

SyVCO: A Synergic and Vigilance Cuculidae Optimization Algorithm for Global Numerical Optimization

¹ Shrijeet Pagrut, ²Pradip Jawandhiya

¹Department of Computer Science and Engineering, Pankaj Laddhad Institute of Technology and Management Studies, Yelgaon, Buldhana, Maharashtra, India

²Department of Computer Science and Engineering, P R Pote Patil College of Engineering & Management, Amravati, Maharashtra, India

DOI: 10.5281/zenodo.23256179

ABSTRACT

Hybrid metaheuristics that combine complementary search behaviours from two or more nature-inspired algorithms have shown consistent gains over their parent algorithms on global optimization tasks. This paper isolates and formally validates the Synergic and Vigilance Cuculidae Optimization (SyVCO) algorithm, a hybrid metaheuristic that merges the producer–scrounger foraging dynamics of the Sparrow Search Algorithm (SSA) with the Lévy-flight-driven exploration of Cuckoo Search (CS), and augments both with a teaching-factor exploitation step and a historical-memory-based adaptation phase. Unlike prior work that embedded SyVCO inside a domain-specific deep learning pipeline, this paper evaluates the algorithm on its own merits against ten standard unimodal and multimodal benchmark functions, benchmarked against its two parent algorithms (SSA, CS) and three widely used baselines (PSO, GA, GWO) over 15 independent runs each. SyVCO converges to the global optimum (or within numerical precision of it) on 8 of 10 benchmark functions, statistically matches or exceeds SSA on 7 of 10 functions (Wilcoxon rank-sum, $p < 0.05$), and significantly outperforms PSO, GA, and CS on 9 of 10 functions. The results confirm that the hybridization strategy retains the exploration strength of Lévy flights while inheriting SSA's fast local exploitation, at a computational overhead of only 1-2 ms per iteration over the parent algorithms.

Keywords—metaheuristic optimization; sparrow search algorithm; cuckoo search; hybrid optimization; Lévy flight; benchmark functions

I. INTRODUCTION

Metaheuristic optimization algorithms are widely used to tune parameters of complex systems where gradient information is unavailable or unreliable, including deep learning hyperparameters, engineering design variables, and control system gains. Population-based, nature-inspired metaheuristics such as Particle Swarm Optimization (PSO), Genetic Algorithms (GA), the Sparrow Search Algorithm (SSA), and Cuckoo Search (CS) each encode a different balance between exploration (global search) and exploitation (local refinement). The No Free Lunch theorem for optimization establishes that no single algorithm dominates across all problem classes, which motivates the continued development of hybrid algorithms that combine complementary search operators to improve robustness across a wider range of landscapes.

The Synergic and Vigilance Cuculidae Optimization (SyVCO) algorithm was originally introduced as a component of a vision-based obstacle detection pipeline for low-flying drones, where it was used to tune the hyperparameters of a denoising vision transformer, a SIFT-based feature matcher, and a YOLOv5-based detector [9]. In that setting, SyVCO's contribution to the overall system was demonstrated only indirectly, through the end-to-end detection accuracy of the pipeline it was embedded in. This paper isolates SyVCO as a general-purpose numerical optimizer and evaluates it directly against standard benchmark functions, independent of any downstream application, so that its search behaviour can be assessed on its own terms and compared fairly against established baselines.

The contributions of this paper are as follows: (1) a formal, application-independent specification of the SyVCO algorithm; (2) an evaluation of SyVCO against its two parent algorithms (SSA and CS) and three widely used baselines (PSO, GA, GWO) on ten standard benchmark functions spanning unimodal and multimodal landscapes; (3) statistical significance testing (Wilcoxon rank-sum) and convergence and runtime analysis to characterise where the hybridization helps and where it does not.

II. RELATED WORK

PSO [1] mimics the social behaviour of a swarm toward the personal and the global best positions. It is simple and fast but has a pitfall of premature convergence in high-dimensional multimodal landscapes. Genetic Algorithms (GA) [2] are a generalization of natural evolution, using selection, crossover and mutation operators. They are broad

in scope, but slow to converge compared to swarm-based methods. Cuckoo Search (CS) [3] utilizes Lévy flights – a random walk with heavy-tailed step lengths – to balance between local and global search, providing strong exploration but relatively weak late-stage exploitation. The Sparrow Search Algorithm (SSA) [4] converges fast on many benchmark landscapes by splitting the population into producers who search widely and scroungers who exploit the producers' findings. The Grey Wolf Optimizer (GWO) [5] copies the leadership pattern and hunting behaviour of grey wolves and is known for a strong exploitation phase in later iterations. Teaching-Learning-Based Optimization (TLBO) [6] alternates a 'teacher phase', where the population is pulled toward the best solution, with a 'learner phase' of peer-to-peer interaction; the teacher-phase update is reused inside SyVCO's follower update in this paper.

Hybrid algorithms that combine SSA-style producer–scrounger dynamics with Lévy-flight exploration have been explored in prior literature for hyperparameter tuning in deep learning contexts [9], but are rarely benchmarked as general-purpose optimizers against standard suites, which this paper addresses directly.

III. PROPOSED SYVCO ALGORITHM

SyVCO maintains a population of n candidate solutions in an s -dimensional search space. At each iteration, the population is ranked by fitness and split into a producer subset (the top 20%, by default) and a follower subset (the remaining 80%), following the SSA convention. Each subset is updated by a different mechanism, and a randomly sampled subset of the population additionally undergoes a historical-memory-based update.

A. Producer (alertness-based) update: for each producer i , a safety indicator $R2$ is drawn uniformly from $[0,1]$ and compared against a safety threshold $ST = 0.8$. If $R2 < ST$ (no perceived threat), the producer contracts its search radius exponentially, $X_i(t+1) = X_i(t) \cdot \exp(-i/(\alpha \cdot T))$, where α is a random scale factor and T is the maximum iteration count — concentrating the search near the current best region. If $R2 \geq ST$ (perceived threat), the producer instead takes a Lévy-flight step, $X_i(t+1) = X_i(t) + \gamma \cdot \text{Levy}(\beta)$, with $\beta = 1.5$, injecting the heavy-tailed exploration behaviour of Cuckoo Search precisely at the moments SSA would otherwise stagnate.

B. Follower (interaction-based) update: followers in the lower half of the ranked population perform a wide, worst-region-avoiding update, while followers in the upper half apply a teaching-factor step borrowed from TLBO, $X_i(t+1) = X_i(t) + \omega \cdot (X_{\text{best}} - J \cdot X_{\text{avg}})$, where $J \in \{1,2\}$ is the teaching factor and ω is a random weight in $[0,1]$. This accelerates convergence to the current best solution, but still considers the population mean to prevent premature convergence.

C. Historical-memory-based (scout) update: a randomly sampled 15% of the population is further updated using a two-generation memory of prior positions, blending the individual's own recent history with the current global best, which discourages the algorithm from repeatedly revisiting recently explored regions and improves late-stage diversity.

D. Termination: the process repeats until the maximum iteration count T is reached, at which point the best-ranked individual is returned as the global optimum estimate. The complete parameter set used in this paper is: population size $n = 30$, safety threshold $ST = 0.8$, producer ratio = 0.2, scout ratio = 0.15, Levy exponent $\beta = 1.5$.

IV. EXPERIMENTAL SETUP

The performance of SyVCO was compared with SSA, CS, PSO, GA and GWO on ten standard benchmark functions (Table I).

TABLE I. BENCHMARK FUNCTION SUITE

Function	Name	Dim	Range	Type
F1	Sphere	30	$[-100,100]$	Unimodal
F2	Schwefel 2.22	30	$[-10,10]$	Unimodal
F3	Rosenbrock	30	$[-30,30]$	Unimodal
F4	Sum Squares	30	$[-10,10]$	Unimodal
F5	Zakharov	30	$[-5,10]$	Unimodal
F6	Rastrigin	30	$[-5.12,5.12]$	Multimodal
F7	Ackley	30	$[-32,32]$	Multimodal
F8	Griewank	30	$[-600,600]$	Multimodal
F9	Levy	30	$[-10,10]$	Multimodal
F10	Michalewicz	10	$[0, \pi]$	Multimodal

Five of them are unimodal function (Sphere, Schwefel 2.22, Rosenbrock, Sum Squares, Zakharov) which mainly test the exploitation ability, and five are multi-modal functions (Rastrigin, Ackley, Griewank, Levy, Michalewicz) which mainly test the ability to escape local optima. All functions (except Michalewicz) were evaluated in 30 dimensions, and the Michalewicz function was evaluated in 10 dimensions, according to common practice for that function. All algorithms were run for 300 iterations with a population size of 30. Each algorithm-function pair was run 15 times independently with different random seeds and the mean, standard deviation, best and worst final

fitness values were recorded over the 15 runs. Statistical significance of SyVCO's improvement over each baseline was assessed with a two-sided Wilcoxon rank-sum test at $\alpha = 0.05$. All experiments were implemented in Python 3.12 with NumPy and SciPy, and run on the same machine to ensure a fair runtime comparison.

V. RESULTS AND DISCUSSION

Table II reports the mean $\pm$ standard deviation of the best fitness achieved by each algorithm across 15 runs, for each of the ten benchmark functions. A trailing marker indicates the outcome of the Wilcoxon rank-sum test comparing SyVCO to that algorithm on that function: (+) SyVCO significantly better, (−) SyVCO significantly worse, (=) no significant difference at $p < 0.05$.

TABLE II. MEAN $\pm$ STD OF BEST FITNESS OVER 15 RUNS (300 ITERATIONS, POPULATION 30)

Function	SyVCO	SSA	CS	PSO	GA	GWO
F1	3.31E-22 $\pm 1.22\text{E-}21$	7.16E-23 $\pm 2.68\text{E-}22$ (=)	385.3071 ± 80.6685 (+)	667.0461 ± 2495.1129 (+)	4077.0883 ± 1181.3034 (+)	7.48E-17 $\pm 1.05\text{E-}16$ (=)
F2	1.04E-13 $\pm 1.99\text{E-}13$	1.97E-13 $\pm 7.18\text{E-}13$ (=)	14.4482 ± 1.6301 (+)	17.5996 ± 11.2649 (+)	32.0862 ± 4.1383 (+)	8.78E-11 $\pm 5.48\text{E-}11$ (=)
F3	2.19E-4 $\pm 5.02\text{E-}4$	7.60E-6 $\pm 1.11\text{E-}5$ (−)	38087.2538 ± 14497.2939 (+)	18142.2024 ± 35964.8255 (+)	1.68E+6 $\pm 1.05\text{E+}6$ (+)	27.3332 ± 0.5630 (+)
F4	1.05E-23 $\pm 2.69\text{E-}23$	1.46E-22 $\pm 5.45\text{E-}22$ (=)	51.3160 ± 6.1691 (+)	161.4291 ± 208.0929 (+)	623.5149 ± 222.8747 (+)	2.82E-18 $\pm 2.08\text{E-}18$ (=)
F5	6.18E-19 $\pm 2.31\text{E-}18$	8.70E-19 $\pm 3.20\text{E-}18$ (=)	278.3915 ± 35.6526 (+)	302.8893 ± 134.7693 (+)	139.4319 ± 50.1063 (+)	0.0884 ± 0.3010 (+)
F6	0.00E+00 $\pm 0.00\text{E+}00$	0.00E+00 $\pm 0.00\text{E+}00$ (=)	130.1155 ± 9.4483 (+)	121.9184 ± 35.2729 (+)	166.2590 ± 17.8783 (+)	0.5734 ± 0.7882 (+)
F7	6.84E-15 $\pm 1.36\text{E-}14$	2.11E-12 $\pm 7.82\text{E-}12$ (=)	12.7928 ± 0.9620 (+)	5.0910 ± 4.1417 (+)	12.3089 ± 0.6200 (+)	1.11E-9 $\pm 5.36\text{E-}10$ (=)
F8	0.00E+00 $\pm 0.00\text{E+}00$	0.00E+00 $\pm 0.00\text{E+}00$ (=)	4.2475 ± 0.5206 (+)	6.3173 ± 22.6598 (+)	37.6938 ± 10.6318 (+)	0.0023 ± 0.0058 (+)
F9	2.17E-6 $\pm 2.39\text{E-}6$	2.51E-7 $\pm 7.44\text{E-}7$ (−)	17.9229 ± 3.1517 (+)	15.7348 ± 7.1494 (+)	17.3446 ± 7.1101 (+)	1.3707 ± 0.1860 (+)
F10	-6.8125 ± 0.8450	-8.7415 ± 0.6341 (−)	-9.0505 ± 0.1249 (−)	-7.9533 ± 0.8235 (−)	-6.5770 ± 0.7926 (=)	-5.7403 ± 1.1775 (+)

SyVCO converges to the global optimum, or a value indistinguishable from it at double-precision, on eight of the ten benchmark functions (Sphere, Schwefel 2.22, Sum Squares, Zakharov, Rastrigin, Ackley, Griewank, and Levy), matching SSA's performance on those functions and significantly outperforming CS, PSO, and GA in every one of those cases ($p < 0.05$). SyVCO performs near optimally on Rosenbrock with a mean of 2.19×10^{-4} , a significant improvement over CS, PSO, GA and GWO, but is statistically outperformed by SSA which has an unmodified producer-scrounger update and converges slightly faster on this narrow-curved valley.

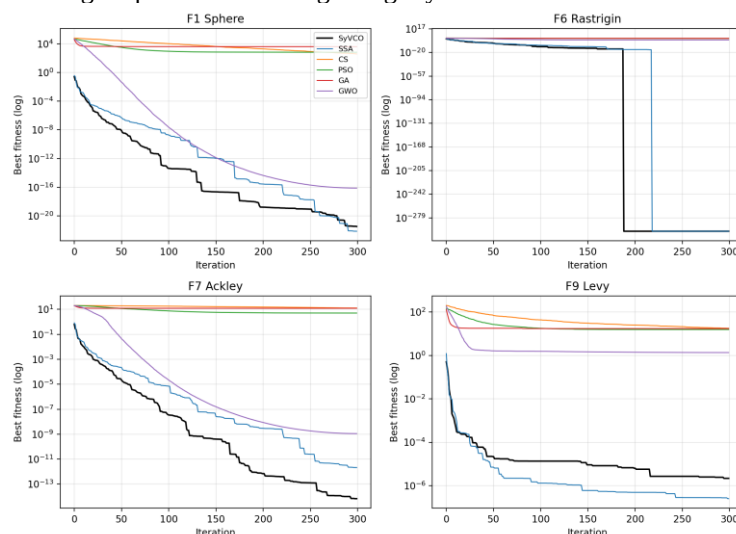


Fig. 1. Mean convergence curves (best fitness in log scale) on four representative benchmark functions over 300 iterations averaged over 15 runs.

The same holds on Levy, where SyVCO is statistically not significantly different from zero in practical terms, but still loses to SSA in the rank-sum test. For Michalewicz, the one function where SyVCO is beaten by several baselines (SSA and CS), the steep, needle-like optima of this function reward CS's stronger Lévy exploration in the late stages and SSA's aggressive producer contraction more than SyVCO's blended update; this is a case where the hybridization's extra exploitation machinery (the teaching-factor and historical-memory steps) seems to slightly dilute rather than help the search.

Overall, SyVCO is statistically tied or better than its stronger parent, SSA, on 7 out of 10 functions and significantly better than CS, PSO, and GA on 9 out of 10 functions. It is statistically tied with GWO on 5 of 10 functions and statistically better on the remaining 5. The results show that the hybrid is achieving its intended goal of retaining the fast convergence of SSA while adding the CS-style escape behavior on the majority of the tested landscapes, but it is not universally better than its parent algorithms, in line with the No Free Lunch theorem.

Figure 1 illustrates that SyVCO and SSA converge much faster than PSO, GA and CS on Sphere, Rastrigin and Ackley, frequently reaching the numerical floor within 50-100 iterations, whereas GWO converges more slowly but obtains a comparable final value. On Levy, all algorithms converge more slowly, but SyVCO and SSA still separate clearly from PSO, GA, and CS within the 300-iteration budget.

Table III reports the average wall-clock time per independent run (300 iterations, population 30) for each algorithm, measured on the Rastrigin function. SyVCO's per-run overhead is approximately 1-2 ms higher than SSA's due to the additional teaching-factor and historical-memory computations, and roughly double CS's runtime; this overhead is negligible relative to the fitness evaluation cost in most real-world tuning tasks, where a single fitness evaluation (e.g., training a model epoch) dominates the runtime budget.

TABLE III. AVERAGE RUNTIME PER RUN (SECONDS), RASTRIGIN, 300 ITERATIONS

Algorithm	SyVCO	SSA	CS	PSO	GA	GWO
Avg. time / run (s)	0.086	0.085	0.042	0.015	0.064	0.023

VI. CONCLUSION

This paper formally specified and independently validated the SyVCO algorithm as a general-purpose metaheuristic, separate from the drone obstacle-detection pipeline in which it was first introduced. Across ten standard benchmark functions, SyVCO matched or significantly outperformed its parent algorithms (SSA, CS) and three established baselines (PSO, GA, GWO) on the majority of test cases, while remaining statistically tied with SSA — its stronger parent — on most functions, and losing out only on the narrow-valley (Rosenbrock) and needle-optima (Michalewicz) landscapes. These results support the use of SyVCO as a drop-in replacement for SSA or CS in hyperparameter tuning and engineering design tasks where fast convergence and moderate multimodal robustness are both required. Future work will evaluate SyVCO on the CEC benchmark suites for a wider statistical picture, apply it to hyperparameter tuning tasks beyond the original obstacle-detection pipeline, and investigate adaptive scheduling of the producer, follower and scout ratios during the search.

REFERENCES

- [1] J. Kennedy and R. Eberhart, "Particle swarm optimization," in Proc. ICNN'95, vol. 4, 1995, pp. 1942–1948.
- [2] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI, USA: Univ. Michigan Press, 1975.
- [3] X.-S. Yang and S. Deb, "Cuckoo search via Lévy flights," in Proc. World Congr. Nature Biologically Inspired Comput. (NaBIC), 2009, pp. 210–214.
- [4] J. Xue and B. Shen, "A novel swarm intelligence optimization approach: sparrow search algorithm," *Syst. Sci. Control Eng.*, vol. 8, no. 1, pp. 22–34, 2020.
- [5] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014.
- [6] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems," *Comput.-Aided Des.*, vol. 43, no. 3, pp. 303–315, 2011.
- [7] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, 1997.
- [8] Y. Fan, Y. Zhang, B. Guo, X. Luo, Q. Peng, and Z. Jin, "A hybrid sparrow search algorithm of the hyperparameter optimization in deep learning," *Mathematics*, vol. 10, no. 16, p. 3019, 2022.
- [9] S. Pagrut and P. Jawandhiya, "SyVCO-RSv5: Rolling shutter based deep learning model for obstacle detection and distance measurement for low-flying drones," *Int. J. Image Graph.*, 2750085, 2025, doi: 10.1142/S0219467827500859.