

A New Multiobjective Particle Swarm Optimization Using Local Displacement and Local Guides

Saïd Charriffaini Rawhoudine, Abdoulhafar Halassi Bacar

Laboratoire des Mathématiques, Statistiques, Informatique et Applications (LMSIA), Département des Mathématiques, Physique-Chimie et Informatique, Faculté des Sciences et Techniques, Université des Comores, Moroni, Comoros
Email: crawhoudine@gmail.com, halassi.abdoul@univ-comores.km

How to cite this paper: Rawhoudine, S.C. and Halassi Bacar, A. (2024) A New Multiobjective Particle Swarm Optimization Using Local Displacement and Local Guides. *Open Journal of Optimization*, 13, 31-49.

<https://doi.org/10.4236/ojop.2024.132003>

Received: February 15, 2024

Accepted: June 24, 2024

Published: June 27, 2024

Copyright © 2024 by author(s) and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution-NonCommercial International License (CC BY-NC 4.0).

<http://creativecommons.org/licenses/by-nc/4.0/>



Open Access

Abstract

This paper introduces a novel variant of particle swarm optimization that leverages local displacements through attractors for addressing multiobjective optimization problems. The method incorporates a square root distance mechanism into the external archives to enhance the diversity. We evaluate the performance of the proposed approach on a set of constrained and unconstrained multiobjective test functions, establishing a benchmark for comparison. In order to gauge its effectiveness relative to established techniques, we conduct a comprehensive comparison with well-known approaches such as SMPSO, NSGA2 and SPEA2. The numerical results demonstrate that our method not only achieves efficiency but also exhibits competitiveness when compared to evolutionary algorithms. Particularly noteworthy is its superior performance in terms of convergence and diversification, surpassing the capabilities of its predecessors.

Keywords

Particle Swarm Optimization, Multiobjective Optimization, Attractor-Based Displacement, Square Root Distance, Crowding Distance

1. Introduction

Optimization problems have multiple variables and constraints; resolving such problems requires compromises through the related variables (customers' desires) to get solutions that fit all the considered variables. The Customers' recommendations impose Engineers to come up with this purpose of optimization according to a set of requirements. The requirements restrain some variables of

the optimization problems into a subset of the searching area to stand to unconstrained or constrained optimization problems. These constrictions contribute to a considerably more complex problem. When leading with multiobjective problems, the objective functions are often conflicting. Consequently, a single optimal solution is inconceivable. Thus, a set of solutions known as the Pareto optimal set is searched instead of a single optimal solution. Moreover, the Pareto Front of all non-dominated solutions is established by the compromise solutions to a multiobjective optimization problem.

A Multiobjective Optimization Problem (MOP) can be stated as:

$$\begin{aligned} &\text{Minimize } f(x) = (f_1(x), \dots, f_n(x)), \\ &\text{Subject to } g(x) \leq 0 \text{ and } h(x) = 0, \end{aligned} \quad (1)$$

where the function $f(x)$ consists of n objective functions concerning inequality constraints functions $g(x)$ and equality constraints functions $h(x)$, where $x = (x_1, \dots, x_m) \in X \subset \mathbb{R}^m$ is an m -valued decision vector and X the decision space. In the following, we are referring to \mathcal{X} , the subset of solutions $x \in X$ that verify $g(x) \leq 0$ and $h(x) = 0$. The set \mathcal{X} is known as the feasible solutions set. Every element $x \in \mathcal{X}$ is paired to an objective vector

$y = (y_1, \dots, y_n) = (f_1(x), \dots, f_n(x))$, of which the image set \mathcal{Y} derives from the feasible solutions set \mathcal{X} by the mapping f is called the objective space.

MOPs are relevant to many engineering fields. The computation of MOP is extremely difficult by applying the exact standard methods. This became inappropriate to use exact methods, such Tabou search and Lexicographic method, in domains of which they must optimize large scale MOPs. Heretofore, various approximation methods are propounded to find their solutions. Among these methods, metaheuristics are essential for solving MOPs. According to techniques developed for that, metaheuristics dwell as the natural program of solving MOPs. Among these metaheuristics methods, genetic-based algorithms and multiobjective ant colony optimization [1] [2] such as NSGA-II [3], SPEA2 [4], and PESA2 [5] are wildly used for engineering problems because of their flexibilities and capacities to provide an acceptable set of optimal solutions. These nature based algorithms are skillful in significantly searching different parts of the Pareto Front [6] [7].

It is impressive how much particle swarm optimization (PSO) [8] has been utilized in recent decades. It has become an essential tool for many different applications. Plus, Pareto dominance plays a significant role in categorizing solutions. From the original PSO, many operators are included to strengthen the algorithm to compute MOP. For instance, various multiobjective particle swarm optimizations (MOPSO) are presented in [9] [10] [11]. In MOPSO-based algorithms, the search for a leader for particles is crucial. Many proposed techniques use a global leader for the swarm at each state. The paper [12] proposes a local leader for each particle using the square root distance. Other techniques, such as elitism techniques and mutation operators, are upgraded into the PSO-based multiobjective optimization algorithms towards better convergence and well-distributed

solutions along the true Pareto front. Significantly, similitude works have been summarized in [11] [12] for MOPSO-based algorithms using evolutionary algorithms techniques. It is known that MOPSO-based algorithms are more intensive but fail in diversifying the Pareto front.

The proposed optimization algorithm relies heavily on finding the best approximate solutions for conclusive results with balance between intensification and diversification. Especially as they compute and consolidate the Pareto Front close to the true Pareto Front and manage the diversification of optimal solutions throughout the entire Pareto Front. It should be noted that using a new local displacement of particles based on local attractors [13] [14] considers the intensification (convergence) process. Moreover, using square root distance for the non-dominated sorting instead of crowding distance improves the diversification. Both techniques prove that the algorithm may produce better optimal solutions to their optimization problems. Hence, in the current paper, the proposed algorithm incorporates the techniques of local attractors and the square root distance while determining the well-served Pareto part where the external archive is complete. Finally, the algorithm incorporates local guides for each particle instead of a global one, as it is widely used.

This paper is structured as follows. Section 2 reviews the MOPSO approach, and the proposed multiobjective particle swarm optimization technique is detailed and explained. Then to validate the proposed algorithm, performance metrics, and numerical validations are considered in Section 3 to compare the effectiveness of the proposed algorithm towards the SMOPSO, SPEA-II, and NSGA-II algorithms. Finally, concluding remarks and further considered works are presented in Section 4.

2. Particle Swarm Optimization

In [8], authors submitted a population-based metaheuristic named the PSO algorithm; their research was to model social interactions to satisfy a given set of desires. For example, the algorithm describes a swarm that travels from its nest to a food location, a fact that, in the end, the entire swarm follows the shortest way to get the food as quickly as possible. When the algorithm runs, a crowd of particles is produced in the search space arbitrarily, then it computes to find solutions and achieves them. Plus, after each iteration, The new solutions are updated in the archive set; each solution will be assessed with the precedent solutions stored in the archive set regarding the diversity (the disposition of particles) and convergence of the solutions. Therefore, only the best solutions are kept in the archive set.

Let us consider a swarm of K particles composed of its position and velocity vector as follows

$$x_k = (x_{k1}, \dots, x_{km}) \in \mathcal{X} \text{ and } V_k = (v_{k1}, \dots, v_{km}),$$

where \mathcal{X} is the m -dimensional feasible space. The following statements are the equations whose conduct particles to every possibility in the search space:

$$\begin{cases} V_{k,t+1} = \psi V_{k,t} + c_1 r_1 (P_{k,t} - x_{k,t}) + c_2 r_2 (G_{b,t} - x_{k,t}) \\ x_{k,t+1} = x_{k,t} + V_{k,t+1} \end{cases} \quad (2)$$

Let be the coefficient ψ , the inertia factor which sways the local and global abilities, and the velocity $V_{k,t}$ of the particle k at the t^{th} iteration. The constants c_1 and c_2 are related to the cognitive and social factors and $r_1, r_2 \sim U(0,1)$. In addition, the quantities $P_{k,t}$ and $G_{b,t}$ are assigned to the best value ($pBest$) of a particle k and the whole global best position ($gBest$) of the swarm, respectively. Globally, Equation (2) describes the motions concerning the particle's position and speed at each iteration.

Some particles will exit the search space when the PSO processes the swarm. Therefore, mitigating that fluctuation by restricting the search space is wise. Hence, particles that would move outside will be relocated by replacing their position values of particles with the boundary values. The repositioning values of particles are to ensure the convergence of solutions. When a particle dwells out of the limits, it prompts to set to the closest extremum value and back its corresponding velocity to an initial state to avoid any displacement beyond the boundary.

The problem with PSO-based algorithms is that they tend to converge quickly toward optimal solutions. For multiobjective optimization problems, they may improve the intensification and deteriorate its diversification. Focusing too much on one aspect of multiobjective optimization can be harmful and result in essential details being overlooked. Compromise is critical to success by weighing the constraints of each other and finding an optimal solution that satisfies and achieves a positive outcome.

2.1. Proposed Approach

The proposed approach, AMOPSO-SRD, extends the algorithm of AMOPSO [13]. The algorithm uses the modified Equation (3) to describe the motion of particles. The square root distance (SRD) computation mechanism is applied to select a local best guide in the external archive. Also, the SRD is used when new candidates' Pareto solutions must be inserted when the external archive is complete. Using a local guide for each particle and changing it at each iteration ensures the diversification of particle solutions. Moreover, the SRD provides a quick convergence, a fact that the proposed algorithm benefits by avoiding premature convergence and clustering particles regarding the diversification of solutions.

2.1.1. Local Displacement Using Attractors

Generally, the primary and classical displacement in Equation (2) are used for most PSO approaches. Nevertheless, a local displacement is proposed and investigated. This displacement is based on the local attraction of particles; to illustrate, a given particle at iteration t which has its position $X_{k,t}$, its velocity $V_{k,t}$, its best personal performance $P_{k,t}$, and the best global performance $G_{b,t}$ of the swarm, evolves in two different positions, $S_{k,t}$ and $T_{k,t}$, and accesses its new

position $X_{k,t+1}$ at the end. This process yields an equilibrium on convergency and diversification of the Pareto set throughout the computation [13]. The displacement mechanism is described by the four following equations at each iteration step:

$$\begin{aligned} S_{k,t} &= X_{k,t} + c_1 V_{k,t}, \\ T_{k,t} &= S_{k,t} + c_2 r_2 (P_{k,t} - S_{k,t}), \\ X_{k,t+1} &= T_{k,t} + c_3 r_3 (G_{k,t} - X_{k,t}), \\ V_{k,t+1} &= \delta V_{k,t} + \beta (P_{k,t} - X_{k,t}) + \gamma (G_{b,t} - X_{k,t}), \end{aligned} \quad (3)$$

where

$$\delta = c_1 (1 - c_2 r_2) \times (1 - c_3 r_3), \quad \beta = c_2 r_2 (1 - c_3 r_3), \quad \gamma = c_3 r_3. \quad (4)$$

c_1 , c_2 , and c_3 are random parameters and r_2 and $r_3 \sim \mathcal{U}(0,1)$. The following two sets of parameters are selected arbitrarily for the program, one set at a time:

$$\begin{aligned} &0 < c_1 < 0.9; \quad 0 < c_2 < 2; \quad 0 < c_3 < 2 \\ \text{or} \quad &0 < c_1 < 0.9; \quad 2 \leq c_2 < 4; \quad 2 \leq c_3 < 4. \end{aligned} \quad (5)$$

2.1.2. Diversification of the Pareto Front

All approaches are sorted out in performance according to their multiple criteria. In the following, the diversification process is introduced; also, a presentation of the operators used to achieve the ability of diversification. Diversification is always applied in the archive set. The archive set is an external set that stores non-dominated solutions at each iteration. When the archive set is deemed complete, there is a technique called elitism that can be applied to either suppress a non-dominated solution in favor of another one or to preserve it. The use of techniques such as hypercubes [9] or crowding distance (CD) [10] [11] can help to ensure that solutions are balanced and that the Pareto Front remains well-distributed. In this work, the CD quantifies the diversity of the Pareto front. The operator (CD) considers particles in the archive set into its process. It performs the criterion of the local best selection of non-dominated solutions. This operator is activated when the archive set is complete, and a non-dominated solution has to be inserted. Consequently, the CD assesses particles in the archive set; the particle with the smallest CD value within its neighbors will be deleted in favor of a new non-dominated solution. So far, particles in the archive set are prone to replacements with new ones if there are increments t . Therefore, the non-dominated solutions stay well diversified in the archive set.

The archive set provides various solutions ordered based on their objective function values, from the lowest to the highest. After processing the CD, all the solutions in the archive set are now available. The CD value of each particle is determined by finding the smallest value between itself and its neighboring particles. A particular assertion is also held: the lowest and highest objective function values must be upheld in the archive set; these boundaries' constants pos-

sess higher CD values allowing them to stay permanently in the archive. The algebraic operator of the k^{th} particle at iteration t is stated as follows:

$$\text{CD}(k) = \sum_{i=1}^n |f_i(x_{k+1,t}) - f_i(x_{k-1,t})|. \quad (6)$$

For each particle k , the CD is computed using the position values of the $k-1$ and $k+1$ particles. To all the CD values obtained, particles with small CD values are localized in crowded regions. Besides, those particles are more likely to be removed from the archive set and replaced by other non-dominated solutions concerning diversification.

Convergence empowers all methods of solving problems; it narrows the search toward a specific path. For example, Dynamic Neighbourhood Particle Swarm Optimization (DNPSO) [15] is a popular application of convergence known for its positive aspects in PSO approaches. Techniques like DNPSO aim to select a particle (among the non-dominated solutions in the archive set) or randomly choose it and use it as the global best guide at iteration t . On the other hand, a unique global best guide, $G_{b,t}$, only involves the convergence of non-dominated solutions. However, it affects the diversification of the Pareto set at iteration t unfavorably.

Using the crowding distance as an elitism technique is an effective diversification tool of the Pareto optimal set. Nevertheless, the convergence requires another operator to uphold the search for solutions toward the Pareto optimal set. Therefore, a local best guide of the particles is selected from the non-dominated solutions for each particle in the population. It demonstrates that each particle assigned by a local guide converges toward a different non-dominated solution in the archive set. As a result, we propose using a practical guide, $G_{k,t}$, to each particle which will allow a quick convergence to a non-dominated solution. The best guide is selected by using the SRD. The SRD between two points is defined as follows:

$$\text{SRD}(x_k, x_j) = \sum_{i=1}^n \sqrt{|f_i(x_k) - f_i(x_j)|}. \quad (7)$$

At iteration t , for each particle in the swarm, the square root distance between the particle and all the external archive members is calculated; the particle in the archive set with the shortest SRD will be selected as the global guide of the considered particle k . To illustrate, if, at a stage that the external archive contains l non-dominated solutions denoted y_1, y_2, \dots, y_l , the best guide $G_{k,t}$ of $x_{k,t}$ is chosen such that

$$\text{SRD}(x_{k,t}, G_{k,t}) = \min_{j \in [1,l]} \left(\sum_{i=1}^n \sqrt{|f_i(x_k) - f_i(y_j)|} \right). \quad (8)$$

3. Numerical Results

3.1. Performance Metrics

Few performance metrics are proposed for quantifying the performance of the

proposed approach. Usually significant aspects of measurement are used in the performance of multiobjective optimization algorithms; the generational distance, the spacing, and the C-metric are the operators for which the assessment of the proposed approach is conducted to quantify its performance. The proposed approach is compared to the SMOPSO [9], the generic-based evolutionary algorithms NSGA-II [3], and SPEA-II [5]. The use of SMOPSO, NSGA-II, and SPEA-II is motivated by their appreciation in computing MOPs. The General distance is well-known to be a useful metric where evaluating the convergence (intensification) of a MOP while the spacing metric is used for evaluating the diversification of the algorithm. And the last, the set coverage is used to compare the proposed method with the established methods.

3.1.1. Generational Distance (GD)

Despite the possibility of getting a solution in \mathcal{PF}_{known} , including in \mathcal{PF}_{true} ; in [16] authors encourage the use of general distance metrics to assess a mean distance of the solutions of \mathcal{PF}_{known} out of \mathcal{PF}_{true} . The algebraic form states as:

$$GD = \frac{1}{K} \sqrt{\sum_{k=1}^K d_k^2}, \quad (9)$$

where the representative of the number of solutions is K in \mathcal{PF}_{known} and the Euclidean distance, d_k , denoted the minimum distance between each vector in \mathcal{PF}_{known} to the closest element of \mathcal{PF}_{true} in the objective space. After computing, let us suppose $GD = 0$, which shows both sets are identical: the exact and approximated fronts, but any value different from zero means that \mathcal{PF}_{known} veers from \mathcal{PF}_{true} . Consequently, when we get a GD that tends to zero, it implies that the algorithm converges better.

3.1.2. Spacing (S)

The spacing metric [17] determines the uniformity of the stored non-dominated vectors scattered in every part of the \mathcal{PF}_{known} . The algebraic form states as follows:

$$S = \sqrt{\frac{1}{K-1} \sum_{k=1}^K (\bar{d} - d_k)^2} \quad (10)$$

with $d_k = \min_l \left(\sum_{i=1}^n |f_i(x_k) - f_i(x_l)| \right)$,

where $k, l = 1, \dots, K; l \neq k$ and \bar{d} describes the average of all d_k , K is the number of non-dominated solutions, and n is the dimension of the number of objectives in the \mathcal{PF}_{known} set. If the value of S is equal to zero in this metric, it indicates that the particles in the \mathcal{PF}_{known} are evenly distanced. The value 0 serves as the reference point for measurement and analysis. The more the computed spacing value tends to zero, the more the method diversifies the solutions in the approximated Pareto Front. As a result, an algorithm excels at finding a set of non-dominated solutions when spacing S tends closer to zero.

3.1.3. Set Coverage (C-Metric)

The C-Metric is a reliable tool for comparing two approximated Pareto sets and determining which is more optimal in a given Pareto set. Let us consider A and B , two approximations set to the \mathcal{PF}_{true} of a MOP; the algebraic form of C-Metric is defined as:

$$C(A, B) = \frac{|\{b \in B \mid \exists a \in A : a \text{ dominates } b\}|}{|B|}. \quad (11)$$

This rate $C(A, B)$ is comprised between 0 and 1. When the value $C(A, B) = 1$, it stands that all solutions in A dominate solutions in B , as well as if $C(A, B) = 0$ means the contrary. The formula $C(A, B)$ is not a probabilistic relation that can fulfill the relation $C(A, B) = 1 - C(B, A)$, in addition. We will represent the AMOPSO-SRD algorithm by A and B , the considered algorithm to be compared for simulation tests.

3.2. Results and Discussion

A set of well-known test functions is involved to guide the assessment of the proposed algorithm. The test problems are as follows, Schaffer's problem, Viennet (3) problem, Kursawe problem, Deb problem, Binh problem, ConstrEx problem, Tanaka problem, and Viennet (4) problem. For all the selected algorithms we are going to use only a subset of parameter settings. Let us consider the parameter settings subset: 2000 generations, a population size of $N = 100$ particles, and the external archive is set to 100 as the maximum number of particles. The learning rates $0 < c_1 < 0.9$, $0 < c_2, c_3 < 2$ or $2 \leq c_2, c_3 < 4$ are randomly sorted out of their range.

The algorithm is performed on the Mathworks[®] MATLAB R2017b on a PC which has the current specifications: Intel[®] Core[™] i7-7700HQ CPU @2.80GHz, CUDA cores 1280 NVIDIA GeForce GTX 1060, and 8GB RAM. The operation system is Windows 10 pro.

3.2.1. Unconstrained Test Problems

The following unconstrained MOPs [17] (Table 1) are selected randomly to weigh up the capabilities and performances of the proposed approach.

The first test problem is the Schaffer problem. This test function is well-known for the propriety of having one decision variable and a simple Pareto front solution. Therefore, we selected it as a fresh start to process our algorithm. A single convex Pareto curve covers this multiobjective problem as \mathcal{PF}_{true} .

The second MOP, the Viennet problem, is an unconstrained three-objective minimization problem. This problem performs a disconnected surface as its optimal Pareto set \mathcal{P}_{true} and a three-dimensional complex curve as its Pareto Front \mathcal{PF}_{true} .

Furthermore, the \mathcal{P}_{true} of the Kursawe problem is displayed with a discontinuous and asymmetrical area. Moreover, its true Pareto Front \mathcal{PF}_{true} has only three discontinuous Pareto.

Table 1. Unconstrained functions.

MOP	Definition	Constraints
Schaffer problem	$F = (f_1(x), f_2(x))$ $f_1(x) = x^2,$ $f_2(x) = (x-2)^2$	$-10^5 \leq x \leq 10^5$
Viennet (3) problem	$F = (f_1(x, y), f_2(x, y), f_3(x, y))$ $f_1(x, y) = 0.5(x^2 + y^2) + \sin(x^2 + y^2),$ $f_2(x, y) = \frac{(3x-2y+4)^2}{8} + \frac{(x-y+1)^2}{27} + 15,$ $f_3(x, y) = \frac{1}{x^2 + y^2 + 1} - 1.1e^{-x^2 - y^2}$	$-4 \leq x, y \leq 4$
Kursawe problem	$F = (f_1(\mathbf{x}), f_2(\mathbf{x}))$ $f_1(\mathbf{x}) = \sum_{i=1}^{n-1} \left(-10e^{(-0.2\sqrt{x_i^2 + y_i^2})} \right),$ $f_2(\mathbf{x}) = \sum_{i=1}^n (x_i ^{0.8} + 5 \sin(x_i^3))$	$-5 \leq x_i \leq 5$ $i = 1, 2, 3$
Dep problem	$F = (f_1(x, y), f_2(x, y))$ $f_1(x, y) = x,$ $f_2(x, y) = (1+10y) \left(1 - \left(\frac{x}{1+10y} \right)^2 - \frac{x}{1+10y} \sin(2p\pi x) \right)$	$0 \leq x, y \leq 1$ $p = 6$

Finally, the Dep problem was propounded by K. Deb, and he used his methodology. The test function has two objective functions, and its \mathcal{P}_{true} is a disconnected set, whereas \mathcal{PF}_{true} is a discontinuous Pareto curve. The disconnected regions are tunable regarding the value p which is assigned to $p = 6$ in this work.

All the test problems are run through the proposed algorithm to acquire the figures (the \mathcal{PF}_{true}) and then compare with the results of the established methods.

The Schaffer test problem proved satisfaction to the proposed algorithm regarding the results (Table 2) and the \mathcal{PF}_{true} (Figure 1). Apart from its historical characteristics, this MOP does not cover all the range of the proposed algorithm capabilities. Nevertheless, the comparison gathered in Table 2 is consistent between the AMOPSO-SRD and the selected approaches, SMOPSO, NSGA-II, and SPEA-II, in terms of convergence. Moreover, from a different point of view, the SMOPSO, NSGA-II, and SPEA-II results are dominated by those from AMOPSO-SRD. For instance, 51% of non-dominated solutions from SMOPSO are dominated by those from AMOPSO-SRD, while only 1% of solutions in SMOPSO dominate results from AMOPSO-SRD. As well as the comparisons with NSGA-II and SPEA-II, the results showed 25% and 32% domination from AMOPSO-SRD solutions over those from NSGA-II and SPEA-II, respectively.

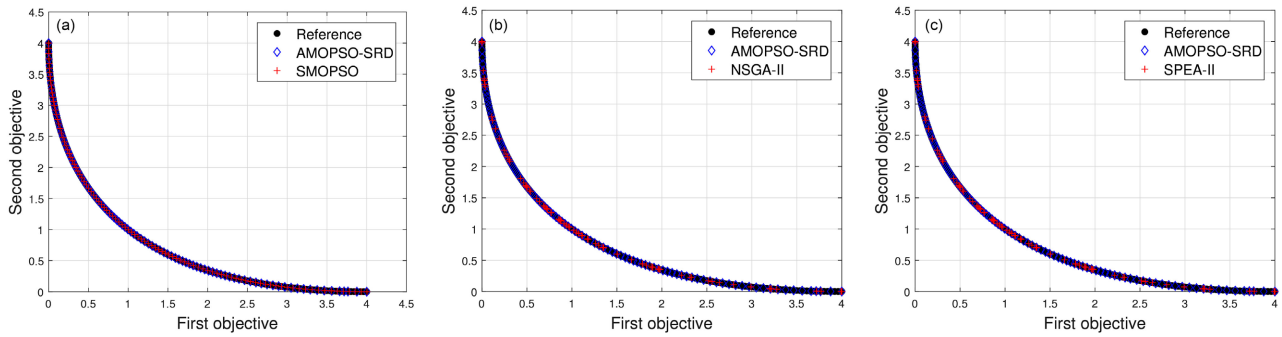


Figure 1. Comparison of the true Pareto Front and the approximated Pareto fronts for the Schaffer test problems. (a) Result from SMOPSO and AMOPSO-SRD; (b) Result from NSGA2 and AMOPSO-SRD; (c) Result from SPEA2 and AMOPSO-SRD.

Table 2. Comparison of the performance metrics between AMOPSO-SRD, SMOPSO, NSGA-II, and SPEA-II for the Schaffer test problem.

Metrics	AMOPSO-SRD	SMOPSO	NSGA-II	SPEA-II
GD	0.008910	0.009136	0.008243	0.008243
S	0.023952	0.009241	0.098832	0.098832
C (A, B)	--	0.510000	0.250000	0.320000
C (B, A)	--	0.010000	0.010000	0.010000

A represents the AMOPSO-SRD; B represents the algorithm in each column.

The Viennet (3) test problem showed progressive results from the proposed approach. The proposed approach’s PF_{true} (Figure 2) is well-distributed across the exact Pareto front. Regarding the results (Table 3), the proposed approach gives satisfactory outcomes. The use of the different metrics aforementioned (section 3.1) illustrates that the solutions from the proposed method are satisfactory towards the solutions from the SMOPSO, NSGA-II, and SPEA-II in general. The solutions’ distribution and convergence from the proposed method are efficient and excel to the target.

After the computation of the Kursawe test problem, the results are not as expected; the curves are not well aligned across the reference. Figure 3 displays specific outcomes about the PF_{true} , and Table 4 portrays disadvantage outcomes as none of the solutions from the proposed approach towards the selected approaches. With this test function, the experiment exhibits some erratic outcomes, although the proposed approach has a better distribution over the SMOPSO.

Furthermore, in the last one, the Deb problem, the more iteration, the better the non-dominated particles are distributed along the subregions. All the graphs made by the non-dominated particles throughout the selected different algorithms are as close as the reference graph. Figure 4 presents that solutions of AMOPSO-SRD are well scattered along the Pareto Front. However, all the comparisons’ results (from the C-metric) are not comparable. Solutions from the AMOPSO-SRD dominate solutions from the other algorithms significantly; the reverse, none of the solutions of SMOPSO, NSGA-II, and SPEA-II dominate so-

lutions from AMOPSO-SRD. Moreover, the results shown in **Table 5** demonstrate the accuracy of the graphs. The convergence of the AMOPSO-SRD is comparable with the convergence of SMOPSO. On the contrary, the results of NSGA-II and SPEA-II are efficient according to the generational distance. Regarding diversity, the solutions from AMOPSO-SRD are well scattered along the \mathcal{PF}_{true} . According to the dominance criteria, no solution from the other considered approaches is dominated by a solution from AMOPSO-SRD. In contrast, 10% of solutions from NSGA-II and 32% of solutions from SPEA-II are dominated by at least one solution from the proposed approach. Therefore, the proposed approach is productive and provides convergent solutions to the exact Pareto Front meticulously.

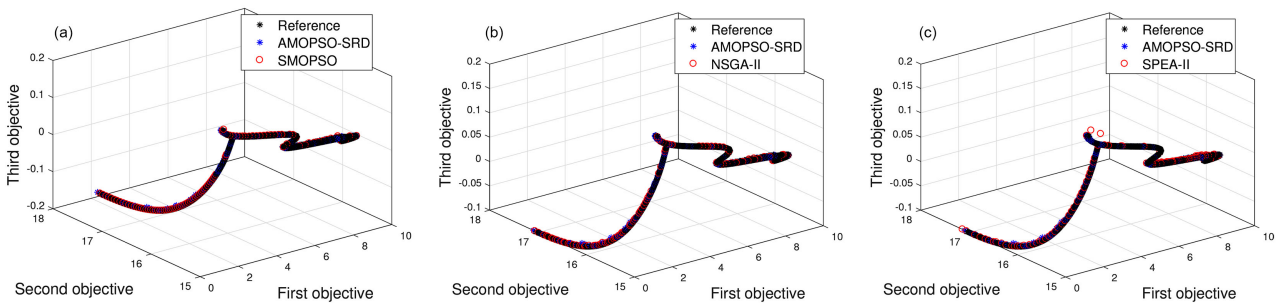


Figure 2. The true Pareto Front and the approximated Pareto fronts for the Viennet (3) test problems. (a) Result from SMOPSO and AMOPSO-SRD; (b) Result from NSGA2 and AMOPSO-SRD; (c) Result from SPEA2 and AMOPSO-SRD.

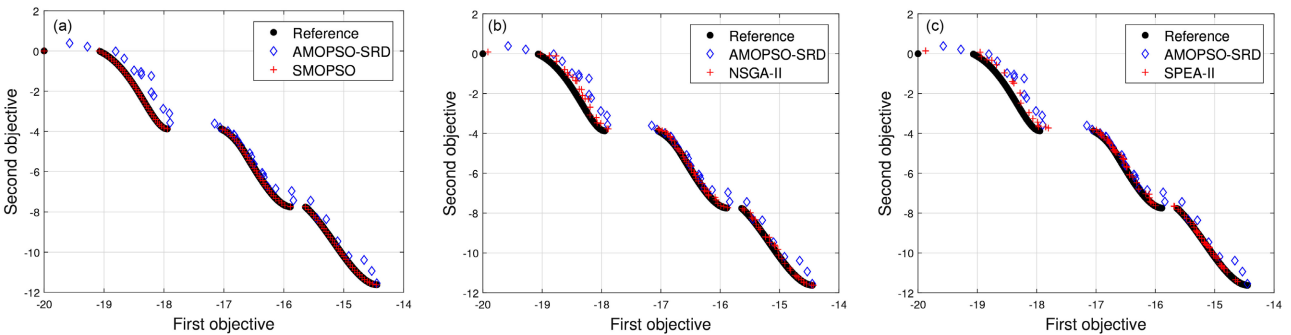


Figure 3. The true Pareto Front and the approximated Pareto fronts for the Kursawe test problems. (a) Result from SMOPSO and AMOPSO-SRD; (b) Result from NSGA2 and AMOPSO-SRD; (c) Result from SPEA2 and AMOPSO-SRD.

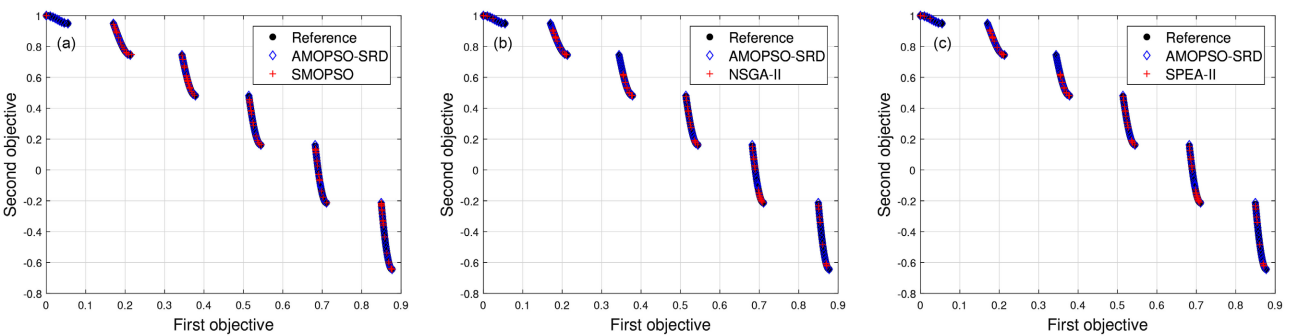


Figure 4. The true Pareto Front and the approximated Pareto fronts for the Deb (2) test problems. (a) Result from SMOPSO and AMOPSO-SRD; (b) Result from NSGA2 and AMOPSO-SRD; (c) Result from SPEA2 and AMOPSO-SRD.

Table 3. Comparison of the performance metrics between AMOPSO-SRD, SMOPSO, NSGA-II, and SPEA-II for the Viennet (3) test problem.

Metrics	AMOPSO-SRD	SMOPSO	NSGA-II	SPEA-II
GD	0.004519	0.004076	0.005560	0.006781
S	0.088465	0.097933	0.098559	0.049733
C (A, B)	--	0.050000	0.105263	0.105263
C (B, A)	--	0.157895	0.090000	0.090000

A represents the AMOPSO-SRD; B represents the algorithm in each column.

Table 4. Comparison of the performance metrics between AMOPSO-SRD, SMOPSO, NSGA-II, and SPEA-II for the Kursawe test problem.

Metrics	AMOPSO-SRD	SMOPSO	NSGA-II	SPEA-II
GD	0.065670	0.009644	0.021847	0.020815
S	0.175633	0.063547	0.126992	0.156223
C (A, B)	--	0.000000	0.030000	0.010000
C (B, A)	--	0.948718	0.846154	0.769231

A represents the AMOPSO-SRD; B represents the algorithm in each column.

Table 5. Comparison of the performance metrics between AMOPSO-SRD, SMOPSO, NSGA-II, and SPEA-II for the Deb (2) test problem.

Metrics	AMOPSO-SRD	SMOPSO	NSGA-II	SPEA-II
GD	0.003339	0.003427	0.002770	0.002770
S	0.014585	0.117855	0.059201	0.059201
C (A, B)	--	0.030000	0.100000	0.320000
C (B, A)	--	0.000000	0.000000	0.000000

A represents the AMOPSO-SRD; B represents the algorithm in each column.

3.2.2. Constrained Test Problems

The second set of problems listed in **Table 6** consists of four distinctive MOP with constrained functions [18].

The first test function of this bunch is a two-constrained MOP propounded by Binh and Korn [19]. It has two objective functions defined in a simple area: its true Pareto \mathcal{P}_{true} and its true Pareto Front \mathcal{PF}_{true} are connected and convex.

The second MOP is proposed and known for its proprieties [3]; it is used for constrained optimization. The two-objective MOP embeds both linear and nonlinear constraints. Multiobjective algorithms converge smoothly to the true Pareto optimal Front; however, the constraints lead the program to run heavily and hard into the process. Moreover, its constraints contribute to hindrances to finding a uniform distribution of solutions in the direction of the non-dominated Front. Therefore, persistent algorithms, the MOEA, rely on their constraint handling ability to perform such an arduous task. The Pareto Front of this function is simple and connected.

Table 6. Constrained functions.

MOP	Definition	Constraints
3*Binh (2) problem	$F = (f_1(x, y), f_2(x, y))$	$-15 \leq x, y \leq 30$
	$f_1(x, y) = 4x^2 + 4y^2$	$(x-5)^2 + y^2 + 25 \leq 0$
	$f_2(x, y) = (x-5)^2 + (y-5)^2$	$-(x-8)^2 - (y+3)^2 + 7.7 \leq 0$
3*ConstrEx problem	$F = (f_1(x, y), f_2(x, y))$	$0.1 \leq x \leq 1, 0 \leq y \leq 5$
	$f_1(x, y) = x,$	$9x + y - 6 \geq 0$
	$f_2(x, y) = \frac{1+y}{x},$	$9x - y - 1 \geq 0$
4*Tanaka problem	$F = (f_1(x, y), f_2(x, y))$	$0.1 \leq x, y \leq \pi$
	$f_1(x, y) = x,$	$-x^2 - y^2 + 1 + 0.1 \cos\left(16 \arctan\left(\frac{x}{y}\right)\right) \leq 0$
	$f_2(x, y) = y$	$(x-5)^2 + \left(y - \frac{1}{2}\right)^2 \leq 0$
4*Viennet (4) problem	$F = (f_1(x, y), f_2(x, y), f_3(x, y))$	$-4 \leq x, y \leq 4,$
	$f_1(x, y) = \frac{(x-2)^2}{2} + \frac{(y+1)^2}{13} + 3,$	$y < -4x + 4$
	$f_2(x, y) = \frac{(x+y-3)^2}{175} + \frac{(2y-x)^2}{17} - 13$	$x > -1$
	$f_3(x, y) = \frac{(3x-2y+4)^2}{8} + \frac{(x-y+1)^2}{27} + 15$	$y > x - 2$

The next is the Tanaka problem which has the following characteristics: two nonlinear constraints, a discrete genotype, and a phenotype Pareto curve. The Pareto Front of this problem is discontinuous. That characteristic leads to an algorithm processing with difficulties to succeed in finding the true Pareto front \mathcal{PF}_{true} [20].

Finally, let us consider the Viennet (4) problem a three-objection function. It performs an irregular-shaped area as the true Pareto in solution space and a three-dimensional convex surface as its true Pareto Front \mathcal{PF}_{true} .

As the precedent bunch of functions, the experiment is run through 2000 iterations with the proposed approach and the selected approaches, SMPSO, NSGA-II, and SPEA-II to obtain the following results of the constrained MOPs.

The first comments are for the Binh test problem. **Figure 5** displayed better distributions along the curve from the AMOPSO-SRD and SMOPSO computations. Furthermore, many elements are located at the edge of the curve for the AMOPSO-SRD, although the SMOPSO results diverge in this part of the curve. Nevertheless, we get a detrimental sample of points with the NSGA-II algorithm at the limit of the Pareto optimal curve and limited distribution of the Pareto optimal solutions. When the AMOPSO-SRD algorithm is assessed by the three metrics listed above, the solutions obtained are efficient and accurate. From these results in **Table 7**, for each solution from AMOPSO-SRD, it is at least a quarter of the solutions if it dominates solutions from SMOPSO, NSGA-II, or

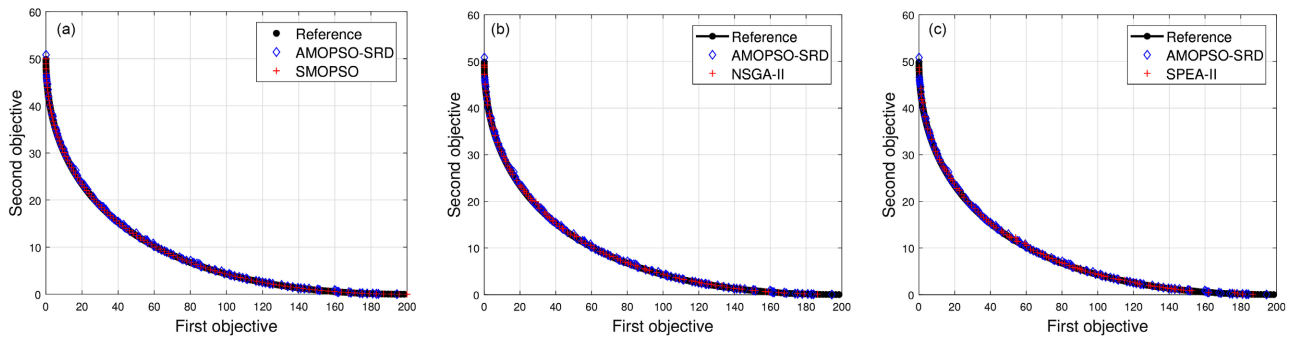


Figure 5. Comparison of the true Pareto Front and the approximated Pareto fronts for Binh2 test problems. (a) Result from SMOPSO and AMOPSO-SRD; (b) Result from NSGA2 and AMOPSO-SRD. (c) Results from SPEA2 and AMOPSO-SRD.

Table 7. Comparison of the performance metrics between AMOPSO-SRD, SMOPSO, NSGA-II, and SPEA-II for the Binh (2) test problem.

Metrics	AMOPSO-SRD	SMOPSO	NSGA-II	SPEA-II
GD	0.044607	0.050745	0.050589	0.058933
S	0.410737	0.292599	0.387257	0.283433
C (A, B)	--	0.316129	0.240000	0.140000
C (B, A)	--	0.000000	0.180645	0.180645

A represents the AMOPSO-SRD; B represents the algorithm in each column.

SPEA-II. SPEA-II or NSGA-II solutions can only dominate 18% of AMOPSO-SRD solutions, while SMOPSO does not dominate any solution from AMOPSO-SRD.

The ConstrEx test problem presents promising results as well as the precedent. The Pareto Front Figure portrayed is given by the NSGA-II, SMOPSO, SPEA-II, and AMOPSO-SRD. It appears in **Figure 6** that the AMOPSO-SRD method achieves the best convergence than others. AMOPSO-SRD and SMOPSO provide the closest solutions on the left part because of a well-distributed Pareto Front. Moreover, the classical genetic algorithms make unsuccessful results to the optimal solutions. Let us compare the metrics (**Table 8**); we have the following results: $GD = 0.006824$ and $S = 0.070980$ for the AMOPSO-SRD while metrics from other algorithms are slightly more significant. Regarding the convergence, both SMOPSO and AMOPSO-SRD results are comparative; however, **Figure 6** shows the AMOPSO-SRD algorithm giving a well-distributed Pareto solution rather than NSGA-II or SPEA-II. Consider the C-metric to analyze the proportion of domination amongst the solutions from different algorithms. The results express that the dominance of solutions from the AMOPSO-SRD is significant toward those solutions from SMOPSO, NSGA-II, and SPEA-II. To illustrate, each solution from AMOPSO-SRD dominates at least 52%, 38%, and 28% of solutions from SMOPSO, NSGA-II, and SPEA-II, respectively. On the contrary, solutions from AMOPSO-SRD are only dominated by 41%, 19%, and 22% from the SMOPSO, NSGA-II, and SPEA-II, respectively.

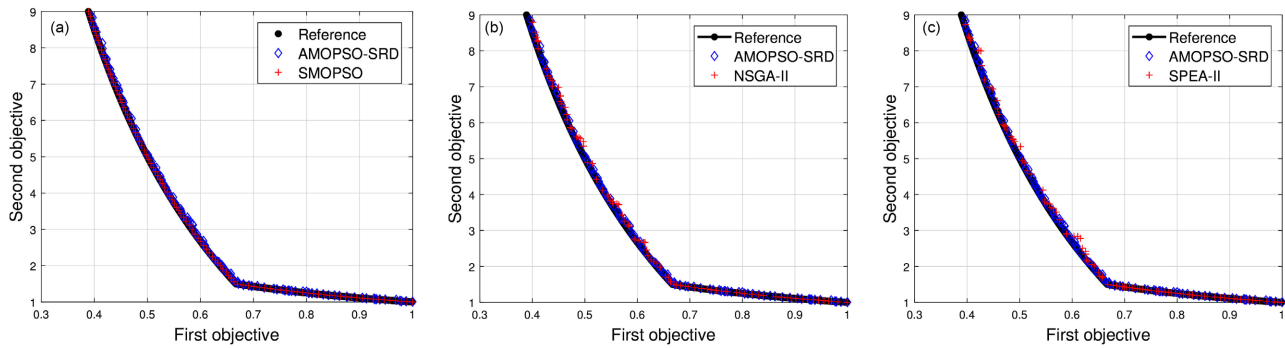


Figure 6. Comparison of the true Pareto Front and the approximated Pareto fronts for ConstrEx test problems. (a) Result from SMOPSO and AMOPSO-SRD; (b) Result from NSGA2 and AMOPSO-SRD. (c) Results from SPEA2 and AMOPSO-SRD.

Table 8. Comparison of the performance metrics between AMOPSO-SRD, SMOPSO, NSGA-II, and SPEA-II for the ConstrEx test problem.

Metrics	AMOPSO-SRD	SMOPSO	NSGA-II	SPEA-II
GD	0.006824	0.006997	0.007521	0.007624
S	0.070980	0.085065	0.118757	0.110930
C (A, B)	--	0.527027	0.380000	0.280000
C (B, A)	--	0.410000	0.189189	0.222973

A represents the AMOPSO-SRD; B represents the algorithm in each column.

Furthermore, the Tanaka problem illustrates significant improvement as **Figure 7** displays that all algorithms, AMOPSO-SRD, SMOPSO, NSGA-II, and SPEA-II output graphs, tend to the true Pareto Front and encompass the Pareto Front. We run 20 times the same algorithm using 2000 iterations to get **Figure 7** and the comparison results (**Table 9**). The performance metrics data prompt that AMOPSO-SRD is better than SPEA-II or NSGA-II in distribution and has efficient results. Besides, the C-metric results sustain the efficiency of AMOPSO-SRD. Each solution can dominate at least 47% of solutions from SMOPSO, 50% from NSGA-II, and 47% from SPEA-II.

At last, the Viennet (4) problem, **Figure 8** shows an exemplary distribution along the true Pareto Front from the computed solutions. The data performed by AMOPSO-SRD show a better convergence toward the Pareto than the NSGA-II and SPEA-II, except for the SMOPSO solutions with the smallest value recorded in the generational distance (**Table 10**). The results computed by the spacing metric demonstrate that solutions from AMOPSO-SRD are better distributed than the results from SMOPSO and NSGA-II; only the results from SPEA-II present a better diversity than those from AMOPSO-SRD, with 0.065328 against 0.084413. Moreover, last, regarding dominance, the dominance of solutions from AMOPSO-SRD towards the other algorithms is small or negligible, with 1% over SMOPSO solutions, 12% over NSGA-II solutions, and 10% over SPEA-II solutions. The reverse, the comparison is significant; a solution from SMOPSO dominates 20% of solutions from AMOPSO-SRD, a solution from NSGA-II dominates at least 26% of solutions from AMOPSO-SRD, and a

solution from SPEA-II dominates at least 43% of solutions from AMOPSO-SRD. It concludes that this test function has overall results which balance both the convergency and distribution better than the SMOPSO and NSGA-II except the SPEA-II.

Table 9. Comparison of the performance metrics between AMOPSO-SRD, SMOPSO, NSGA-II, and SPEA-II for the Tanaka test problem.

Metrics	AMOPSO-SRD	SMOPSO	NSGA-II	SPEA-II
GD	0.005905	0.005595	0.011648	0.011325
S	0.030650	0.015411	0.051720	0.059284
C (A, B)	--	0.467890	0.500000	0.477612
C (B, A)	--	0.070000	0.082569	0.073394

A represents the AMOPSO-SRD; B represents the algorithm in each column.

Table 10. Comparison of the performance metrics between AMOPSO-SRD, SMOPSO, NSGA-II, and SPEA-II for the Viennet (4) test problem.

Metrics	AMOPSO-SRD	SMOPSO	NSGA-II	SPEA-II
GD	0.012228	0.008051	0.012867	0.013069
S	0.084413	0.124016	0.158684	0.065328
C (A, B)	--	0.010000	0.120000	0.100000
C (B, A)	--	0.208000	0.264000	0.434000

A represents the AMOPSO-SRD; B represents the algorithm in each column.

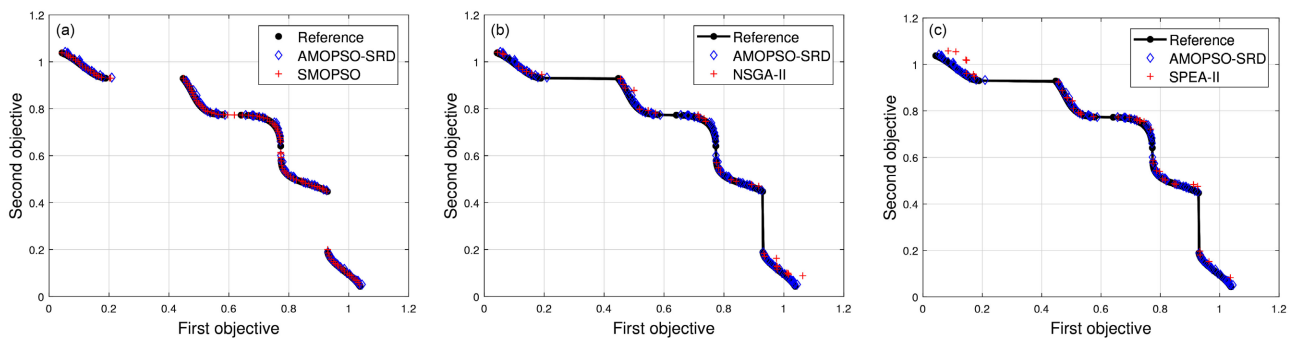


Figure 7. Comparison of the true Pareto Front and the approximated Pareto fronts for the Tanaka test problems. (a) Result from SMOPSO and AMOPSO-SRD; (b) Result from NSGA2 and AMOPSO-SRD. (c) Results from SPEA2 and AMOPSO-SRD.

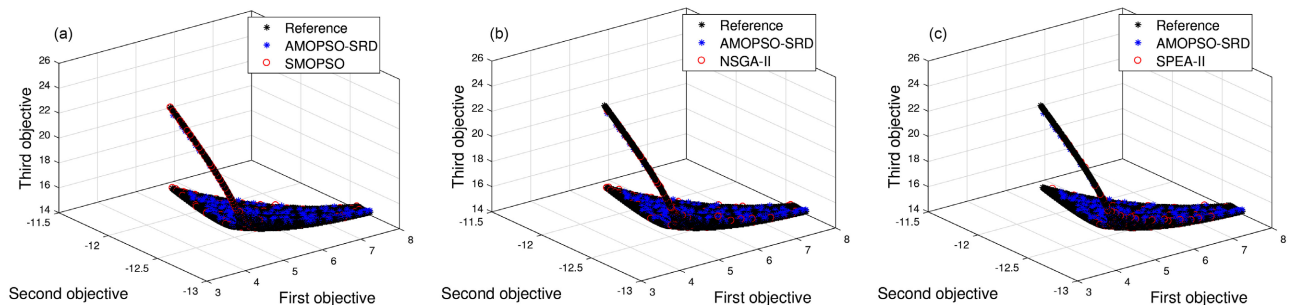


Figure 8. Comparison of the true Pareto Front and the approximated Pareto fronts for the Viennet (4) test problems. (a) Result from SMOPSO and AMOPSO-SRD; (b) Result from NSGA2 and AMOPSO-SRD. (c) Results from SPEA2 and AMOPSO-SRD.

These results show that the AMOPSO-SRD algorithm gives an advantageous distribution and convergence throughout the Pareto Front.

By analysing the results of the selected test functions given by each technique, the proposed method excels in diversification better than the established techniques. The test functions prove that the proposed method is effective and competitive regarding the established techniques. Regarding the diversity, the proposed method displays incredible alignment graphs towards the true Pareto Front; however the established techniques do not meet that aspect. This particular aspect drives the proposed algorithm to lift the difficulty that the established techniques face. The convergence occurs stable and consistent with all the considered techniques.

4. Concluding Remarks

In this study, we highlighted a novel variant of the particle swarm approach, incorporating local particle displacement and utilizing the square root distance. The proposed method is employed on a curated set of test functions chosen as samples to validate its capabilities. Additionally, we consider three distinct programs—SMOPSO, SPEA-II, and NSGA-II—for the purpose of comparing and evaluating the effectiveness of the proposed method.

The analysis of the numerical results demonstrates that the proposed method performs better than the SMOPSO, SPEA-II, and NSGA-II towards multiobjective problems. Furthermore, the assessment and analysis of the convergence and diversity of solutions from AMOPSO-SRD with the solutions from SMOPSO, SPEA-II, and NSGA-II through the three different performance metrics to the true Pareto and the true Pareto Front set demonstrate that AMOPSO-SRD exceeds the SMOPSO, SPEA-II, and NSGA-II. Therefore, the results prove that using adaptive techniques in the native PSO technique for multiobjective problems improves its capabilities regarding diversification and convergence of the Pareto optimal set. The forthcoming research will center around the utilization of multiquadric radial basis functions to optimize both the associated shape parameter and the conditioning number of the system matrix.

Acknowledgements

The authors would like to thank anonymous referees for giving very helpful comments and suggestions that have greatly improved this paper.

Disclosure Statement

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding

This work was supported by the French Embassy in Comoros, through the

project “FEF No. 0209-COM-23-0002—Comores Recherche, Innovation et Technologies (CRIT)”.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Rada-Vilela, J., Chica, M., Cordon, Ó. and Damas, S. (2013) A Comparative Study of Multi-Objective Ant Colony Optimization Algorithms for the Time and Space Assembly Line Balancing Problem. *Applied Soft Computing*, **13**, 4370-4382. <https://doi.org/10.1016/j.asoc.2013.06.014>
- [2] Özkale, C. and Fıglalı, A. (2013) Evaluation of the Multiobjective Ant Colony Algorithm Performances on Biobjective Quadratic Assignment Problems. *Applied Mathematical Modelling*, **37**, 7822-7838. <https://doi.org/10.1016/j.apm.2013.01.045>
- [3] Deb, K., Agrawal, S., Pratap, A. and Meyarivan, T. (2000) A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. In: Schoenauer, M., et al., Eds., *Parallel Problem Solving from Nature PPSN VI*, Springer, 849-858. https://doi.org/10.1007/3-540-45356-3_83
- [4] Zitzler, E. Laumanns, M. and Thiele, L. (2001) SPEA2: Improving the Strength Pareto Evolutionary Algorithm. *TIK Report*, **103**. <https://doi.org/10.3929/ethz-a-004284029>
- [5] Corne, D.W., Jerram, N.R., Knowles, J.D. and Oates, M.J. (2001) PESA-II: Region-Based Selection in Evolutionary Multiobjective Optimization. *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2001)*, San Francisco, 7-11 July 2001, 283-290.
- [6] Ishibuchi, H., Narukawa, K., Tsukamoto, N. and Nojima, Y. (2008) An Empirical Study on Similarity-Based Mating for Evolutionary Multiobjective Combinatorial Optimization. *European Journal of Operational Research*, **188**, 57-75. <https://doi.org/10.1016/j.ejor.2007.04.007>
- [7] Sarker, R., Liang, K. and Newton, C. (2002) A New Multiobjective Evolutionary Algorithm. *European Journal of Operational Research*, **140**, 12-23. [https://doi.org/10.1016/s0377-2217\(01\)00190-4](https://doi.org/10.1016/s0377-2217(01)00190-4)
- [8] Kennedy, J. and Eberhart, R. (1995) Particle Swarm Optimization. *Proceedings of ICCN'95—International Conference on Neural Networks*, **4**, 1942-1948.
- [9] Coello Coello, C.A. and Lechuga, M.S. (2002) MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization. *Congress on Evolutionary Computation*, **2**, 1051-1056.
- [10] Li, W.X., Zhou, Q., Zhu, Y. and Pan, F. (2012) An Improved MOPSO with a Crowding Distance Based External Archive Maintenance Strategy. In: Tan, Y., Shi, Y. and Ji, Z., Eds., *Advances in Swarm Intelligence. ICSI 2012*, Springer, 74-82.
- [11] Raquel, C.R. and Naval, P.C. (2005). An Effective Use of Crowding Distance in Multiobjective Particle Swarm Optimization. *Proceedings of the 7th annual conference on Genetic and Evolutionary Computation*, Washington DC, 25-29 June 2005, 257-264. <https://doi.org/10.1145/1068009.1068047>
- [12] Leung, M., Ng, S., Cheung, C. and Lui, A.K. (2014). A New Strategy for Finding Good Local Guides in Mopso. 2014 *IEEE Congress on Evolutionary Computation*

- (CEC), Beijing, 6-11 July 2014, 1990-1997. <https://doi.org/10.1109/cec.2014.6900449>
- [13] Halassi, A. (2016) An Attractor-Based Multiobjective Particle Swarm Optimization. *International Journal of Applied and Computational Mathematics*, **3**, 1019-1036. <https://doi.org/10.1007/s40819-016-0156-9>
- [14] Kemmoé Tchomt , S. and Gourgand, M. (2009) Particle Swarm Optimization: A Study of Particle Displacement for Solving Continuous and Combinatorial Optimization Problems. *International Journal of Production Economics*, **121**, 57-67. <https://doi.org/10.1016/j.ijpe.2008.03.015>
- [15] Hu, X. and Eberhart, R. (2002) Multiobjective Optimization Using Dynamic Neighbourhood Particle Swarm Optimization. *Proceedings of the 2002 Congress on Evolutionary Computation*, Honolulu, 12-17 May 2002, 1677-1681.
- [16] van Veldhuizen, D.A. and Lamont, G.B. (1999). Multiobjective Evolutionary Algorithm Test Suites. *Proceedings of the 1999 ACM Symposium on Applied Computing*, San Antonio, 28 February-2 March 1999, 351-357. <https://doi.org/10.1145/298151.298382>
- [17] Schott, J. (1995) Fault Tolerant Design Using Single and Multi-Criteria Genetic Algorithms. Ph.D. Thesis, Massachusetts Institute of Technology of Boston.
- [18] Coello Coello, C.A., Lamont, G.B. and Van Veldhuizen, D.A. (2007) Evolutionary Algorithms for Solving Multi-Objective Problems. 2nd Edition, Springer.
- [19] Binh, T.T. and Korn, U. (1997) MOBES: A Multiobjective Evolution Strategy for Constrained Optimization Problems. *Proceedings of the Third International Conference on Genetic Algorithms (MENDEL97)*, 176-182.
- [20] Tanaka, M., Watanabe, H., Furukawa, Y. and Tanino, T. (1995) GA-Based Decision Support System for Multicriteria Optimization. *IEEE International Conference on Systems, Man and Cybernetics. Intelligent Systems for the 21st Century*, Vancouver, 22-25 October 1995, 1556-1561.