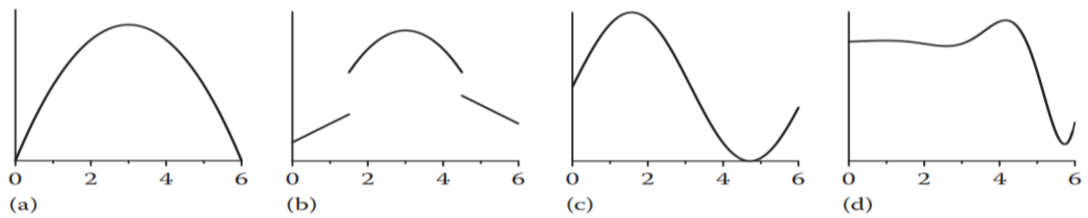

FIG. 1. EXAMPLES OF THE FUNCTIONS S (I.E. ORDINATE) OF ONE DESIGN VARIABLE P (ABSCISSA).

Avoiding local optima, maintaining sufficient diversity, and assisting the algorithm in identifying suitable building blocks which could be later put together by crossover are all significant problems in SOP [24].

C. Multi-objective Optimization Problems

Naturally, the majority of optimization problems have many goals to achieve ((which are typically at odds with one another), yet for the purpose of simplifying their solution, they're handled as if they had just one (the rest of the objectives are usually treated as if they were constraints). "multi-objective" or "vector" optimization problems refer to problems with many objectives. They were initially researched in the area of economics. But engineers and scientists quickly discovered that these problems naturally exist in all fields of knowledge [27].

Basic definitions

Definition1 (Global minimum). Considering the function $f: \Omega \subseteq R^n \rightarrow R$, $\Omega \neq \emptyset$, for $\vec{x} \in \Omega$ value $f^* \triangleq f(\vec{x}^*) > -\infty$ would be referred to as global minimum only in the case where:

Received 05/April/2023; Accepted 27/April/2023
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.23.4.5

$$\forall \vec{x} \in \Omega: f(\vec{x}^*) \leq f(\vec{x}). \quad (3)$$

Then, $\vec{x}^*$ Represents solution(s) of global minimum, f represents an objective function and set Ω represents a feasible region (Ω ∈ S), where S denotes the entire search space.

Definition2 (General MOP (i.e., MOP problem). Find vector $\vec{x}^* = [x^*_1, x^*_2, \ldots, x^*_n]^T$ satisfying m constraints of inequality:

$$gi(\vec{x}) \geq 0 \ i = 1, 2, \ldots, m \quad (4)$$

ρ constraints of equality

$$hi(\vec{x}) = 0 \ i = 1, 2, \ldots, p \quad (5)$$

and will lead to optimization of a vector function

$$\vec{f}(\vec{x}) = f1(\vec{x}), f2(\vec{x}), \ldots, fk(\vec{x})]T \quad (6)$$

where $\vec{x}$ = [x1, x2,..., xn]T represents a decision variable vector[27].

The idea of "optimal" changes in the case where there are multiple objective functions because, unlike in global optimization, the goal of MOPs is identifying good compromises (or "trade-offs") instead of the solution. The definition of "optimum" that is most frequently used was first put forth by Vilfredo Pareto and later generalized by Francis Y sidro Edgeworth[28]. Even though a few publications refer to this idea as Edge worth-Pareto optimal, using the term Pareto optimum is typically recommended. The formal definition follows.

Definition 3 (Pareto optimality). A point $\vec{x}^* \in \Omega$ would be referred to as Pareto optimal if for each $\vec{x} \in \Omega$ and I = {1, 2,..., k} either,

$$\forall i \in I \ (fi(\vec{x}) = fi(\vec{x}^*)) \quad (7)$$

or there is a minimum of one i ∈ I such that

$$fi(\vec{x}) > fi(\vec{x}^*) \quad (8)$$

In other words, that definition states that x is Pareto optimal when an infeasible vector x exists that might reduce certain criteria without simultaneously increasing at least a single other criterion. Unless otherwise stated, "Pareto optimal" is taken to mean about complete choice variable space.

Other significant definitions that are related to the Pareto optimality include:

Definition4 (Pareto dominance). A vector $\vec{u} = (u_1, \ldots, u_k)$ would be considered to dominate $\vec{v} = (v_1, \ldots, v_k)$ (denoted by $\vec{u} \preceq \vec{v}$) only in the case where u is partially less than v; i.e.,

$\forall i \in \{1, \ldots, k\}, ui \leq vi \land \exists i \in \{1, \ldots, k\}: ui < vi.$

Definition5 (Pareto optimal set). For some certain MOP $\vec{f}(x)$, a Pareto optimal set ($\mathcal{P}^*$) would be expressed as:

$$\mathcal{P}* := \{x \in \Omega \ \neg\exists x' \in \Omega \ \vec{f}(x') \preceq \vec{f}(x)\}. \quad (9)$$

Definition6 (Pareto front)[29]. For some certain MOP $\vec{f}$ (x) and Pareto optimal set $\mathcal{P}*$, a Pareto front ($\mathcal{P}$F∗) would be represented as:

$$\mathcal{P}F* := \{\vec{u} = \vec{f} = (f1(x), \ldots, fk(x)) \mid x \in \mathcal{P}*\}. \quad (10)$$

Pareto front of set $\mathcal{P}*$ is is given as in Equation (10) and can be seen in *Fig. 2*.

Received 05/April/2023; Accepted 27/April/2023

DOI: https://doi.org/10.33103/uot.ijccce.23.4.5

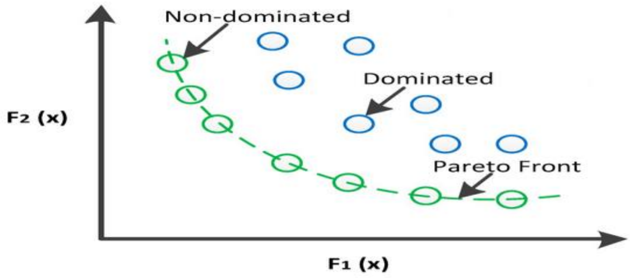

FIG. 2. DOMINATED, NON-DOMINATED AND PARETO-FRONT SOLUTION SET.

Finding an analytical expression of a surface or line which includes such points is typically difficult and, in many instances, impossible. To create the Pareto front, one would typically compute points (Ω) and their corresponding f (Ω). It is after that easy to identify non-dominated points and create a Pareto front once there are enough of these. [9][30] Since it can't be dominated by a solution x, this solution is known as Pareto optimal solution. A group of best non-dominated solutions exists for any problem. This set is taken into account as a MOP solution. As a result, a set that has been referred to as Pareto optimal front stores Pareto optimal solutions' projection in the objective space [13][31].

D. Multi-Objective Optimization Techniques

Many different approaches to solving MOP problems have been developed throughout the years thanks to the efforts of a sizable number of operational researchers. They apply stochastic programming or metaheuristics when they have really complex (particularly non-convex) problems. Multiple configurations could be examined with various search variable values when employing a metaheuristic optimization algorithm, which is another benefit. This is especially helpful for assessing how well various objectives perform in a multi-scenario study. A posteriori [25], a priori, and interactive [32] are the three primary methods for solving multi-objective problems utilizing metaheuristics [33]. The first solution that comes to mind is a brute-force search. It is neither a practical nor appropriate solution because it necessitates many function evaluations and intensive computer resources. A few problems could be transformed into SO problems because the amount of priority of objectives has been determined by a designer's desire or their conditions. A priori preference specification refers to such approaches, which include weighted min-max [33], weighted sum [3], goal programming [34], and lexicographic approaches [35].

More than one objective would be combined into a single objective in the first approach. As a result, MOP has been reduced to the following single-objective problem [13]:

$$\text{Minimizing: } F(\vec{x}) = w_1 f_1(\vec{x}) + w_2 f_2(\vec{x}) + \ldots + w_0 f_0(\vec{x}) \quad (11)$$

$$\text{Subject to: } g_i(\vec{x}) \geq 0,\ i = 1, 2, \ldots, m \quad (12)$$

$$h_i(\vec{x}) = 0,\ i = 1, 2, 3, 4, \ldots, p \quad (13)$$

$$L_i \leq x_i \leq U_i,\ i = 1, 2, 3, 4, \ldots, n \quad (14)$$

Here, w1, w2,w3,,..., w0 represent weight values of the objectives, n represents number of the variables, o represents number of the objective functions, m denotes number of the constraints of inequality, p denotes the number of constraints of equality, hi represents ith

Received 05/April/2023; Accepted 27/April/2023
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.23.4.5

constraints of equality, gi represents ith constraint of inequality, and [Li,Ui] represent boundaries of ith variable.

Since decisions are made before optimization when setting the weights, these are known as a priori approaches. A priori techniques have more drawbacks than gains. To identify the Pareto optimal front when utilizing such approaches, the designer's primary responsibility is running the algorithm several times simultaneously while varying the weight values. The a priori procedures have drawbacks. First off, not all MO problems contain preference information. Second, because the SOP solution focuses on a single problem, the designer does not understand how objectives can be traded off. Multi-objective metaheuristics' second method is a posteriori. The decision-making process takes place after optimization, as the name suggests. There is no longer any aggregation, and these techniques continue to formulate the problem as having multiple objectives. The major benefits of this class are the capability to identify a Pareto optimal solution that has been set in a single run, information sharing across Pareto optimal solutions throughout optimizations, and determining Pareto optimal front of any type. To handle multiple, frequently conflicting objectives, posteriori techniques, on the other hand, call for unique processes.

Additionally, these approaches typically have higher computing costs compared to aggregation procedures. Interactive MOP is the name of the last method discussed above. Decisions are made throughout optimization, as implied by the term. An expert preference is fetched continuously and included throughout optimization so as to obtain required Pareto optimal solutions. This technique is known as interactive optimization or "human-in-the-loop" optimization [36]. The body of research demonstrates that posteriori approaches dominate MOP. Most of the well-regarded SOP algorithms were altered to do a posteriori MOP. They all use an archive to keep the optimal Pareto optimum solutions found thus far, and they all compare solutions depending on Pareto dominance. The fundamental structure of every a posteriori approach is the same. With a group of random solutions, they start the optimization process. They attempt to enhance solutions in order to find better Pareto optimal solutions after discovering them and saving them into an archive. When a criterion has been satisfied, the process of enhancing Pareto optimal solutions has been terminated. Finding a highly accurate approximation of actual (i.e. true) Pareto optimum solutions for some certain multi-objective problem is the fundamental goal of an a posteriori multi-objective algorithm. The solutions must be distributed among all of the objectives as evenly as possible because decision-making often follows optimization results. Finding accurate Pareto optimal solutions (i.e., convergence) is difficult because it conflicts with distribution of solutions (i.e., the coverage). To solve a multi-objective problem, the MOP must successfully balance those two aspects [37][13].

E. Multi-Objective Evolutionary Algorithm

An MOEA has been initially used in the middle of the 1980s. Since then, a significant amount of study has been conducted in this field, which is currently known as evolutionary multi-objective optimization (EMO) [30]. A stochastic optimization method is the MOEA. MOEAs are utilized to determine the best Pareto solutions for particular problems, much like other optimization algorithms, yet they differ from population-based methods. Almost all current MOEAs base their behavior on dominance [38]. Except for the utilization of dominance relationships, the MOEA's optimization method is extremely comparable to evolutionary algorithms. For selecting a potentially superior solution for the generation of a hereditary population, the objective value is determined for each individual at each iteration

Received 05/April/2023; Accepted 27/April/2023
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.23.4.5

and utilized to define the relation of dominance in the population. This population could be combined with its parent populations to create populations for the following generation. Also, the presence of objective space can allow MOEA the flexibility to use some traditional support approaches, such as niching [37]. Evolutionary algorithms mostly determine the Pareto front and collection of the multi-objective problems. Obtaining Pareto set and Pareto front, which are near true Pareto set and true Pareto front (i.e., convergence), is one of the two primary objectives of the MOP algorithms [39]. Multi-objective functions come with a predefined true Pareto set and front. The other motive is to achieve the maximal well-distributed Pareto set and front (i.e., the coverage). This suggests that solutions must be evenly dispersed in the Pareto set and front rather than being concentrated close to one another. Multimodal and MOP problems are the names given to this group of problems [40]. A multimodal MOP problem with 2 Pareto sets for one Pareto front is shown in *Fig. 3*. The same color and shape are used to symbolize solutions and the accompanying objective value. The far-off solutions in the choice space might be crowded in objective space.

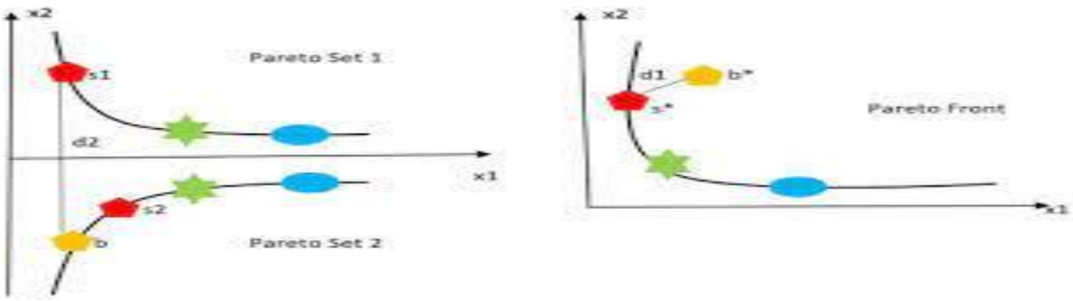

FIG. 3. ILLUSTRATES A MULTIMODAL MOO PROBLEM.

A few of the most well-known posteriori approaches have been based upon Evolutionary Algorithms (EA) [41], which includes NSGA-II [42] and Non-dominated Sorting Genetic Algorithm (NSGA)[43], [44], [45], MO Particle Swarm Optimization (MOPSO) [28], [46], and Pareto-frontier Differential Evolution (PDE) [47]. While the recently developed nature-inspired EAs have been utilized in real-life MOPs as well, the algorithms that have been mentioned above were first introduced more than ten years ago. Excellent examples of EA use in MO optimizations include the Ant Lion Optimizer (ALO) [41], Ant Colony Optimization [15] flower pollination algorithm [48], Water Cycle Algorithm (WCA) [32], Grey Wolf Optimizer (GWO) [49], Cuckoo Search Algorithm [50] and ABC Algorithm [51]. Fig. 4: Metaheuristic-based classification of optimization algorithms.

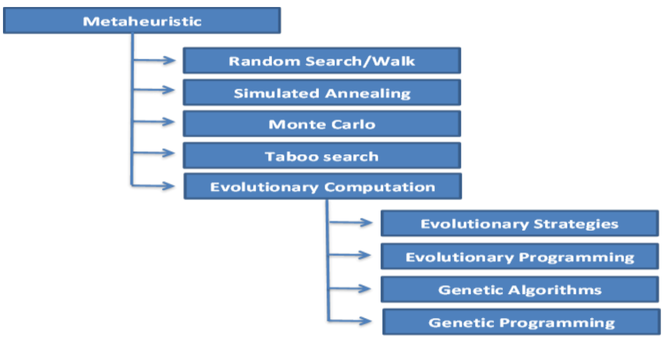

FIG. 4. CLASSIFICATION OF OPTIMIZATION BASED ON METAHEURISTICS.

Received 05/April/2023; Accepted 27/April/2023
 ISSN (Print) 1811-9212 *ISSN (Online) 2617-3352*

DOI: https://doi.org/10.33103/uot.ijccce.23.4.5

IV. ANALYSIS AND DISCUSSION

Through our study of the three topics, SOP, MOP, and MOE algorithms, we can compare these techniques to clarify the possibility of each technique, as shown in Table II. In summary, SOP involves optimizing a single objective function, while MOP involves optimizing multiple objective functions simultaneously. MOEAs are an optimization algorithm that can be utilized for finding Pareto optimal solutions for MOP problems. In Table III, we compare techniques used to solve MOP using metaheuristics.

TABLE II. COMPARISON AMONG SOP, MOP, MOEA

Optimization problems	Definition	Goal	Example
Single objective (SOP)	Involve optimizing a single objective function	find the solution that optimizes this objective function.	finding the shortest route between two points
Multi-objective (MOP)	Involve optimizing multiple objective functions simultaneously	find a set of solutions that represent the trade-offs between the different objective functions.	designing an airplane that is both fuel efficient and has a low noise level.
Multi-objective evolutionary algorithms (MOEA)	A type of optimization algorithm that can be used to find Pareto optimal solutions for multi-objective optimization problems	Use principles of natural selection and genetics to search for solutions. Often used in complex optimization problems where traditional optimization methods may not be effective.	multiobjective optimization model is applied using the metaheuristics cuckoo search optimization algorithm (MCSO) to enhance the performance of a cloud system with limited computing resources while minimizing the time and cost.

TABLE III. TECHNIQUES FOR SOLVING MULTI-OBJECTIVE PROBLEMS USING METAHEURISTICS

Techniques	advantage	disadvantage	Comment
priori	aggregation of objectives allows single-objective optimizers to effectively find Pareto optimal solutions.	Need to run the algorithm multiple times while changing weight values to find Pareto optimal front.	Decision-making is done before the optimization within determining weights. The disadvantages of *a priori* approaches outweigh their benefits
Posteriori	the capacity to identify Pareto optimal front of any type, exchange information amongst Pareto ideal solutions throughout optimization, and obtain the optimum solution set in one run.	Require certain mechanisms for addressing multiple and usually conflicting goals. Additionally, those methods' computational costs are typically higher than those of the aggregation methods.	the dominant approaches of MOP
interactive	by directly incorporating the human decision-maker in the search process, the user can learn from solutions as they're developed, refining their preferences and restricting the search to the most relevant solution space areas.	It is complicated, and it costs more	This is a common response to the limitations of a priori and a posteriori approaches.

V. CONCLUSIONS AND RECOMMENDATIONS

A few of the most fascinating and difficult problems in the area of computer sciences and mathematics are optimization problems. Following the thorough research, we may draw the next conclusions:

Received 05/April/2023; Accepted 27/April/2023
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.23.4.5

- The application seeking optimality is associated with the distinction between MOP and SOP. According to the study, direct applications with a single goal prefer to utilize SOP to save time and consume less CPU power, whereas MOP is more sophisticated and utilizes MOP algorithms which could simultaneously maximize multiple objectives that are at odds with one another. To determine the best compromise solution amongst multiple objectives, such algorithms might produce a set of the trade-off solutions which had been referred to as Pareto front.
- Various optimization algorithms are available, such as derivative-free algorithms (such as PSO and evolutionary algorithms) and gradient-based algorithms (like gradient descent and stochastic gradient ratio), given that this work studied evolutionary algorithms.
- Finding Pareto-optimal solutions for MOP and SOP could be done using MOEAs, an optimization algorithm.

We could offer guidance to optimization researchers after researching and examining optimization problems. An initial step is to specify the objectives that one hopes to accomplish. Any aim could be crucial, in which case we advise utilizing MOP. Since the goals could differ, it is important to concentrate on the higher goal. In this situation, it is advised to employ SOP rather than Pareto solutions due to their energy, time, and storage requirements for system training. The application determines which algorithm should be used. For instance, a financial analytics program could optimize a mutual fund's portfolio. The analyst's goal may be to increase return on investment while lowering portfolio risk and guaranteeing that the portfolio is well-diversified. In this case, when deciding which assets to include in the portfolio in this scenario, the analyst might have to consider several conflicting objectives. Non-dominated sorting genetic algorithms (NSGA), Multi-objective evolutionary algorithms (MOEAs), Weighted product method, Weighted sum method, multi-objective linear programming, and Pareto optimization are some algorithms frequently utilized for solving multi-objective problems in financial analytics. Consider a financial analyst attempting to maximize the return on a portfolio of investments as an example of a single objective function. The analyst might search for the portfolio that maximizes the objective function, which might be the return on the portfolio. Financial analytics SOPP could be solved using a variety of algorithms, such as Simulated annealing, PSO, Genetic Algorithms, Tabu search, and Gradient Descent.

REFERENCES

[1] I. Rahimi, A. H. Gandomi, F. Chen, and E. Mezura-Montes, “A Review on Constraint Handling Techniques for Population-based Algorithms: from single-objective to multi-objective optimization,” *Arch. Comput. Methods Eng.*, no. 0123456789, 2022, doi: 10.1007/s11831-022-09859-9.

[2] C. Gambella, B. Ghaddar, and J. Naoum-Sawaya, “Optimization problems for machine learning: A survey,” *Eur. J. Oper. Res.*, vol. 290, no. 3, pp. 807–828, 2021, doi: 10.1016/j.ejor.2020.08.045.

[3] F. J. Gibson, R. M. Everson, and J. E. Fieldsend, “Guiding surrogate-assisted multi-objective optimisation with decision maker preferences,” *GECCO 2022 - Proc. 2022 Genet. Evol. Comput. Conf.*, pp. 786–795, 2022, doi: 10.1145/3512290.3528814.

[4] W. C. Yeh, W. Zhu, Y. F. Peng, and C. L. Huang, “A Hybrid Algorithm Based on Simplified Swarm Optimization for Multi-Objective Optimizing on Combined Cooling, Heating and Power System,” *Appl. Sci.*, vol. 12, no. 20, 2022, doi: 10.3390/app122010595.

[5] A. Slowik and H. Kwasnicka, “Evolutionary algorithms and their applications to engineering problems,” *Neural Comput. Appl.*, vol. 32, no. 16, pp. 12363–12379, 2020, doi: 10.1007/s00521-020-04832-8.

[6] D. Yazdani, R. Cheng, D. Yazdani, J. Branke, Y. Jin, and X. Yao, “A Survey of Evolutionary Continuous Dynamic Optimization over Two Decades-Part B,” *IEEE Trans. Evol. Comput.*, vol. 25, no. 4, pp. 630–650, 2021, doi: 10.1109/TEVC.2021.3060012.

Received 05/April/2023; Accepted 27/April/2023
 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352

DOI: https://doi.org/10.33103/uot.ijccce.23.4.5

[7] J. Liang *et al.*, "A Survey on Evolutionary Constrained Multi-objective Optimization," *IEEE Trans. Evol. Comput.*, vol. PP, no. January, p. 1, 2022, doi: 10.1109/TEVC.2022.3155533.

[8] J. C. Fromer and C. W. Coley, "Computer-aided multi-objective optimization in small molecule discovery," *Patterns*, vol. 4, no. 2, p. 100678, 2023, doi: 10.1016/j.patter.2023.100678.

[9] J. Bossek, P. Kerschke, and H. Trautmann, "A multi-objective perspective on performance assessment and automated selection of single-objective optimization algorithms," *Appl. Soft Comput. J.*, vol. 88, p. 105901, 2020, doi: 10.1016/j.asoc.2019.105901.

[10] A. G. Gad, *Particle Swarm Optimization Algorithm and Its Applications: A Systematic Review*, vol. 29, no. 5. Springer Netherlands, 2022. doi: 10.1007/s11831-021-09694-4.

[11] M. Dehghani *et al.*, "A spring search algorithm applied to engineering optimization problems," *Appl. Sci.*, vol. 10, no. 18, pp. 1–21, 2020, doi: 10.3390/APP10186173.

[12] H. Bayzidi, S. Talatahari, M. Saraee, and C.-P. Lamarche, "Social Network Search for Solving Engineering Optimization Problems," *Comput. Intell. Neurosci.*, vol. 2021, p. 8548639, 2021, doi: 10.1155/2021/8548639.

[13] S. Z. Mirjalili, S. Mirjalili, S. Saremi, H. Faris, and I. Aljarah, "Grasshopper optimization algorithm for multi-objective optimization problems," *Appl. Intell.*, vol. 48, no. 4, pp. 805–820, 2018, doi: 10.1007/s10489-017-1019-8.

[14] K. B. Bey, S. Bouznad, F. Benhammadi, and H. Nacer, "Improved Virus Optimization algorithm for two-objective tasks Scheduling in Cloud Environment," *Commun. Pap. 2019 Fed. Conf. Comput. Sci. Inf. Syst.*, vol. 20, pp. 109–117, 2019, doi: 10.15439/2019f63.

[15] L. T. Rasheed, "Optimal Tuning of Linear Quadratic Regulator Controller Using Ant Colony Optimization Algorithm for Position Control of a Permanent Magnet DC Motor," *Iraqi J. Comput. Commun. Control Syst. Eng.*, vol. 20 No 3, no. December 2019, pp. 29–41, 2020, doi: 10.33103/uot.ijccce.20.3.3.

[16] J. S. D. Seyedali Mirjalili, "Multi - Objective Optimization using Artificial Intelligence Techniques Authors :," *SpringerBriefs Appl. Sci. Technol.*, p. 2020, 2020, [Online]. Available: link.springer.com/book/10.1007/978-3-030-24835-2

[17] R. Enkhbat and G. Battur, "Multi-objective Optimization Approach to Malfatti's Problem," *Contrib. to Game Theory Manag.*, vol. 14, no. March, pp. 82–90, 2021, doi: 10.21638/11701/spbu31.2021.07.

[18] S. S. Jasim, A. Karim, A. Hassan, and S. Turner, "Optimizing Artificial Neural Networks Using Levy-Chaotic Mapping on Wolf Pack Optimization Algorithm for Detect Driving Sleepiness," *Iraqi J. Comput. Commun. Control Syst. Eng.*, vol. 22, no. 3, pp. 128–136, 2022, [Online]. Available: https://ijccce.uotechnology.edu.iq/pdf_176802_20a636a509a8500bede79fe4d94ada68.html

[19] R. Stewart, T. S. Palmer, and S. Bays, "An agent-based blackboard system for multi-objective optimization," *J. Comput. Des. Eng.*, vol. 9, no. 2, pp. 480–506, 2022, doi: 10.1093/jcde/qwac009.

[20] F. Wei, W. Xu, and C. Hua, "A Multi-Objective Optimization of Physical Activity Spaces," *Land*, vol. 11, no. 11, p. 1991, 2022, doi: 10.3390/land11111991.

[21] X. Xiang, Y. Tian, R. Cheng, X. Zhang, S. Yang, and Y. Jin, "A benchmark generator for online dynamic single-objective and multi-objective optimization problems," *Inf. Sci. (Ny).*, vol. 613, no. September, pp. 591–608, 2022, doi: 10.1016/j.ins.2022.09.049.

[22] P. Aspar, V. Steinhoff, L. Schäpermeier, P. Kerschke, H. Trautmann, and C. Grimme, "The objective that freed me: a multi-objective local search approach for continuous single-objective optimization," *Nat. Comput.*, vol. 0123456789, no. 2001, 2022, doi: 10.1007/s11047-022-09919-w.

[23] K. Duan, S. Fong, S. W. I. Siu, W. Song, and S. S. U. Guan, "Adaptive incremental Genetic Algorithm for task scheduling in cloud environments," *Symmetry (Basel).*, vol. 10, no. 5, pp. 1–13, 2018, doi: 10.3390/sym10050168.

[24] M. T. Jensen, "Guiding Single - Objective Optimization Using Multi - objective Methods," in *(LNCS*, vol. 2611, 2003, pp. 268–279.

[25] E. G. Talbi, M. Basseur, A. J. Nebro, and E. Alba, "Multi-objective optimization using metaheuristics: Non-standard algorithms," *Int. Trans. Oper. Res.*, vol. 19, no. 1–2, pp. 283–305, 2012, doi: 10.1111/j.1475-3995.2011.00808.x.

[26] S. Watanabe and K. Sakakibara, "Multi-objective approaches in a single-objective optimization environment," *2005 IEEE Congr. Evol. Comput. IEEE CEC 2005. Proc.*, vol. 2, no. May, pp. 1714–1721, 2005, doi: 10.1109/cec.2005.1554895.

[27] M. G. Misegiannis *et al.*, "From Cloud to Serverless : MOO in the new Cloud epoch To cite this version : HAL Id : hal-03925696 From Cloud to Serverless : MOO in the new Cloud epoch," 2023.

[28] C. A. Coello, G. T. Pulido, and M. S. Lechuga, "Handling multiple objectives with particle swarm optimization," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 256–279, Jun. 2004, doi: 10.1109/TEVC.2004.826067.

[29] Z. Liang, T. Wu, X. Ma, Z. Zhu, and S. Yang, "A Dynamic Multiobjective Evolutionary Algorithm Based on Decision Variable Classification," *IEEE Trans. Cybern.*, vol. 52, no. 3, pp. 1602–1615, 2022, doi: 10.1109/TCYB.2020.2986600.

[30] C. A. C. Coello and N. C. Cortés, "Solving multiobjective optimization problems using an artificial immune system," *Genet. Program. Evolvable Mach.*, vol. 6, no. 2, pp. 163–190, 2005, doi: 10.1007/s10710-005-6164-x.

Received 05/April/2023; Accepted 27/April/2023

DOI: https://doi.org/10.33103/uot.ijccce.23.4.5

[31] K. Mahesh, P. Nallagownden, and I. Elamvazuthi, “Advanced Pareto front non-dominated sorting multi-objective particle swarm optimization for optimal placement and sizing of distributed generation,” *Energies*, vol. 9, no. 12, 2016, doi: 10.3390/en9120982.

[32] S. Mirjalili, S. Saremi, S. M. Mirjalili, and L. dos S. Coelho, “Multi-objective grey wolf optimizer: A novel algorithm for multi-criterion optimization,” *Expert Syst. Appl.*, vol. 47, pp. 106–119, Apr. 2016, doi: 10.1016/j.eswa.2015.10.039.

[33] L. G. Hernández-Pérez and J. M. Ponce-Ortega, “Multi-objective Optimization Approach Based on Deterministic and Metaheuristic Techniques to Resource Management in Health Crisis Scenarios Under Uncertainty,” *Process Integr. Optim. Sustain.*, vol. 5, no. 3, pp. 429–443, 2021, doi: 10.1007/s41660-020-00154-3.

[34] R. W. Hanks, B. J. Lunday, and J. D. Weir, “Robust goal programming for multi-objective optimization of data-driven problems: A use case for the United States transportation command’s liner rate setting problem,” *Omega*, vol. 90, pp. 1–10, Jan. 2020, doi: 10.1016/j.omega.2018.10.013.

[35] L. Lai, L. Fiaschi, and M. Cococcioni, “Solving mixed Pareto-Lexicographic multi-objective optimization problems: The case of priority chains,” *Swarm Evol. Comput.*, vol. 55, pp. 1–17, Jun. 2020, doi: 10.1016/j.swevo.2020.100687.

[36] B. Doerr and W. Zheng, “Theoretical Analyses of Multi-Objective Evolutionary Algorithms on Multi-Modal Objectives,” *35th AAAI Conf. Artif. Intell. AAAI 2021*, vol. 14A, pp. 12293–12301, 2021, doi: 10.1609/aaai.v35i14.17459.

[37] S. Cajot, N. Schüler, M. Peter, A. Koch, and F. Maréchal, “Interactive optimization with parallel coordinates: Exploring multidimensional spaces for decision support,” *Front. ICT*, vol. 5, no. JAN, pp. 1–28, 2018, doi: 10.3389/fict.2018.00032.

[38] J. Behnamian, M. Zandieh, and S. M. T. Fatemi Ghomi, “Bi-objective parallel machines scheduling with sequence-dependent setup times using hybrid metaheuristics and weighted min–max technique,” *Soft Comput.*, vol. 15, no. 7, pp. 1313–1331, Jul. 2011, doi: 10.1007/s00500-010-0673-0.

[39] J. G. Falcón-Cardona and C. A. C. Coello, “Indicator-based Multi-objective Evolutionary Algorithms: A Comprehensive Survey,” *ACM Comput. Surv.*, vol. 53, no. 2, 2020, doi: 10.1145/3376916.

[40] G. Sun, H. Chen, and Y. Mao, “The Efficiency Analysis Model of Technological Innovation Resource Allocation of Coastal Enterprises Based on Genetic Algorithm,” *J. Coast. Res.*, vol. 103, no. SI, pp. 691–695, 2020.

[41] L. M. El Bakrawy *et al.*, “A Modified Ant Lion Optimization Method and Its Application for Instance Reduction Problem in Balanced and Imbalanced Data,” *Axioms*, vol. 11, no. 3, 2022, doi: 10.3390/axioms11030095.

[42] N. Srinivas and K. Deb, “Muiltiobjective Optimization Using Nondominated Sorting in Genetic Algorithms,” *Evol. Comput.*, vol. 2, no. 3, pp. 221–248, Sep. 1994, doi: 10.1162/evco.1994.2.3.221.

[43] M. Babor, L. Pedersen, U. Kidmose, O. Paquet-Durand, and B. Hitzmann, “Application of Non-Dominated Sorting Genetic Algorithm (NSGA-II) to Increase the Efficiency of Bakery Production: A Case Study,” *Processes*, vol. 10, no. 8, 2022, doi: 10.3390/pr10081623.

[44] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002, doi: 10.1109/4235.996017.

[45] W. Zheng, Y. Liu, and B. Doerr, “A First Mathematical Runtime Analysis of the Non-dominated Sorting Genetic Algorithm II (NSGA-II) (Hot-off-the-Press Track at GECCO 2022),” *GECCO 2022 Companion - Proc. 2022 Genet. Evol. Comput. Conf.*, pp. 53–54, 2022, doi: 10.1145/3520304.3534070.

[46] B. Cao, Y. Gu, Z. Lv, S. Yang, J. Zhao, and Y. Li, “RFID Reader Anticollision Based on Distributed Parallel Particle Swarm Optimization,” *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3099–3107, 2020.

[47] L. Shen, X. Li, and L. Gao, “An improved multi - objective firefly algorithm for energy - efficient hybrid flowshop rescheduling problem,” *J. Clean. Prod.*, vol. 385, no. 20 January 2023, p. 135738, 2023, [Online]. Available: https://doi.org/10.1016/j.jclepro.2022.135738

[48] S. Mirjalili, P. Jangir, and S. Saremi, “Multi-objective ant lion optimizer: a multi-objective optimization algorithm for solving engineering problems,” *Appl. Intell.*, 2017, doi: 10.1007/s10489-016-0825-8.

[49] I. Grey and W. Optimization, “Improved Grey Wolf Optimization Algorithm and Application,” pp. 1–19, 2022.

[50] A. N. Malti, B. Benmammar, and M. Hakem, “Task Scheduling in Cloud Computing Based on FPA Metaheuristic Algorithm,” *Iraqi J. Comput. Commun. Control Syst. Eng. (IJCCCE)*, vol. 22, No. 1, no. March, pp. 41–46, 2022, doi: 10.1109/SSD54932.2022.9955799.

[51] E. Hancer, B. Xue, M. Zhang, D. Karaboga, and B. Akay, “A multi-objective artificial bee colony approach to feature selection using fuzzy mutual information,” in *2015 IEEE Congress on Evolutionary Computation (CEC)*, May 2015, pp. 2420–2427. doi: 10.1109/CEC.2015.7257185.

Received 05/April/2023; Accepted 27/April/2023

 ISSN (Print) 1811-9212 ISSN (Online) 2617-3352