



Smart Starts: Accelerating Convergence through Uncommon Region Exploration

Xinyu Zhang
Stony Brook University
Stony Brook, New York, USA
zhang146@cs.stonybrook.edu

Mário Antunes†
University of Aveiro
Aveiro, Portugal
mario.antunes@ua.pt

Tyler Estro
Stony Brook University
Stony Brook, New York, USA
testro@cs.stonybrook.edu

Erez Zadok
Stony Brook University
Stony Brook, New York, USA
ezk@cs.stonybrook.edu

Klaus Mueller
Stony Brook University
Stony Brook, New York, USA
mueller@cs.stonybrook.edu

ABSTRACT

Initialization profoundly affects evolutionary algorithm (EA) efficacy by dictating search trajectories and convergence. This study introduces a hybrid initialization strategy combining empty-space search algorithm (ESA) and opposition-based learning (OBL). OBL initially generates a diverse population, subsequently augmented by ESA, which identifies under-explored regions. This synergy enhances population diversity, accelerates convergence, and improves EA performance on complex, high-dimensional optimization problems. Benchmark results demonstrate the proposed method's superiority in solution quality and convergence speed compared to conventional initialization techniques.

CCS CONCEPTS

• **Theory of computation** → *Theory of randomized search heuristics; Random search heuristics.*

KEYWORDS

Evolutionary algorithms, initialization, opposition-based learning, empty-space search

ACM Reference Format:

Xinyu Zhang, Mário Antunes†, Tyler Estro, Erez Zadok, and Klaus Mueller. 2025. Smart Starts: Accelerating Convergence through Uncommon Region Exploration. In *Genetic and Evolutionary Computation Conference (GECCO '25 Companion)*, July 14–18, 2025, Malaga, Spain. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3712255.3726720>

1 INTRODUCTION

Evolutionary algorithms (EAs) are essential for complex, high-dimensional optimization across diverse domains. Their population-based search, simulating natural evolution, offers robust global exploration, particularly for multimodal and non-convex problems [4].

† Corresponding author.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
GECCO '25 Companion, July 14–18, 2025, Malaga, Spain
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-1464-1/2025/07
<https://doi.org/10.1145/3712255.3726720>

However, initial population distribution critically impacts EA performance in intricate landscapes. Concentrated populations risk premature convergence, whereas diverse populations enhance search space coverage [1]. Effective initialization is thus crucial, especially for rugged, multi-modal fitness landscapes.

Traditional uniform random initialization often fails in high-dimensional spaces. Systematic methods like Latin Hypercube Sampling (LHS) [19] and Sobol sequences [14] improve distribution but suffer from the “curse of dimensionality”. To address this, researchers explore under-explored region identification, including niching methods for multimodal problems [13, 12, 15]. Kazimipour *et al.* categorized initialization methods by randomness, compositionality, and generality [7]. Recent studies include orthogonal array-based initialization [8] and interval-based multi-objective initialization [18].

Compositional methods, like opposition-based learning (OBL) [9], enhance diversity by generating “opposite” solutions. However, OBL may miss under-explored subregions in complex landscapes. The Empty-space Search Algorithm (ESA) [20], designed for data visualization, can complement OBL by targeting these regions.

This paper proposes a novel initialization strategy combining OBL and ESA. OBL generates an initial diverse population, subsequently augmented by ESA to fill under-explored areas. This hybrid approach aims to accelerate convergence and enhance EA performance in complex, high-dimensional optimization. Experimental results on benchmark problems, compared with conventional techniques, validate the method's efficacy.

The paper is structured as follows: Section 2 reviews relevant technologies; Section 3 details the proposed method; Section 4 presents experimental results and statistical analysis; and Section 5 summarizes the findings.

2 PRELIMINARIES

This section lays the groundwork for the proposed methodology by outlining two fundamental concepts: Opposition-Based Learning (OBL) and Empty-space Search Algorithm (ESA) in Sections 2.1 and 2.2 respectively.

2.1 Opposition-Based Learning

Opposition-Based Learning (OBL) [9] accelerates optimization by simultaneously evaluating candidate solutions x and their opposites x' . For $x \in R^d$ within a bounded search space $S = [a_1, b_1] \times [a_2, b_2] \times$

$\dots \times [a_d, b_d]$, the opposite x' is defined as:

$$x'_i = a_i + b_i - x_i, \forall i \in 1, 2, \dots, d$$

where a_i and b_i are the i -th dimension's bounds. The superior solution, based on the objective function f , is selected, effectively doubling exploration without increasing population size.

2.2 Empty-space Search Algorithm

The Empty-space Search Algorithm (ESA) [20] is a heuristic method designed to identify sparse, under-explored regions ("empty spaces") within high-dimensional search spaces. While Zhang *et al.* applied ESA to identify novel high quality solutions, we find that locating these regions can enhance candidate solution diversity and quality in optimization.

In a data space, ESA first randomly places some points called "agents" and then employs a physics-based approach called Lennard-Jones (L-J) Potential to guide the agents to sparse regions. The L-J Potential models the interaction between particles, but here it is used to determine the moving direction of agents. The force of a single data point to an agent is given below:

$$F(r) = 24 \frac{1}{\sigma} \left[2 \left(\frac{\sigma}{r} \right)^{13} - \left(\frac{\sigma}{r} \right)^7 \right] \quad (1)$$

where r is the distance of an agent from a data point and σ is the particle effect size. The force is positive if r is smaller than σ , which means that the agent is too close to a data point and will be pushed away, and otherwise the force is negative and pulls the agent back. The final moving direction of the agent is the resultant force direction of k nearest data points:

$$\Sigma \vec{F} = \sum_i^k \vec{u}_i F(r_i) \quad \vec{d} = \frac{\Sigma \vec{F}}{\|\Sigma \vec{F}\|} \quad (2)$$

where \vec{u}_i is the unit vector that directs the agent to the data point. Through iterative equilibrium, the agents converge to positions representing the centers of empty spaces.

3 PROPOSED METHOD

This section presents OBLESA (OBL+ESA), a novel initialization method designed to enhance EA exploration in complex landscapes. OBLESA addresses the limitation of random or biased initialization by identifying and populating under-explored regions.

Initially, OBL is applied to a random population, generating opposite solutions to expand diversity. Subsequently, ESA, utilizing Lennard-Jones-like potential functions, identifies and populates sparse regions, avoiding dense clusters. The candidate set is temporarily tripled to ensure thorough exploration, then reduced to the top one-third.

OBLESA delivers a higher-quality, diverse initial population, facilitating early exploitation of promising solutions and mitigating premature convergence.

3.1 ESA for Population Initialization

Zhang *et al.* [20] designed ESA for Pareto front expansion, employing momentum for trajectory-based sampling. This work adapts ESA to optimize population distribution within empty regions.

Instead of trajectory sampling, we place representative points within each region, avoiding oversampling. We also eliminate momentum, but selecting converged agent positions instead. Furthermore, we replace k -nearest neighbor (kNN) with approximate nearest neighbor (ANN) to mitigate high-dimensional query bottlenecks. Algorithm 1 details this simplified ESA variant.

Algorithm 1: Empty-Space Search for a Single Agent

```

Set the number of neighbors  $k$ , the particle effective
diameter  $\sigma$ , the number of search steps  $n$ , the step size  $\alpha$ ,
the vanishing threshold  $\delta$ ;
Initialize an agent  $\pi = \mathbf{c}$  where  $\mathbf{c}$  is a random coordinate;
Specify constraints on the target function:
 $f_1(\pi) \leq 0; f_2(\pi) \leq 0; \dots, f_p(\pi) \leq 0;$ 
for  $i \dots n$  do
    Get  $k$  approximated nearest neighbors of the agent from
    the dataset;
    Use Eq. 2 to calculate  $\vec{d}$ ;
    if  $\|\Sigma \vec{F}\| < \delta$  then
        | return  $\tau$ ;
    end
     $\pi = \pi + \vec{d} * \alpha$ ;
    if  $\pi$  violates any constraint  $f$  then
        | return  $\pi$ ;
    end
end
return  $\pi$ ;

```

There are three key parameters in this algorithm that affect the agent convergence: the number of neighbors k , the particle effective diameter σ , and the step size α . Zhang *et al.* recommended setting $k = d + 1$ where d is the dimensionality of the problem, $\sigma = \text{average distance to the neighbors}$, and $\alpha = 0.01$ in their work. In our experiments, we generally adopted their recommendations, but empirically reduced σ to half of average distance to the neighbors to prevent agents from moving out of the bounding box.

3.2 Initialization Strategy

High-dimensional empty region counts increase exponentially, a consequence of the *curse of dimensionality* [2]. Delaunay Triangulation (DT) reveals a space complexity of $O(n^{\lceil d/2 \rceil})$ [10], demonstrating the significant complexity growth with dimension and data size. Thus, thorough space exploration is crucial. Given a population size of n_{pop} , we summarize our algorithm in four steps: i) Random population initialization (n_{pop}); ii) Opposition point generation; iii) ESA agent deployment and empty-space population augmentation; and iv) Population refinement to n_{pop} .

Random initialization, followed by OBL, ensures broad initial coverage. ESA then refines exploration, targeting under-explored regions. The population, tripled post-ESA, is evaluated, and the top one-third is retained for subsequent optimization.

4 EVALUATION

This section presents a comprehensive evaluation of the proposed methodology. We first detail the experimental setup (subsection 4.1),

including the benchmark functions, parameter settings, and evaluation metrics employed. Subsequently, we analyze the results obtained (subsection 4.2), comparing the performance of the proposed approach against established baseline algorithms and discussing the key observations and insights gained from the experiments.

4.1 Evaluation Settings

In this section, we present the evaluation results of OBLESA against two baselines: random initialization and random + OBL initialization, using 24 benchmark functions. Specifically, we first selected target functions from COCO benchmark [5] and generated the initial population using each method, setting the initial population size to $n_{pop} = 100$. The generated initial population was then used as input for two evolutionary algorithms: Differential Evolution (DE) [16] and Enhanced Grey Wolf Optimization (EGWO) [6, 11] and tested their performance.

When evaluating initialization strategies, employing both DE and EGWO offers a robust approach due to their distinct characteristics. DE, a well-established evolutionary algorithm, provides a solid baseline, making it a reliable tool for assessing fundamental performance. Conversely, EGWO, representing a state-of-the-art metaheuristic, allows for the examination of how advanced optimization techniques respond to varying initial conditions.

Each function in COCO benchmark has an internal convergence threshold, meaning that the optimization process stops as soon as the threshold is reached. Additionally, each function has multiple variants (e.g., rotations) and default budget. The optimization process terminates immediately either upon reaching the convergence threshold or when the budget is exhausted. To ensure consistency, we left all default settings in the COCO benchmark, including the internal convergence threshold, budget, and problem dimensionality (2D, 3D, 5D, 10D, 20D, and 40D). We measured performance by calculating the fraction of benchmark functions that reached the convergence threshold during the optimization process.

To account for the randomness inherent in evolutionary algorithms, we ran each initialization strategy 10 times with different random seeds (1 - 10).

4.2 Result Analysis

We put our results in the supplementary material ¹ where we observe that the performance of both algorithms degrades as dimensionality increases. However, EGWO consistently outperforms DE across all dimensions. In addition, OBLESA initialization is better than the other two baselines in most cases for both EGWO and DE across the six dimensionalities.

We also notice that the advantage lead by OBLESA is mostly small. To assess the impact of randomness, we aggregated the results of 10 seeds and performed a statistical analysis to test the significance of the difference between OBLESA and the baselines. Specifically, we assigned scores to each initialization strategy based on their rank per dimensionality per seed. For instance, if OBLESA ranked first and Random ranked third, OBLESA received 3 points while Random received 1. Next we summed the scores over 10 seeds for each dimensionality. Then we did ANOVA and post-hoc test

to analyze whether the differences are statistically significant. The results are listed in Tab. 1 and Tab. 2.

Overall, OBLESA generally achieved higher scores than the baselines. The ANOVA and the associated post-hoc test show that OBLESA is similar to baselines, but it leads significant advantage in higher dimensional space. As expected given the nature of ESA, it should provide greater benefits the larger the problem dimensionality.

In our experiments, we also find that OBLESA does not show a noticeable advantage against the baselines. Some possible factors could be (1) the refinement step: we simply evaluated the entire augmented population and chose the top one third after OBL and ESA. It may lead to some mode collapse since the landscape of a target function in high-dimensional space is complex. Simply choosing the best data probably traps the algorithm to local optima. In the follow-up research, we will try different ways to refine the population, e.g., assign a probability to reject high-fitness data points or pair a random point with its associated opposite point and empty-space agent, choose the best one among the three candidates in each pair; (2) the initial population size: in our experiments, we simply set $n_{pop} = 100$ to save time and computational resources. However, due to counterintuitive facts in high-dimensional space [3, 17], the empty space will be much more complex. Increasing the population size can probably improve performance; (3) parameter fine-tuning: we adopted most parameters recommended by Zhang *et al.* in their work, but they aim to find Pareto optimal configurations. The goal differs from our work. Therefore, further investigation is necessary about the parameter setting.

In further research, we will work more on the aspects mentioned above and look deeply into the algorithm mechanism to improve OBLESA's performance across the benchmarks. The algorithm implementation and experiment settings are also available in the supplementary material.

5 CONCLUSION

The experimental results highlight the effectiveness of OBLESA in improving population diversity and accelerating convergence in evolutionary algorithms. By integrating Opposition-Based Learning (OBL) with the Empty-Space Search Algorithm (ESA), OBLESA addresses key limitations of existing initialization strategies in high-dimensional problems. Traditional random sampling ensures broad coverage of the search space but often leaves large unexplored regions, while OBL enhances exploration by generating complementary solutions yet lacks an explicit mechanism to target under-sampled areas. ESA bridges this gap by actively identifying and filling the sparse regions, leading to a more balanced and representative initial population.

Our results demonstrate that OBLESA significantly improves optimization performance, particularly in high-dimensional spaces (20D and 40D). Statistical analysis confirms that these improvements are not incidental, as OBLESA consistently outperforms baselines in these challenging settings. However, in lower-dimensional problems, the advantage of OBLESA is less pronounced, suggesting that its benefits become increasingly relevant as the complexity of the search space grows.

¹The supplementary material is available at github.com/Lagrange/SmartStart.

Table 1: Advantage score of each initialization strategy on EGWO. The p-value shows the statistical significance of the difference calculated by ANOVA. OBLESA-OBL and OBLESA-RANDOM are the post-hoc analysis results. Each entry in the second table is the p-value to analyze the significance of pairwise difference.

Strategy	2D	3D	5D	10D	20D	40D
OBLESA	21	23	17	24	27	29
OBL	20	20	23	17	21	16
RANDOM	19	17	20	19	12	15
p-vale	0.873	0.280	0.280	0.153	< 0.0001	< 0.0001
post-hoc	2D	3D	5D	10D	20D	40D
OBLESA - OBL	0.963	0.696	0.249	0.144	0.062	< 0.0001
OBLESA -RANDOM	0.861	0.249	0.696	0.359	< 0.0001	< 0.0001

Table 2: Advantage score of each initialization strategy on DE. The p-value shows the statistical significance of the difference calculated by ANOVA. OBLESA-OBL and OBLESA-RANDOM are the post-hoc analysis results. Each entry in the second table is the p-value to analyze the significance of pairwise difference.

Strategy	2D	3D	5D	10D	20D	40D
OBLESA	19	23	26	28	23	29
OBL	22	21	18	18	19	18
RANDOM	19	16	16	14	18	13
p-vale	0.663	0.153	0.119	< 0.0001	0.375	< 0.0001
post-hoc	2D	3D	5D	10D	20D	40D
OBLESA - OBL	0.711	0.844	0.053	0.002	0.536	< 0.0001
OBLESA -RANDOM	1	0.144	0.013	< 0.0001	0.382	< 0.0001

ACKNOWLEDGMENTS

This work was supported by FCT - Fundação para a Ciência e Tecnologia, I.P. by project reference UIDB/50008, and DOI identifier 10.54499/UIDB/50008.

REFERENCES

- [1] Jeffrey O. Agushaka, Absalom E. Ezugwu, Laith Abualigah, Samaher Khalaf Alharbi, and Hamiden Abd El-Wahed Khalifa. 2022. Efficient initialization methods for population-based metaheuristic algorithms: a comparative study. *Archives of Computational Methods in Engineering*, 30, 3, (Dec. 2022), 1727–1787. doi: 10.1007/s11831-022-09850-4.
- [2] Naomi Altman and Martin Krzywinski. 2018. The curse(s) of dimensionality. *Nature Methods*, 15, 6, (May 2018), 399–400. doi: 10.1038/s41592-018-0019-x.
- [3] Jack Anderson, Cristian Cobeli, and Alexandru Zaharescu. 2024. Counterintuitive patterns on angles and distances between lattice points in high dimensional hypercubes. *Results in Mathematics*, 79, 2, 94.
- [4] Zahra Beheshti and Siti Mariyam Hj Shamsuddin. 2013. A review of population-based meta-heuristic algorithms. *Int. j. adv. soft comput. appl.*, 5, 1, 1–35.
- [5] N. Hansen, A. Auger, R. Ros, O. Mersmann, T. Tušar, and D. Brockhoff. 2021. COCO: a platform for comparing continuous optimizers in a black-box setting. *Optimization Methods and Software*, 36, 114–144, 1. doi: https://doi.org/10.1080/10556788.2020.1808977.
- [6] Himani Joshi and Sankalp Arora. 2017. Enhanced grey wolf optimization algorithm for global optimization. *Fundamenta Informaticae*, 153, 3, (June 2017), 235–264. doi: 10.3233/fi-2017-1539.
- [7] Borhan Kazimipour, Xiaodong Li, and A. K. Qin. 2014. A review of population initialization techniques for evolutionary algorithms. In *2014 IEEE Congress on Evolutionary Computation (CEC)*, 2585–2592. doi: 10.1109/CEC.2014.6900618.
- [8] Abhishek Kumar, Partha P. Biswas, and Ponnuthurai N. Suganthan. 2022. Differential evolution with orthogonal array-based initialization and a novel selection strategy. *Swarm and Evolutionary Computation*, 68, 101010. doi: http://doi.org/10.1016/j.swevo.2021.101010.
- [9] Sedigheh Mahdavi, Shahryar Rahnamayan, and Kalyanmoy Deb. 2018. Opposition based learning: a literature review. *Swarm and Evolutionary Computation*, 39, 1–23. doi: https://doi.org/10.1016/j.swevo.2017.09.010.
- [10] Raimund Seidel. 1995. The upper bound theorem for polytopes: an easy proof of its asymptotic version. *Computational Geometry*, 5, 2, 115–116.
- [11] Isha Sharma, Vijay Kumar, and Sanjeevani Sharma. 2022. A comprehensive survey on grey wolf optimization. *Recent Advances in Computer Science and Communications (Formerly: Recent Patents on Computer Science)*, 15, 3, 323–333.
- [12] Ofer M Shir, Michael Emmerich, and Thomas Bäck. 2010. Adaptive niche radii and niche shapes approaches for niching with the cma-es. *Evolutionary computation*, 18, 1, 97–126.
- [13] Grzegorz Rozenberg, Thomas Bäck, and Joost N. Kok, (Eds.) 2012. *Niching in evolutionary algorithms. Handbook of Natural Computing*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1035–1069. ISBN: 978-3-540-92910-9. doi: 10.1007/978-3-540-92910-9_32.
- [14] Ilya Meierovich Sobol'. 1967. On the distribution of points in a cube and the approximate evaluation of integrals. *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, 7, 4, 784–802.
- [15] Catalin Liviu Stoean, Mike Preuss, Ruxandra Stoean, and Dumitru Dumitrescu. 2007. Disburdening the species conservation evolutionary algorithm of arguing with radii. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, 1420–1427.
- [16] Rainer Storn and Kenneth Price. 1997. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11, 341–359.
- [17] Michel Verleysen and Damien François. 2005. The curse of dimensionality in data mining and time series prediction. In *International work-conference on artificial neural networks*. Springer, 758–770.
- [18] Yu Xue, Xu Cai, and Ferrante Neri. 2022. A multi-objective evolutionary algorithm with interval based initialization and self-adaptive crossover operator for large-scale feature selection in classification. *Applied Soft Computing*, 127, 109420. doi: https://doi.org/10.1016/j.asoc.2022.109420.
- [19] Kenny Q Ye. 1998. Orthogonal column latin hypercubes and their application in computer experiments. *Journal of the American Statistical Association*, 93, 444, 1430–1439.
- [20] Xinyu Zhang, Tyler Estro, Geoff Kuenning, Erez Zadok, and Klaus Mueller. 2025. Into the void: mapping the unseen gaps in high dimensional data. *arXiv preprint arXiv:2501.15273*.