Iranian Journal of Numerical Analysis and Optimization
Vol. 15, No. 1, 2025, pp 197–219
https://doi.org/10.22067/ijnao.2024.86247.1371
https://ijnao.um.ac.ir/

Research Article

Smith chart-based particle swarm optimization algorithm for multi-objective engineering problems

A. Falloun*, Y. Dursun and A. Ait Madi

Abstract

Particle swarm optimization (PSO) is a widely recognized bio-inspired algorithm for systematically exploring solution spaces and iteratively identifying optimal points. Through updating local and global best solutions, PSO effectively explores the search process, enabling the discovery of the most advantageous outcomes. This study proposes a novel Smith chart-based particle swarm optimization to solve convex and nonconvex multi-objective engineering problems by representing complex plane values in

*Corresponding author

Received 5 January 2024; revised 22 April 2024; accepted 26 May 2024

Abdelbari Falloun
Advanced Systems Engineering Laboratory, National School of Applied Sciences, Kenitra, Morocco. e-mail: abdelbari.falloun@uit.ac.ma

Yunus Dursun
Electrical and electronic engineering, Marmara University,Istanbul, Türkiye. e-mail: yunus.dursun@marmara.edu.tr

Abdessalam Ait Madi
Advanced Systems Engineering Laboratory, National School of Applied Sciences, Ibn Tofail University, Kenitra, Morocco. e-mail: abdessalam.aitmadi@uit.ac.ma

How to cite this article
Falloun, A., Dursun, Y. and Ait Madi, A., Smith chart-based particle swarm optimization algorithm for multi-objective engineering problems. *Iran. J. Numer. Anal. Optim.*, 2025; 15(1): 197-219. https://doi.org/10.22067/ijnao.2024.86247.1371

a polar coordinate system. The main contribution of this paper lies in the utilization of the Smith chart's impedance and admittance circles to dynamically update the location of each particle, thereby effectively determining the local best particle. The proposed method is applied to three test functions with different behaviors, namely concave, convex, noncontinuous, and nonconvex, and performance parameters are examined. The simulation results show that the proposed strategy offers successful convergence performance for multi-objective optimization applications and meets performance expectations with a well-distributed solution set.

AMS subject classifications (2020): Primary 90C26; Secondary 58E17, 90C24.

Keywords: Multi-objective optimization (MOO); particle swarm optimization (PSO); meta-heuristic optimization.

1 Introduction

Modern mathematical design problems necessitate the simultaneous solution of multiple objectives. Most of these problems involve objectives with contrasting behaviors. When one objective is enhanced, others tend to suffer as a consequence due to their contrasting behaviors. Therefore, it is not accurate enough to talk about a single solution to these problems; instead, getting a collection of solutions is needed. Multi-objective optimization methods help find optimal solutions that simultaneously balance and optimize multiple objectives, considering their contrasting behaviors. There are several strategies concerning multi-objective optimization in literature. The most well-known ones are evolutionary algorithm [22], genetic algorithm [1], NSGA-II [4], and particle swarm optimization (PSO).

PSO is an optimization method that uses the social behavior of fish populations or flocks of birds and is based on a population of stochastic agents. In the PSO algorithm, particles search for the optimal solution through the direction of the particle with the most excellent performance. The global optimum is a singular solution point; changing the position according to the particle performs best. However, in multi-objective optimization problems, since multiple solution points represent multi-objective solutions, to obtain the solution set, the particle positions should be updated not according to the same particle but according to the most suitable particle to reach the solution set. Therefore, in PSO with multiple objectives, the concept of “local best” replaces the idea of “global best” that guides the swarm [19, 13].

The conventional multi-objective particle swarm optimization (MOPSO) combines PSO and multi-objective optimization. Its weaknesses need to be addressed, particularly regarding its convergence in high-dimensional problems and its capability to handle constraints. Therefore, several modified multi-objective particle swarm optimization (M-MOPSO) algorithms have emerged as promising options that overcome these limitations and demonstrate good convergence in solving multi-objective problems, making them suitable for real-world applications. To highlight the advantages of M-MOPSO, we can compare it to MOPSO as follows [3].

The first difference lies in the population evolution procedure. MOPSO employs a swarm intelligence technique where the entire population coordinates their movements by following a single leader. On the other hand, M-MOPSO utilizes a boundary control mechanism, allowing each individual to evolve independently according to their own boundary. The second difference pertains to the mutation procedure. In MOPSO, the mutation is applied after finding the new position of the population. However, in M-MOPSO, the mutation is directly applied to the old population before evolution. The third difference involves the repository update procedure. MOPSO updates the contents of its repository only after generating and mutating the new population. In contrast, M-MOPSO updates the contents of its repository after mutating the old population and generating the new population. Additionally, there are differences in the repository member selection procedure. MOPSO determines the domination of each repository member every time new members are admitted, while M-MOPSO only determines the domination of necessary members and disregards any redundant solutions, not admitting them into the repository. Another distinction is found in the repository member deletion procedure. MOPSO deletes its repository members based on the density of the grids, whereas M-MOPSO deletes its repository members based on the Euclidean distance in the objective space between each repository member and the latest admitted member. Finally, the difference lies in the global best selection or population update. In MOPSO, a single new leader (global best) is selected in each iteration, replacing the previous leader. In contrast, in M-MOPSO, each individual has their own best particle (local best). If an individual cannot improve to find a better solution after a predetermined number of iterations, they are no longer updated.

Since the basic purpose of multi-objective optimization is to obtain all solution points that make up, the solution set most accurately, a successful method or approach is needed to determine the local best particle. The proposed technique uses the Smith chart's impedance and admittance circles to determine the local best particle. Three test functions with different behaviors are used to apply the suggested method, and its efficacy is assessed in terms of the general distance and spacing metrics.

1.1 Related works

The authors in [21] presented a multi-adaptive strategy-based PSO. The primary goal is to successfully preserve population variety, which is accomplished by segmenting the population into a number of swarms that may be moved about throughout the course of the evolutionary process. Particles in a swarm dynamically select their learning exemplars within each generation based on their personal performance. This enables individual particles within a single swarm to exhibit distinctive search behaviors in every generation, and even the same particle can exhibit various search behaviors throughout multiple generations. An adaptive method for population size (APS) is suggested to make the most use of computing resources. APS gives the population the ability to selectively eliminate harmful particles and introduce beneficial particles during the evolutionary process.

A unique strategy known as the classified perturbation mutation refers to a technique used in optimization algorithms to introduce small changes or perturbations to the current solution or position in the search space. This technique-based PSO method is suggested in this study. Every iteration of this algorithm evaluates and classifies each new personal best position's performance as either high quality or low quality. For the high-quality personal best position, a mutation approach with a smaller perturbation is subsequently designed to improve local search skills inside the potential search region. On the other hand, a more extensive perturbation mutation technique is intended for the low-quality personal best position in order to explore new areas and increase population variety [14].

A novel adaptation of PSO known as noninertial opposition-based particle swarm optimization (NOPSO) is an enhanced variant of the traditional PSO algorithm that incorporates the concept of opposition-based learning. It introduces noninertial movement dynamics to improve the exploration and exploitation capabilities of the algorithm. This approach integrates an adaptive elite mutation strategy and a generalized opposition-based learning strategy [12].

1.2 The contributions of the paper

The main contributions of the paper can be listed as follows:

- In this paper, we introduce a novel approach that utilizes the PSO algorithm to address problems characterized by multiple objectives. Our proposed technique leverages the impedance and admittance circles of the Smith chart to identify the local best particle. impedance and admittance circles are used to update particle locations. The Smith chart-based particle swarm optimization (SC-PSO) technique is repeated during the iterative refining phase to allow particles to progressively approach optimum solutions and explore the solution space. Convergence towards optimal results is facilitated by updating particle locations and velocities according to their local and global best solutions.

- The paper compares the simulation performance of the Smith-chart-based multi-objective optimization algorithm with some well-known benchmark schemes. The algorithm's performance was assessed using three test functions: Kursawe, Fonseca & Fleming, and Schaffer. The analysis in a Smith chart-based PSO algorithm focused on two specific performance metrics: generational distance and spacing. Generational distance is a metric commonly used in multi-objective optimization techniques to assess the convergence and diversity of solutions. It measures the average distance between the algorithm's generated solutions and a reference set of known optimal or Pareto-optimal solutions. A lower generational distance indicates that the algorithm is producing solutions that are closer to the true optimal solutions. On the other hand, spacing is another performance metric used in multi-objective optimization to evaluate the uniformity and distribution of solutions across the objective space. It gauges how well the algorithm explores and covers the entire Pareto front or solution space. A higher spacing value suggests that the algorithm can generate diverse and well-distributed solutions. These performance metrics are employed to evaluate the quality of the algorithm's generated solutions in terms of their proximity to the true optimal solutions (generational distance) and the diversity and spread of solutions across the Pareto front (spacing). Through the analysis and monitoring of these metrics during the algorithm's execution, researchers can gain insights into its convergence, diversity, and overall effectiveness in finding optimal solutions for optimization problems related to RF applications. The analysis focused on two performance metrics: generational distance and spacing. The simulation results demonstrated the superior performance of the proposed algorithm compared to the benchmark schemes MISA, MicroGA, NSGA-II, and PAES. In addition, the time complexity of the proposed algorithm is provided.

Organization

The remainder of the paper is structured in the following manner: Section 2 introduces PSO and related works in engineering problems. Section 3 explains the multi-objective optimization concept and the Pareto optimal analysis. Section 4 presents the Smith-chart-based novel approach to the multi-objective optimization problem. Section 5 discusses the efficacy of the proposed method by numerical and statistical results. The paper is then concluded, and future work is discussed in Section 6.

2 Particle swarm optimization (PSO)

Kennedy and Eberhart created the population-based optimization method known as PSO to study the social interactions between groups of fish and birds [9, 23]. The PSO algorithms search a swarm of particles. They are used to solve an optimization problem, with each particle acting as a possible solution point where three parameters are defined for each particle.

- x_i is the particle's current location.
- v_i is the particle's current speed.
- p_i is the particle's optimal location for anyone.
- $fx_{i(t+1)}$ is the fitness value of particle i's new position at iteration $t+1$.
- $p_{i(t)}$ is the previous position of particle i at iteration t.
- $p_{i(t+1)}$ is the updated best previous position of particle i at iteration $t+1$.
- $p_{x_i(t+1)}$ is the current position of particle i's new position at iteration $t+1$.
- $p_{x_i(t)}$ is the current position of particle i at iteration t.
- i represents particles, and t represents iteration.

The particles update their best positions at each process step by comparing their experience with their current performance. The update process is performed. Here, f is the objective function:

$$p_i(t+1) = \begin{cases} p_i(t), & f(x_i(t+1)) \geq f(p_i(t)), \\ x_i(t+1), & f(x_i(t+1)) < f(p_i(t)). \end{cases} \tag{1}$$

In the PSO algorithms, particles update their velocity and position at each process step by comparing their performances and using where a particle in a cluster is located based on how well it performs in comparison to other particles. Using (1), the velocity of each particle is recalculated for each objective in each iteration by using (2a) and (2b):

$$v_{i,j}(t+1) = \omega[v_{i,j}(t)] + c_1 r_{1,j}(t)\,[p_{i,j}(t) - x_{i,j}(t)] + c_2 r_{2,j}(t)\,[g_j(t) - x_{i,j}(t)], \tag{2a}$$

$$x_i(t+1) = x_i(t) + v_i(t+1). \tag{2b}$$

In this case, the inertia coefficient ω specifies how much of the particles' initial velocity they will retain in the following stage of processing. The algorithm is based on two uses independent random variables ($\mathrm{r}_{1,j}\,(t) \sim U\,(0,1)$ and $r_{2,j}\,(t) \sim U\,(0,1)$). These random coefficients c_1 and c_2 are learning coefficients with values between [0,2]. Also, $p_{i,j}(t)$ is the value in the jth dimension of the most excellent possible location for each particle ith up to the tth processing step. Moreore, $g_j(t)$ represents the value of the particle i's best individual location in the jth dimension at the tth processing phase. $v_i(t+1)$ is the updated velocity of particle i at iteration $t+1$. The place in the processing phase of the swarm particle with the greatest performance j-dimensional number is shown in (3a) and (3b):

$$f(g(t)) = \min\left\{f(p_1(t)),\, f(p_2(t)), \ldots,\, f(p_s(t))\right\}, \tag{3a}$$

$$g(t) \in \left\{p_1(t), p_2(t), \ldots, p_s(t)\right\}. \tag{3b}$$

As seen in (2a), three parameters are important when updating a particle's location. First is the number of the particle's speed. The second represents the particle's personal best location. Lastly, the swarm particle performs best (globally best). Hence, the particles update the position by remembering their experience and being guided by the swarm's top-performing particles [23, 20].

2.1 PSO for engineering problems

PSO has been utilized successfully for solving various optimization problems, including function optimization, engineering design, and machine learning.

One benefit of PSO is that it is reasonably easy to apply and needs a few parameters to modify [8, 7].In [8], the authors used it in solving optimization problems in the area of solar energy and other renewable energy sources. PSO may be an effective technique for improving the structure and performance of solar energy structures, helping to improve their efficiency, reliability, and cost-effectiveness. In [15], the authors used PSO in solving minimize energy consumption while ensuring the quality of service requirements. This study revealed that a clustering algorithm based on PSO could be an effective solution. This algorithm was used to optimize the selection of cluster heads by considering different factors, such as the distance between nodes, the remaining energy of each node, and the network topology. In [10], the authors suggested a quantum mechanics-based PSO algorithm. It uses quantum-inspired operators to update the positions of particles in a swarm. The authors showed that QPSO could be used in various applications, including IoT (Internet of Things) applications. Overall, they are powerful optimization algorithms that can be used for various IoT applications. Its ability to efficiently search high-dimensional and nonconvex search spaces makes them particularly well-suited for optimization problems in the IoT domain. Gad [9] systematically reviewed and showed that PSO also had been successfully often used in engineering domains such as mechanical, electrical, civil, and chemical engineering. In mechanical engineering, PSO has been used to optimize the design of mechanical systems such as engines, turbines, and robots. PSO has been used in electrical engineering for power system optimization, load forecasting, and parameter estimation. In civil engineering, PSO has been used for structural optimization, traffic flow control, and water resource management. PSO has been used in chemical engineering for process optimization, parameter estimation, and experiment design. In [7], the authors studied the impact of PSO on wireless network applications such as wireless sensor networks, ad hoc networks, and routing optimization. PSO has been used for coverage optimization, energy-efficient routing, and node localization in wireless sensor networks. In ad-hoc networks, PSO has been used for topology control, routing, and power management. In routing optimization, PSO has been used to optimize the routing protocol parameters, such as the packet size, number of hops, and packet delay.

3 Multi-objective optimization

Multi-objective optimization is simultaneously minimizing or maximizing m objective functions $\overrightarrow{f}(\overrightarrow{x}) = (f_1(\overrightarrow{x}), f_2(\overrightarrow{x}), \ldots, f_m(\overrightarrow{x}))$ considering the constraints and the n decision variables on the solution set X. Combining each goal into a single objective function is one approach to solving issues involving multiple objectives in optimization. The most commonly used method of combining objective functions is to express these functions as weighted linear sums. The weights that each objective function should be multiplied by should be chosen so that the objective does not lose weight in the entire process [16]. The value obtained will depend on the consequences defined for each purpose. Nevertheless, a linear sum function constructed by appropriately selecting the weight values defined for each objective function can be optimized with any optimization algorithm to obtain highly successful values. The Pareto optimal analysis is one of the most reliable techniques for multi-objective optimization issues.

3.1 Pareto optimal analysis

In multi-objective optimization issues, the goals typically behave in opposition to one another. Therefore, a solution point converging to one of the available objectives moves away from the other. Multi-objective optimization issues in this situation can only be solved [2]. It is not always possible to guarantee that one suggested strategy outperforms the others. Instead, to achieve a collection of ideal solutions, so-called nondominated solutions (Figure 1). Pareto dominance relations define this collection of answers. Utilizes to be attained. For example; $\overrightarrow{x_1}, \overrightarrow{x_2} \in X$ let be two solution vectors in m objective functions be used in the optimization issue with multiple objectives. Following is an expression for the Pareto dominance connection between these two solution vectors:

1 If $\vec{x}_1 \preceq \vec{x}_2$ ($\vec{x}_1$ is less dominant), then $f_i(\vec{x}_1) \leq f_i(\vec{x}_2)$ for each objective function f_i, where $i \in \{1, 2, \ldots, m\}$. This implies that the objective

functions should be minimized.

2 If $\vec{x}_1 \prec \vec{x}_2$ ($\vec{x}_1$ is dominant), then $\vec{x}_1 \preceq \vec{x}_2$ and $f_j(\vec{x}_1) \leq f_j(\vec{x}_2)$ for at least one objective function f_j, where $j \in \{1, 2, \ldots, m\}$. This indicates that the objective functions should be minimized.

3 If $\vec{x}_1 \sim \vec{x}_2$ ($\vec{x}_1$ and $\vec{x}_2$ are not different), it means that neither $\vec{x}_1$ dominates $\vec{x}_2$ nor $\vec{x}_2$ dominates $\vec{x}_1$. In this case, there is no specific direction indicated for the objective functions.

Here x_1 *and* x_2 dominate the other solution vectors rather than one another. In this case, these two solution vectors are considered mutually optimal and called Pareto-optimal. The Pareto Frontier is a group of Pareto-optimal answers depicting the surface trade-off between various objective functions [17].

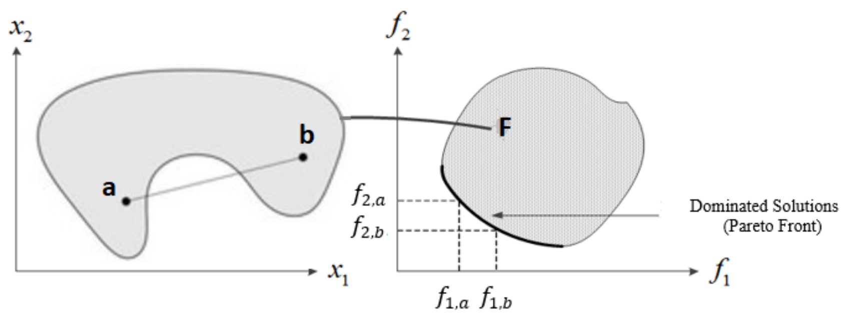

Figure 1: Transition from parameter space to function space and nondominated solutions

4 Smith chart approach

the Smith chart use as a visualization tool within the optimization process. There are several ways in which the Smith chart can be employed alongside the PSO algorithm. Firstly, in RF design scenarios, the PSO algorithm's fitness function often involves assessing impedance matching and transmission line properties. By employing the Smith chart, it becomes possible to visually examine and analyze the impedance variations of different solutions, thereby gaining valuable insights into their fitness values. Additionally, the Smith chart can aid in constraint analysis, particularly concerning impedance matching and transmission line parameters. Through visualizing the solutions on the Smith chart, it becomes simpler to identify any violations of constraints and guide the PSO algorithm towards exploring feasible regions. Furthermore, once the PSO algorithm concludes or reaches termination, the Smith chart can be employed to visually represent the best solutions obtained. This graphical representation offers a clear depiction of the optimized impedance matching or transmission line characteristics. In summary, the Smith chart serves as an auxiliary tool for visualizing and analyzing RF-related aspects of the optimization problem when used in conjunction with the PSO algorithm. In this section, Smith's chart-based PSO algorithm is explained.The optimization problem encompasses various dimensions, including the lower and upper limits of the parameters. It involves defining objective functions, determining the number of particles, and identifying the specific problems that require solutions. We first establish the boundaries within which the parameters can vary to tackle this optimization issue. These lower and higher limits provide the range within which the optimization algorithm will explore possible solutions. Next, we define the objective functions that we seek to optimize. These functions represent the goals we want to achieve or the criteria we want to maximize or minimize. By evaluating these functions, we can measure the quality or effectiveness of different solutions. Furthermore, we determine the number of particles to employ in the optimization algorithm. These particles represent potential solutions to the multi-objective problem. The number of particles influences the exploration and exploitation balance within the search space. Finally, we identify the specific problems that require improvement. These problems can be diverse in nature, ranging from optimizing complex to nonconvex functions. By addressing these problems through the optimization process, we aim to enhance performance. The particle locations are randomly distributed within the upper and lower limits. The best locations of each particle are identical to their starting positions, and the particle's initial velocities are given a value of zero. An archive is created to store the best solutions in each iteration. Each particle in the swarm is applied to the function values acquired and the objective functions. Each particle updates its performance by comparing its function values in the target space and past function values. Using the Pareto technique, the target space of particle values contains not dominated solutions. Also, the locations of the not dominated particles are then determined. The function values are sent to the archive. By selecting the finest guide, each particle modifies its weight and location.

The local best particle is identified as follows:

- The function values of the particles in the archive are matched to the Smith chart space with (4). Function values of ith archive member being $(f_{i,1}, f_{i,2})$ are converted to impedance and admittance values:

$$\begin{aligned} Z_i &= f_{i,1} + j f_{i,2} (\Omega), \\ Y_i &= \frac{1}{Z_i} (\Omega)^{-1}, \end{aligned} \tag{4}$$

where

- $f_{i,1}$ corresponds to the first function value of the ith archive member. Also, $f_{i,2}$ represents the second function value of the ith archive member.

- Z_i represent the impedance of particle i and Y_i represent the admittance of particle i.

- The function values of the other particles are also given in (4) and are mapped to the abstraction space.

- In this stage, each particle selects the most appropriate local best particle from the particles in the archive. Here, four different local best particles are chosen for a particle four different local best particles are selected for a particle as shown in Figure 2.

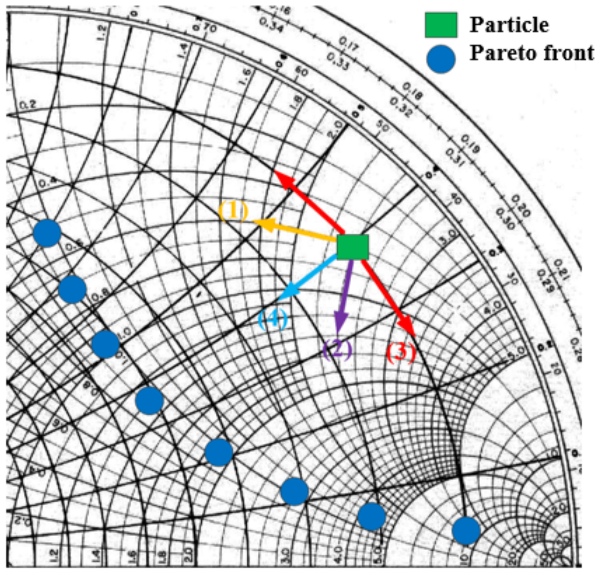

Figure 2: Local best particle determination of a particle

The archive members are categorized based on their proximity to different circles in the system. In Figure 2, (1) refers to archive members located on the nearest resistance circle. These members are situated in close proximity to the resistance values within the system. (2) represents archive members located on the nearest reactance circle, indicating their association with reactance values. (3) represents archive members positioned on the nearest conductivity circle, indicating their relationship with conductivity characteristics. Lastly, (4) represents archive members located on the closest susceptance circle, highlighting their connection to susceptance values within the system. Organizing the archive members this way makes it easier to effectively cover the whole solution space.

Figure 3 shows the particles and how they update their location depending on the different archive members. The figure also shows graphically the effect of the four circles itemized above. As can be seen, each case is concentrated in a specific region of the Pareto curve and eventually leads to the actual Pareto curve on the Pareto front.

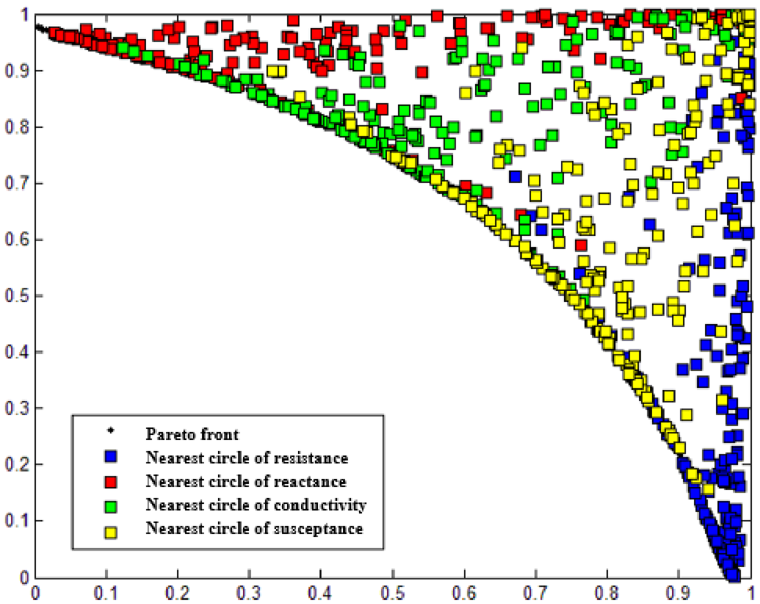

Figure 3: Effect of local best designation varieties on the Pareto front

4.0.1 Complexity analysis

The suggested Smith chart-based multi-objective PSO algorithm's time complexity is determined by the number of paths through the admittance and impedance circles, The number of swarming particles, the number of iterations, and the complexity of the optimized fitness function. Therefore, the time complexity able to be expressed as $\mathcal{O}(4 \times NP \times I \times F)$, where F denotes the complexity of the fitness function and NP is the number of particles. I denotes the number of iterations.

5 Results

Applying the suggested strategy to three frequently used test functions allows us to see how successfully the suggested approach works. For each test function, the algorithm was run thirty times. The maximum number of the iteration was 2000, and the number of particles was 100. Figures 4, 5, and 6 show the graphical results. The figures also show that the proposed algorithm finds the optimal solutions for different types of functions.

Test Function-I: Kursawe Test Function.

Minimize:

$$f_1(x) = \sum_{i=1}^{n-1} \left[-10e^{-0.2\sqrt{x_i^2+x_{i+1}^2}} \right], \tag{5}$$

$$f_2(x) = \sum_{i=1}^{n} \left(|x_i|^{0.8} + 5\sin(x_i^3) \right), \tag{6}$$

where $x = [x_1, x_2, \ldots, x_n]$ represents the decision variables, and x is the number of decision variables.

Kursawe is a well-known multi-objective optimization test function used to evaluate performance algorithms. Kursawe is twofold; the first is to minimize the sum of the absolute values of the difference between adjacent variables, while the second concerns minimizing a nonconvex function involving these variables. This test has been widely used as a benchmark function for testing the performance of multi-objective optimization algorithms due to its nonlinearity, nonconvexity, and presence of multiple local optima. All in all, the goal is to minimize both objectives simultaneously as shown in (5).

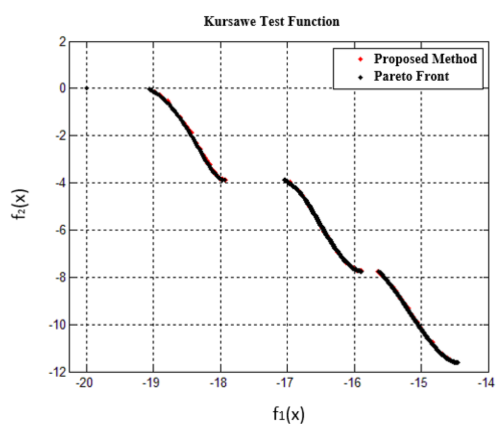

Figure 4: Pareto front obtained for Kursawe

Test Function-II: Schaffer-I Test Function

Minimize:

$$f_1(x) = x^2, \tag{7}$$

$$f_2(x) = (x-2)^2, \tag{8}$$

where $-A \leq x \leq A$, with $A = 100$. The complexity of this problem becomes considerably higher when the values of A exceed 10^5.

The Schaffer-I function is a popular test function used to evaluate the performance of an optimization algorithm. The Schaffer-I function is used as a single benchmark to test optimization algorithms due to its simplicity and a well-defined global minimum. It is often used with other more complex test functions to provide a more comprehensive evaluation of optimization algorithms, as shown in (7).

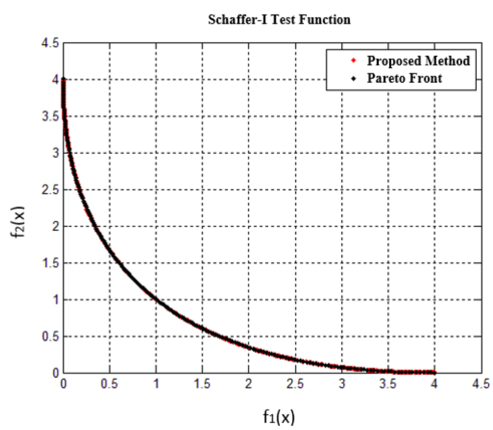

Figure 5: Pareto front obtained for Schaffer-I

Test Function-III: Fonseca & Fleming Test Function

Minimize:

$$f_1(x) = 1 - \exp\left(- \sum_{i=1}^{n} (x_i - \frac{1}{\sqrt{n}})^2 \right), \tag{9}$$

$$f_2(x) = 1 - \exp\left(-\sum_{i=1}^{n}(x_i + \frac{1}{\sqrt{n}})^2\right), \tag{10}$$

where $x = [x_1, x_2, \ldots, x_n]$ represents the decision variables, and n is the number of decision variables.

The Fonseca-Fleming is another popular test function used in the field of multi-objective optimization. This function involves a set of decision variables and two objectives that must be optimized simultaneously. The Fonseca-Fleming function is a challenging test for multi-objective optimization because it is nonlinear, nonconvex, and has multiple local optima; it is often used as a benchmark function for testing the performance of multi-objective optimization algorithms, as shown in (9).

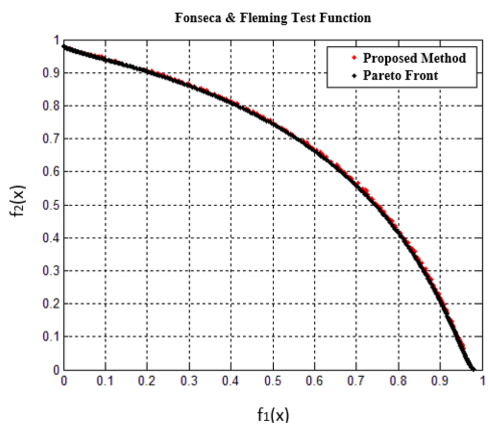

Figure 6: Pareto front obtained for Fonseca & Fleming

We also conducted evaluations to assess how well the suggested technique works in addition to the graphical results. These quantitative evaluations, in Tables 1 and 2, are the convergence and diversity parameters, respectively, developed by Coello and Cortés [5]. The convergence metric expresses the degree to which the Pareto front generated by the algorithm resembles the actual Pareto front. This metric has “0” as both the lowest and desired numbers. The other metric gives the distribution of the points that make up the Pareto front obtained by the algorithm. This metric, which has a minimum and target value of “0”, gives the distribution of the Pareto front between two extrapolated points. In other words, it expresses how evenly the obtained solutions are distributed. It can be concluded from also the tables that the proposed algorithm performs good convergence and diversity.

In Table 1, we evaluated the performance of our proposed algorithm on the Kursawe test function and compared it against other techniques.

Table 1: Generational distance performance of the proposed method

Test Function	Minimum	Maximum	Average	Median	Std. Dev
Kursawe	0.000879	0.0014	0.0011	0.0011	0.00007648
Schaffer	0.00049564	0.0005153	0.0005097	0.0005072	0.00000612
Fonseca and Fleming	0.00051618	0.0006628	0.0005821	0.0005896	0.00068063

Table 2: Spacing performance of the proposed method

Test Function	Minimum	Maximum	Average	Median	Std. Dev
Kursawe	0.0393	0.0996	0.0503	0.0464	0.0120
Schaffer	0.0109	0.0137	0.0123	0.0122	0.0013
Fonseca and Fleming	0.0066	0.0077	0.0071	0.0068	0.0005115

Table 3 clearly shows that our proposed method outperforms other methods in terms of spacing and generational distance metrics.

Table 3: Comparison with other methods in literature

Generational Distance	MISA	Micro GA	NSGA-II	PAES	Proposed Method
Average	0.00466	0.01133	0.01663	0.02450	0.0011
Best	0.00433	0.00698	0.00680	0.00648	0.000879
Worst	0.00558	0.02298	0.03447	0.09593	0.014
Std. Dev	0.00031	0.00538	0.00773	0.0244	0.00007648
Spacing	**MISA**	**Micro GA**	**NSGA-II**	**PAES**	**Proposed Method**
Average	0.10906	0.12571	0.06134	0.19530	0.0503
Best	0.06274	0.09505	0.04645	0.06556	0.0393
Worst	0.14373	0.14264	0.09914	0.49154	0.0996
Std. Dev	0.01592	0.01577	0.01284	0.01284	0.0120

6 Conclusion

PSO is an effective and easy-to-use tool for nonlinear optimization problems, this study introduces a novel approach called SC-PSO for solving multi-objective engineering problems. By utilizing the impedance and admittance circles on the Smith chart, the proposed method dynamically updates the particle locations, leading to effective determination of the local best particle. The performance of the SC-PSO algorithm is evaluated on three test functions with different characteristics, demonstrating successful convergence and providing a well-distributed solution set. The results highlight the potential of SC-PSO as an efficient approach for multi-objective optimization applications, showcasing its ability to explore solution spaces and identify optimal points in both convex and nonconvex problem domains. In future studies, we will apply it to a nonorthogonal multiple access network to optimize various parameters.

Declarations

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.

Yunus Dursun was supported by the Republic of T¨urkiye Ministry of National Education.

All authors contributed equally and approved the version of the manuscript to be published.The algorithms are available at https://github.com/YUNUSDURSUN/MultiObjectivePSO.git

References

[1] Babajamali, Z., khabaz, M.K., Aghadavoudi, F., Farhatnia, F., Eftekhari, S.A. and Toghraie, D.*Pareto multi-objective optimization of*

tandem cold rolling settings for reductions and inter stand tensions using NSGA-II ISA Trans. 130 (2022) 399–408.

[2] Bejarano, L.A., Espitia, H.E. and Montenegro, C.E.*Clustering analysis for the Pareto optimal front in multi-objective optimization*, Comput. 10(3) (2022), 37.

[3] Bin Mohd Zain, M.Z., Kanesan, J., Chuah, J.H., Dhanapal, S. and Kendall, G. *A multi-objective particle swarm optimization algorithm based on dynamic boundary search for constrained optimization,* Appl. Soft Comput. 70 (2018) 680–700.

[4] Chiu C.-C. and Lai, C.-M. *Multi-objective missile boat scheduling problem using an integrated approach of NSGA-II, MOEAD, and data envelopment analysis,* Appl. Soft Comput. 127 (2022), 109353.

[5] Coello, C.A.C. and Cortés, N.C. *Solving multiobjective optimization problems using an artificial immune system*, Genet. Program. Evolvable Mach. 6(2) (2005) 163–190.

[6] Dursun, Y. and Özkaya, U. *Çok Hedefli Parçacık Sürü Optimizasyonu için Smith Abağı Yaklaşımı.* Akıllı Sistemlerde Yenilikler Ve Uygulamaları Sempozyumu (2014), 187–192.

[7] Elbes, M., Alzubi, S., Kanan, T., Al-Fuqaha, A. and Hawashin, B.*A survey on particle swarm optimization with emphasis on engineering and network applications*, Evol. Intel. 12(2) (2019), 113–129.

[8] Elsheikh, A.H. and Abd Elaziz, M. *Review on applications of particle swarm optimization in solar energy systems* Int. J. Environ. Sci. Technol. 16(2) (2019) 1159–1170.

[9] Gad, A.G. *Particle swarm optimization algorithm and its applications: A systematic review*, Arch Computat Methods Eng. 29(5) (2022), 2531–2561.

[10] Ghorpade, S.N., Zennaro, M., Chaudhari, B.S., Saeed, R.A., Alhumyani, H. and Abdel-Khalek, S.*Enhanced differential crossover and quantum*

particle swarm optimization for IoT applications, IEEE Access, 9 (2021) 93831–93846.

[11] Gu, Q., Wang, Q., Chen, L., Li, X. and Li, X.*A dynamic neighborhood balancing-based multi-objective particle swarm optimization for multi-modal problems,* Expert Syst. Appl. 205 (2022) 117713.

[12] Kang, L., Chen, R.-S., Cao, W. and Chen, Y.-C. *Non-inertial opposition-based particle swarm optimization and its theoretical analysis for deep learning applications*, Appl. Soft Comput. 88 (2020), 106038.

[13] Li, Y., Zhang, Y. and Hu, W.*Adaptive multi-objective particle swarm optimization based on virtual Pareto front* Inf. Sci. 625 (2023), 206–236.

[14] Liang, J., Ge, S., Qu, B., Yu, K., Liu, F., Yang, H., Wei, P. and Li, Z. *Classified perturbation mutation-based particle swarm optimization algorithm for parameters extraction of photovoltaic models*, Energy Convers. Manag. 203 (2020), 112138.

[15] Loganathan S. and Arumugam, J. *Energy efficient clustering algorithm based on particle swarm optimization technique for wireless sensor networks,* Wireless Pers. Commun. 119(1) (2021), 815–843.

[16] Petchrompo, S., Coit, D.W., Brintrup, A., Wannakrairot, A. and Parlikad, A.K.*A review of Pareto pruning methods for multi-objective optimization,* Comput. Ind. Eng. 167 (2022), 108022.

[17] Petchrompo, S., Wannakrairot, A. and Parlikad, A.K.*Pruning Pareto optimal solutions for multi-objective portfolio asset management*, Eur. J. Oper. Res. 297(1). 203–220.

[18] Shami, T.M., El-Saleh, A.A., Alswaitti, M., Al-Tashi, Q., Summakieh, M.A., and Mirjalili, S. *Particle Swarm Optimization: A Comprehensive Survey* IEEE Access, 10 (2022), 10031–10061.

[19] Valencia-Rodrıguez D.C. and Coello Coello, C.A. *Influence of the number of connections between particles in the performance of a multi-objective particle swarm optimizer,* Swarm Evol. Comput. 77 (2023), 101231.

[20] Wang, D., Tan, D. and Liu, L. *Particle swarm optimization algorithm: An overview*, Soft Comput. 22(2) (2018), 387–408.

[21] Wei, B., Xia, X., Yu, F., Zhang, Y., Xu, X., Wu, H., Gui, L. and He, G. *Multiple adaptive strategies-based particle swarm optimization algorithm,* Swarm Evol. Comput. 57 (2020) 100731.

[22] Xu, Y., Zhang, H., Huang, L., Qu, R. and Nojima, Y.*A Pareto front grid guided multi-objective evolutionary algorithm* Appl. Soft Comput. 136 (2023) 110095.

[23] Zhang, Y. and Kong, X. *A particle swarm optimization algorithm with empirical balance strategy* Chaos, Solitons. Fractals, 10 (2023), 100089.