Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X Volume 4, Issue 3, 2026

SWARM OPTIMIZATION ALGORITHMS FOR SOLVING MULTI-OBJECTIVE PROBLEMS IN COMPLEX SYSTEMS

[1]Talha Nadeem, [2]Hood Khizer, [3]Huzaifah Bin Khawar, [4] Muhammad Zarar

[1] Systems Architect AI/ML, Research and Development Department, Trouve Labs, AHOY DMCC and The Superior University Lahore, Pakistan talhanadeem.ieee@gmail.com

[2] CEO Trouve Labs, CTO AHOY DMCC, Research and Development Department, Trouve Labs, AHOY DMCC, Dubai hoodkhizer@gmail.com

[3] AI Researcher, Research and Development Department, Trouve Labs, AHOY DMCC and The Superior University Lahore, Pakistan huzaifahkhawar.ieee@gmail.com

[4] AI Researcher, Research and Development Department, Trouve Labs, AHOY DMCC and The Superior University Lahore, Pakistan muhammadzarar.ieee@gmail.com

DOI:https://doi.org/10.5281/zenodo.24117456

Keywords
Swarm optimization, multi-objective optimization, complex systems, particle swarm optimization, ant colony optimization, firefly algorithm, Pareto dominance, decision-making.

Article History
Received: 10 Jan, 2026

Accepted: 18 March, 2026

Published: 26 March, 2026

Copyright @Author
Corresponding Author: *
Talha Nadeem*

Abstract

Multi-objective optimization in complex systems requires decision-makers to compare nondominated trade-offs rather than search for one universally best solution. This paper presents an evidence-bounded review and reproducible comparison framework for multi-objective particle swarm optimization, ant colony optimization and the firefly algorithm. The three families are analysed through their search mechanisms, archive management, constraint handling, convergence behaviour, diversity preservation and computational cost. Ten editable equations define the common optimization problem, Pareto dominance, method-specific updates, hypervolume and dominant complexity. A structured notation table and an IEEE/ACM-style algorithm specify how matched evaluation budgets, repeated runs and uncertainty reporting should be used. Because the source manuscript contains no raw run-level outputs, the revised paper does not claim new empirical superiority. Instead, it identifies defensible method-selection hypotheses and a transparent protocol for engineering design, logistics, resource allocation and environmental applications.

1. INTRODUCTION

Complex systems have several objectives which may not be optimized simultaneously, uncertainty, and interacting components which result in nonlinear responses. Engineering design could involve low cost, low mass and high reliability; logistics could involve short delivery time, low emissions and service equity; and resource systems could be about their efficiency, resilience and protection of environment. The former of those two factors can mask these conflicts, while multi-objective optimization provides a collection of nondominated alternatives to the decision makers.

A collection of basic agents whose local updates yield collective search is called a swarm optimization, in which candidate solutions are represented. The particle swarm optimization (PSO) algorithm relies on velocity and social learning [1] [2]; the ant colony optimization (ACO) algorithm [4] [5] depends on pheromone-assisted construction based on distance dependence; the firefly algorithm [7] relies on attraction due to distance and random motion. Multi-objective variants extend; selection and diversity control [3, 16] is extended by adding an external Pareto archive.

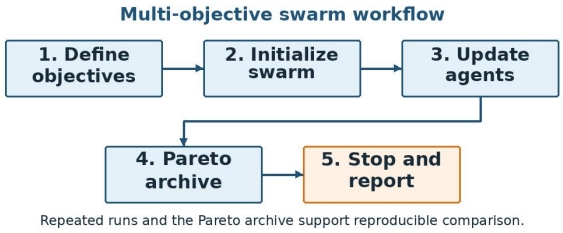


Figure 1. Common workflow for multi-objective swarm optimization and archive-based reporting.

The main methodological risk is that there is an unfair comparison. An apparent ranking may be reversed as a result of varying population sizes, budget evaluation criteria, stopping points, rules of constraints, reference points and/or random seed. In this revision, therefore, performance claims are regarded to be contingent on a documented protocol—not properties that apply to all complex systems.

1.1 Background and Motivation

A formal alternative to premature collapsing of objectives is Pareto dominance. A solution is nondominated if there is no possible solution that is at least as good in all objectives and strictly better in at least one objective. The front that incorporates the resulting compromises, thus, needs domain judgement and not automatic selection [8]-[12].

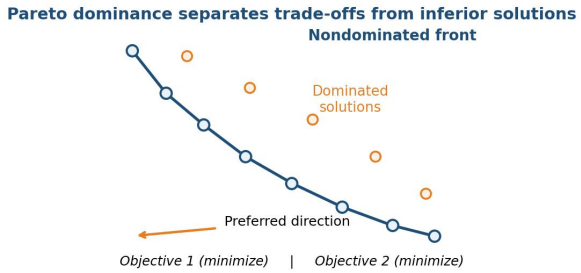


Figure 2. Pareto front, dominated alternatives and the preferred minimization direction.

1.2 Problem Statement

The swarms also need to explore new regions, exploit well-known regions, take care of multiple trade-offs and meet constraints. They are found to be geometry-dependent, problem-representation-dependent, parameter-schedule-dependent and archive-design-dependent. The paper thus enquires as to the formalisation of PSO, ACO and firefly mechanism under the same circumstances for complex multiobjective problems, and compares them.

1.3 Objectives of the Study

1.3.1 Formalize the common optimization problem

Set up with the same notation the goal, feasible domain, Pareto dominance, archive quality and computational cost.

1.3.2 Compare search mechanisms

Differentiate between velocity-based, pheromone-based and attraction-based updates but do not stigmatize one family system compared to the others.

1.3.3 Standardize evaluation

Incorporated matched evaluation budgets, repeated runs independently, identical runs and uncertainty-aware reporting.

1.3.4 Connect algorithms to complex systems

Apply properties of algorithms to continuous design problem, combinatorial routing problem, multimodal search and constrained allocation problems.

1.3.5 State limitations and responsible-use controls

Distinguish analytical from empirical evidence and ensure the objectives, constraints and decision oversight are transparent.

2. LITERATURE REVIEW

2.1 Multi-Objective Evolutionary Search

NSGA-II created quick nondominated sorting and diversity preservation via crowding [8]. However, PAES showed the importance of a bounded archive [10] and MOEA/D was used to break a multi-objective problem down into scalar subproblems [13]. Hypervolume-based methods directly link survival with the volume that is dominated in objective space [12], [14] and NSGA-III introduced a new concept of introducing reference direction selection to many objective spaces [15]. Even if the search operator is based on swarms these methods give guidelines on the higher principles.

2.2 Multi-Objective Particle Swarm Optimization

We say that the original PSO implementation updates velocity from inertia, personal experiences and social guidance [1], [2]. MOPSO is a way to replace one global best with leaders taken from a nondominated archive and generally add the control of grid, crowding or reference-direction [3, 16]. Of course, it would work best with continuous variables but convergence can be too early if one gets focused or velocity is not set correctly.

2.3 Multi-Objective Ant Colony Optimization

The Ant System and Ant Colony System build discrete solutions using both the intensity of the pheromone and the heuristic desirability [4] and [5]. A specific multi-objective ACO taxonomy is provided in [6] which presents a representation of how the objective-specific or aggregated pheromones and nondominated archives can organize these extensions. Attractive for routing, scheduling and assignment – pheromone stagnation and costly solution construction needs explicit control – their constructive bias.

2.4 Multi-Objective Firefly Optimization

Random exploration and moving less attractive agents to brighter agents is the strategy here, as a component of the firefly algorithm [7]. In multi-objective configurations, brightness should be computed based on dominance, decomposition or an archive-based guide. Although the method is easy to understand, and can search multimodal continuous spaces, the parameters for scale, absorption and randomization do depend on variable ranges and the iteration schedule.

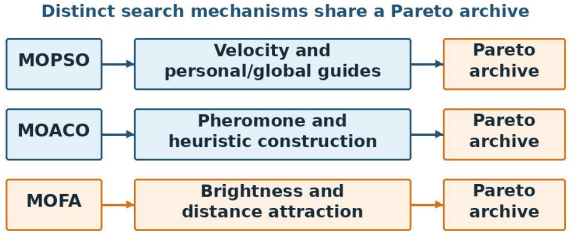


Figure 3. Comparison of MOPSO, MOACO and MOFA search mechanisms with a common Pareto archive.

2.5 Performance Assessment

There is no single indicator for a nondominated approximation. More precisely, Hypervolume represents convergence and spread from a reference point; generational distance indicates proximity; inverted generational distance does both proximity and coverage with a reliable reference set; and finally, spread or spacing indicators describes diversity [9], [11], [12]. Feasible-solution rate, objective evaluations, wall-clock time and peak memory should also be reported for the evaluation.

3. METHODOLOGY

3.1 Evidence and Comparison Scope

The paper applies a structured analytical comparison of the definitions of canonical algorithms and the well-known principles to evaluate performance. No new run-level experiments are presented, since supplied manuscript does not have any of the following: raw archives, seeds, parameter files or the logs of benchmarks. Statements of relative strength, therefore, become method-selection hypotheses and thus must be registered empirically in identical fashion.

3.2 Benchmark and Budget Control

Defensible experiments should contain continuous and combinatorial instances, constrained and unconstrained cases, various counts of objectives and even a number of dimensions of decisions. All algorithms have to be subjected to the same number of objective evaluations at the most. Populate 1 to 200 Archive Limit, Init Distribution, Termination Rule & Choice of Constraint Handler should be given. Avoid sharing if the stream of random seeds and the stream of independent seeds are different.

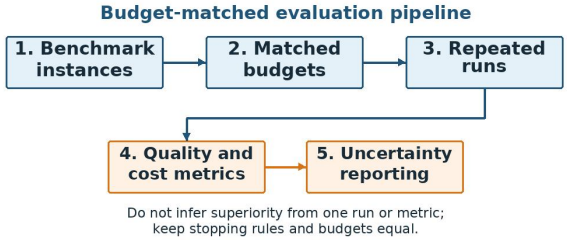


Figure 4. Budget-matched evaluation pipeline with repeated runs and uncertainty reporting.

3.3 Statistical and Reproducibility Requirements

The physical and statistical requirements must be met for this section.Requirements physical and statistical: required for this section.

Always keep the run level not reducing to one best result. Medians, interquartile ranges and confidence intervals should be used for skewed distributions. A Pairwise or Multiple-Algorithm test needs assumptions and effect size, as well as multiplicity control. In addition to the final archive, include failed runs and excluded results along with the code version and software environment and hardware.

3.4 Complex-System Applicability

The same algorithm can work differently at different stages of changes to variables, constraints or object-object evaluations. Vector updates may be the best option for continuous engineering design, constructive trails may be the best option for routing, and attraction and randomization may be the best options for multimodal landscapes. The biological metaphor is often less important than the problem representation and evaluation.

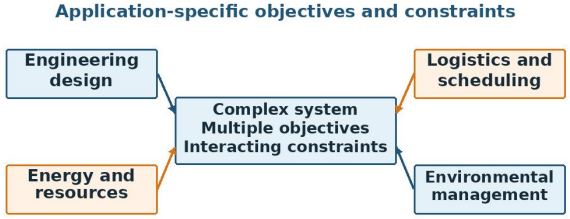


Figure 5. Complex-system application domains and their shared multi-objective structure.

4. MATHEMATICAL AND ALGORITHMIC SPECIFICATION

All the symbols introduced in (1)-(10) and in Algorithm 1 are defined in Table 1. The equations are HTML-formatted Word Math-equations that are split into the two article columns, but not beyond the width of a column.

Table 1. Consistent notation for multi-objective swarm optimization.

Symbol	Meaning	Dimension or range
x, d	Decision vector and dimension	x∈R^d
F, f_q	Objective vector and objective q	q=1,…,m
Ω	Feasible decision region	Constraint-defined set
≺	Pareto-dominance relation	Binary relation
x_i^t, v_i^t	PSO position and velocity	Vectors in R^d
p_i, g_i	Personal guide and archive leader	Feasible vectors
ω, c_1, c_2	Inertia, cognitive and social coefficients	Non-negative scalars
τ_{ij}, η_{ij}	Pheromone and heuristic desirability	Positive scalars
ρ	Pheromone evaporation rate	$0<\rho\leqslant 1$
β_0 , γ, α_t	Firefly attraction, absorption and randomization	Non-negative scalars
A	External nondominated archive	Finite solution set
HV, r^{ref}	Hypervolume and reference point	Objective-space values
T, N	Iteration/evaluation horizon and population size	Positive integers
C_F, C_U, C_A	Fitness, update and archive costs	Cost terms

4.1 Problem and Pareto Relations

$$\min_{x\in\Omega} F(x) = [f_1(x), f_2(x), \ldots, f_m(x)] \quad (1)$$

$$\Omega = \{x \in \mathbb{R}^d : g_j(x) \le 0,\ h_k(x) = 0\} \quad (2)$$

$$x \prec y \Leftrightarrow \left(\forall q,\ f_q(x) \le f_q(y)\right) \wedge \left(\exists r,\ f_r(x) < f_r(y)\right) \quad (3)$$

The following minimization problem is an m-objective optimization problem (1). The feasible region is defined by equation (2). Pareto dominance, according to Equation (3): x dominates y if x is not worse than y in all objectives; and x is at least better than y in one objective.

4.2 Method-Specific Update Relations

$$v_i^{t+1} = \omega v_i^t + c_1 r_1 (p_i - x_i^t) + c_2 r_2 (g_i - x_i^t) \quad (4)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (5)$$

$$p_{ij}^{(k)} = \frac{\tau_{ij}^{\alpha}\eta_{ij}^{\beta}}{\sum_{l\in N_i^{(k)}} \tau_{il}^{\alpha}\eta_{il}^{\beta}} \quad (6)$$

$$\tau_{ij}^{t+1} = (1-\rho)\tau_{ij}^t + \sum_{k=1}^{n_a} \Delta\, \tau_{ij}^{(k)} \quad (7)$$

$$x_i^{t+1} = x_i^t + \beta_0 e^{-\gamma r_{ij}^2}\left(x_j^t - x_i^t\right) + \alpha_t \varepsilon_i^t \quad (8)$$

The velocity and position updated equations for PSO are presented in equations (4) and (5). The transition probability, in the ACOs, is defined by Equation (6) and updating pheromone after burning is defined by Equation (7). The equation (8) provides the firefly attraction and a random term that depends on the distance. All the coefficients and schedules must be reported.

Algorithm 1. Reproducible Comparison of Multi-Objective Swarm Optimizers.

Require:	Problem instances P; algorithm set {MOPSO, MOACO, MOFA}; population N; budget T; runs R; indicators M.
Ensure:	Run-level records, final nondominated archives, uncertainty summaries and a reproducible comparison.
1	Define objectives F(x), constraints Ω, reference point and decision-variable bounds before optimization.
2	Fix common initialization, evaluation budget, stopping rules and independent random seeds.
3	For each problem instance p∈P and each algorithm a, execute R independent runs.
4	Initialize N feasible agents and an empty external archive A of nondominated solutions.
5	Update agents with the method-specific rule: Equations (4)-(5), (6)-(7), or (8).
6	Repair or penalize infeasible solutions using one predeclared constraint-handling rule.
7	Evaluate F(x), remove dominated archive members and apply diversity control when A exceeds its limit.
8	Stop only when the common evaluation budget T is exhausted; save convergence traces and final A.
9	Compute hypervolume, convergence, diversity, feasibility and computational-cost indicators.
10	Summarize medians, intervals and run-level distributions; perform multiplicity-aware tests if used.
11	Report parameters, seeds, software, hardware, failures and every excluded run.
12	Return archives and evidence without claiming universal superiority from one benchmark.

Algorithm 1 (continued)

4.3 Quality and Cost Relations

$$HV(A) = \lambda_m \quad \left(\bigcup_{x\in A}\left[F(x), r^{ref}\right]\right) \tag{9}$$

$$C_{total} = O \quad \left(TN\left(C_F + C_U + C_A\right)\right) \tag{10}$$

The hypervolume as compared to a specific reference is defined in equation (9). The dominant cost, over T iterations, N agents, objective evaluation, agent update and archiving maintenance, is expressed in equation (10). It's not a hardware timing statement, but an asymptotic one.

Table 2. Dominant computational terms for the comparison framework.

Method	Principal operation	Indicative time term	Memory
MOPSO	Velocity/position update and leader selection	O(TN(d+C_F+C_A))	O(Nd+|Ald)
MOACO	Construct solutions and update pheromone	O(TN(C_construct+C_F+C_A))	Problem dependent
MOFA	Attraction comparisons and movement	$O(TN^2(d+C_F)+TN\ C_A)$	O(Nd+|Ald)
Archive	Dominance, truncation and diversity control	Implementation dependent	O(|Ald)

The terms are purposely indicative since both objective evaluation and archive maintenance can be significant parts of the update. Measured evaluation, wall-clock, hardware and implementation details should be included in the final reports.

Table 3. Minimum reporting protocol for a controlled swarm-optimizer comparison.

Component	Required specification	Control
Problem definition	Objectives, bounds, constraints, variable types	Same encoded instance for every algorithm
Search budget	Maximum objective evaluations and stopping rule	Matched budget; no algorithm-specific early advantage
Initialization	Population/colony size and seed set	Independent runs from disclosed seed schedule
Archive	Limit, dominance, truncation and reference point	Same quality definition across methods
Indicators	HV, distance, diversity, feasibility and cost	Report distributions and uncertainty
Reproducibility	Parameters, code, software, hardware and failures	Release run-level logs and final archives

Table 4. Qualitative method-selection hypotheses requiring matched validation.

Family	Structural strength	Principal risk	Plausible starting domain

Family	Structural strength	Principal risk	Plausible starting domain
MOPSO	Continuous vector search; inexpensive updates	Leader concentration; velocity sensitivity	Continuous design and allocation
MOACO	Constructive search; explicit component memory	Pheromone stagnation; construction cost	Routing, assignment and scheduling
MOFA	Attraction plus stochastic multimodal exploration	Scale and parameter-schedule sensitivity	Continuous multimodal optimization

5. COMPARATIVE ANALYSIS

5.1 Evidence-Bounded Findings

MOPSO can directly move towards a specific vector and is generally very easy to implement for bounded continuous decision spaces. It has a co-mandering of its archive leaders that affect the whole swarm, thus diversity control is essential. MOACO can naturally be used in routing, assignment and scheduling as a natural hypothesis for incremental solution construction. MOFA is attractive and stochastic exploration, and is a possible model for continuous multimodal landscapes. These are affinities, they are not universal rankings.

It can be concluded in a way which is well-sustained that representation and evaluation design are the key to choosing an algorithm. This method may work well on a high hypervolume in a low dimensional task, but it is unsuitable on a more restricted, costly or many objective optimization task. On the other hand, a slower method may be more appropriate when considerations are the feasibility or interpretability of the construction process.

5.2 Convergence, Diversity and Cost

If only a small area is archived, then it is not enough that it has converged quickly in the beginning. These distance and diversity indicators should therefore be coupled by hypervolume [9], [11], [12], [14].. Convergence traces should be picked by objective assessments along the horizontal axis, with the algorithms being compared under one budget, without any compromise freeing up wall clock time.

5.3 Constraint Handling

When comparing in a feasibility first way, one of the following approaches must be declared: repair, penalty, stochastic ranking, or epsilon constraint or dominance rules that favour feasible solutions. The changing of the constraint handler across algorithms is not a desirable feature, since the comparison of the search mechanisms will then be confounded. Reports should report feasible-run rate (not ignore failed runs), and constraint violation and objective quality.

5.4 Method-Selection Hypotheses

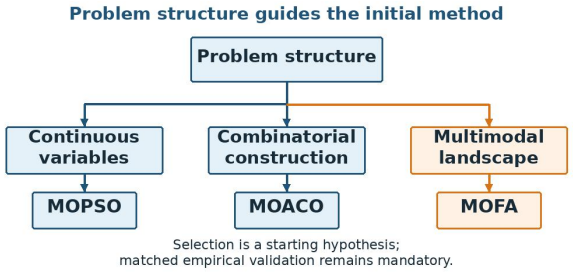


Figure 6. Problem-structure hypotheses for choosing an initial swarm-optimization family.

The starting shows in Figure 6 is a guide. If a weakness is diagnosed, and you wish to improve it in a specific local area, for example, local refinement, archive diversity or discrete-continuous coupling, may be justifiable; however, the more operators which are added, the more difficult tuning and assignment becomes. Studies of ablation need

to isolate whether the new mechanism in itself does affect the desired process.

6. DISCUSSION

6.1 Implications for Complex Systems

Swarm optimization can be helpful in the scenarios where gradients cannot be obtained, there is a conflict of objectives, and a population of alternatives is useful. It does not replace the need to establish objectives, to test constraints or to select one of the ultimate options; it is simply a tool for conducting a trade-off procedure. Mathematically nondominated solutions may still be unacceptable in safety-critical engineering given standards, uncertainty or stakeholder requirements.

6.2 Responsible Decision Support

Objective definitions contain values. Scale and thresholds of costs, emissions, reliability and service quality and equity can not be considered unalterable metrics due to influence on stakeholders. Constraints should be validated by the analyst and trade-offs documented outside the model, and the person doing the validation should be noted. To deploy, human oversight is necessary prior to deployment, especially for optimization measures that impact infrastructure, public services or environmental resources.

6.3 Limitations

This paper is NOT a new benchmark study, it is a review and a protocol. Does not provide evidence at run level arguments to claims about faster convergence/convergent algorithms with better fronts. Implementation constants and problem-specific objective costs are not included as part of the complexity parameters. Depending on the different encoding typologies, constraints, number of objects to be found, noise, and computational platform, method-selection hypotheses may change.

6.4 Future Research

Future research directions should focus on preregistering benchmark suites, adding in expensive and dynamic objectives, testing many-objective archive strategies, reporting on the energy cost and memory size, and testing variants using robust or surrogate criteria. Problem definitions, source code, seeds and final archive along with all run-level indicators should be released as Reproducible studies.

6.5 Application-Specific Validation

All of the engineering design objectives and constraints must relate to a validated simulation, measurement protocol or governing standard. The drawback to using surrogate models could be the reduction in costly evaluations, but the prediction error of these surrogate models will need to be added when comparing to the original model and reviewed at regular intervals. Even if a front came from a supposedly "hyper-volume" simulation, it would not count as decision evidence.

There are encodings that can preserve route and precedence feasibility, capacity feasibility, and time window feasibility that need to be created for the logistics and scheduling issues. MOACO can take advantage of constructive feasibility (dominating or swamping the opponent), while continuing (may require decoding or repair). Applications involving energy and environmental additionally demand scenario analysis since the energy and environmental demands and ecological responses are still uncertain. The robustness of results needs to be reported separately from nominal objective quality.

6.6 From Pareto Archive to Decision

The external archive does not reflect final decision, it is an intermediate result. Objective ranges, analysis of knee regions, analysis of constraint areas, and sensitivity to normalization/reference points should be evaluated by the decision maker. The preference info could come before or after search, or during search, as long as it is or is not described where in the process and how it affects computational search.

During handover, each solution selected should have a decision vector, the optimal objective values, information about feasibility of the solution, information about its provenance, and parameter configuration and uncertainty information. A few easily identifiable compromises generally are more defensible than selecting the "center" of a plot by eye. The archive should be reevaluated rather than reused without evaluation and without regard to the changing preference or constraint.

7. CONCLUSION

The paper is revised and offers a two-column representation where multi-objectives swarm algorithms can be dynamically compared in complex systems. All three search methods (MOPSO, MOACO and MOFA) need a robust basis that doesn't rely on random selection or random naming like "hugely popular" and "totally unpopular": clear goals and limits, a nondominated archive, comparable budgets, multiple runs, multiple measures of success, and the production of clear uncertainty reports. Technically auditable, with ten editable equations, consistent notation table and a standard algorithm box and pure-white comparison tables throughout. The key message isn't indicative, it's conditional: The choice of algorithms should correspond to the structure of problems, and must be supported by measured evidence.

REFERENCES

. J. Kennedy and R. Eberhart, "Particle swarm optimization," in Proc. IEEE Int. Conf. Neural Networks, 1995, pp. 1942-1948.

. Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in Proc. IEEE Int. Conf. Evolutionary Computation, 1998, pp. 69-73.

. C. A. Coello Coello and M. S. Lechuga, "MOPSO: A proposal for multiple objective particle swarm optimization," in Proc. IEEE Congress on Evolutionary Computation, 2002, pp. 1051-1056.

. M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: Optimization by a colony of cooperating agents," IEEE Trans. Syst., Man, Cybern. B, vol. 26, no. 1, pp. 29-41, 1996.

. M. Dorigo and L. M. Gambardella, "Ant colony system: A cooperative learning approach to the traveling salesman problem," IEEE Trans. Evol. Comput., vol. 1, no. 1, pp. 53-66, 1997.

. O. García-Martínez, C. García-Martínez, and F. Herrera, "A taxonomy and an empirical analysis of multiple objective ant colony optimization algorithms for the bi-criteria TSP," Eur. J. Oper. Res., vol. 180, no. 1, pp. 116-148, 2007.

. X.-S. Yang, "Firefly algorithms for multimodal optimization," in Stochastic Algorithms: Foundations and Applications, LNCS 5792, 2009, pp. 169-178.

. 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, 2002.

9. E. Zitzler and L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach," IEEE Trans. Evol. Comput., vol. 3, no. 4, pp. 257-271, 1999.
10. J. D. Knowles and D. W. Corne, "Approximating the nondominated front using the Pareto archived evolution strategy," Evol. Comput., vol. 8, no. 2, pp. 149-172, 2000.
11. E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. da Fonseca, "Performance assessment of multiobjective optimizers: An analysis and review," IEEE Trans. Evol. Comput., vol. 7, no. 2, pp. 117-132, 2003.
12. J. Bader and E. Zitzler, "HypE: An algorithm for fast hypervolume-based many-objective optimization," Evol. Comput., vol. 19, no. 1, pp. 45-76, 2011.
3. Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," IEEE Trans. Evol. Comput., vol. 11, no. 6, pp. 712-731, 2007.
4. N. Beume, B. Naujoks, and M. Emmerich, "SMS-EMOA: Multiobjective selection based on dominated hypervolume," Eur. J. Oper. Res., vol. 181, no. 3, pp. 1653-1669, 2007.
5. K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints," IEEE Trans. Evol. Comput., vol. 18, no. 4, pp. 577-601, 2014.
6. M. Reyes-Sierra and C. A. Coello Coello, "Multi-objective particle swarm optimizers: A survey of the state-of-the-art," Int. J. Comput. Intell. Res., vol. 2, no. 3, pp. 287-308, 2006.