# Simulation Experiments of Different Metaheuristics Algorithms using Benchmark Functions: A Performance Study

Imad El hajjami
*Faculty of Sciences*
*Moulay Ismail University*
Meknes, Morocco
i.elhajjami@edu.ac.umi.ma

Bachir Benhala
*Faculty of Sciences*
*Moulay Ismail University*
Meknes, Morocco
b.benhala@ieee.org

*Abstract*— **Metaheuristics have been commonly used in several engineering optimizations, they prerequisite reduced time to converge and produce a better-improved solution. The most applied are Evolutionary Algorithms. Many complex problems are no longer considered difficult due to swarm intelligent optimization algorithms that supply rapid and reliable methods for solutions. and this returns to its features such as robustness, flexibility, self-organization, parallel, and distributive. In this paper, a comparative study among five metaheuristics algorithms in terms of convergence, robustness, and computing time is accomplished, and three benchmark functions are applied to perform simulation experiments with Genetic Algorithm (GA), Firefly Algorithm (FA), Particle Swarm Algorithm (PSO), Invasive Weed Optimization (IWO), and Grey Wolf Optimizer (GWO). The experimental results show that GE provides more accuracy for complex optimization problems, while the GWO and PSO are better in terms of convergence speed.**

***Keywords— Metaheuristics, Particle Swarm Algorithm (PSO), Optimization Performance, Grey Wolf Optimizer (GWO), Convergence Velocity, Firefly Algorithm (FA), Genetic Algorithm (GA), Invasive Weed Optimization (IWO).***

## I. INTRODUCTION

In the face of optimization troubles of several complex problems, the methodologies practiced are habitually predicated on metaheuristics [1], which have assured their efficiency in resolving complex issues, they have been very vastly used in numerous practical engineering problems, such as telecommunication networks, electric power systems, robot control, and parameter estimation…

The most metaheuristics commonly exercised for optimization are Evolutionary Algorithms (EA) [2], such as Differential Evolution (DE) [1] [3], and the Genetic Algorithm (GE) [4] [5] [6]. SI algorithms, or 'Swarm Intelligence Techniques', are a new set of nature-inspired heuristic optimization algorithms inspired by the collective behavior of animal groups such as bee swarms, ant colonies, or flocks of birds [1], the most popular swarm intelligence techniques are Gravitational Search Algorithm (GSA) [7], Ant Colony Optimization (ACO) [8], Artificial Bee Colony (ABC) [6], Dragonfly Algorithm [9], Grey Wolf Optimizer [10], Bacterial Foraging [11], and Particle Swarm Optimization (PSO) [12]…

The frequent weakness that collides with these algorithms is that each metaheuristic has a problem in distinct phases such as a minimum optimization precision, a dawdling convergence velocity, difficulty, or ease in falling into the perfect solution [13]. In this work, a performance study of the most practical algorithms is achieved, this essay presents an application of five commonly used metaheuristics optimization techniques since they have been and through applied in different complex cases in the latest decade.

The paper is arranged in that way: In section 2 metaheuristics algorithms are introduced. After that, the computer experiments are achieved in section 3, and the results are assayed in detail. Last and not least, the conclusion illustrates the last part.

## II. METAHEURISTICS OVERVIEW

### A. Genetic Algorithm (GA)

A genetic algorithm (GA) is a metaheuristic that belongs to the evolutionary algorithms (EA) class which is inspired by the process of natural selection.

Genetic algorithms are usually used to optimize many complex problems.

GA builds on biologically inspired operators such as mutation, crossover, and selection [14] [15].

### B. Grey Wolf Optimizer (GWO)

The wolf as the upper predator in the food cycle has a powerful capacity to catch hunting. Wolves usually like social life and in the interior of the wolves exists a strict social hierarchy [16] [13].

The grey wolf optimizer was proposed in 2014 by Seyedali Mirjalili et al. It is a novel heuristic swarm intelligent optimization algorithm suggested.

978-1-6654-9558-5/22/$31.00 ©2022 IEEE

The GWO algorithm mimics the behavior of wolves in nature, the hunting mechanism of grey wolves, and the leadership hierarchy they are the basics of the GWO.

The mechanism of GWO uses four types of grey wolves including alpha, beta, delta, and omega are utilized for simulating the leadership hierarchy. besides that, three fundamental processes of hunting, searching for prey, encircling prey, and attacking prey, are carried out to implement optimization [10].

### C. Particle Swarm Optimization (PSO)

The PSO algorithm is proposed by Kennedy and Eberhart in 1995. It is a metaheuristic method that is suitable to optimize nonlinear continuous functions. The author derived the algorithm inspired by the connotation of swarm intelligence, overwhelmingly seen in the animal set, like shoals and flocks [17].

In the PSO algorithm, a swarm of birds flying everywhere should find a point that depends on various cases to land and survival by minimizing and maximizing respectively the danger of the presence of predators and the availability of food ...

In this situation, the movement of the birds is a choreography; the birds still fly simultaneously for a time till the best area available for food and security is assured then the flock lands at the same time. As a consequence, the determination of which location the whole swarm should land is a complicated issus[17].

### D. Firefly Algorithm (FA)

The Firefly algorithm (FA) is proposed by Xin-She Yang in 2008. It is a metaheuristic algorithm for global optimization, that is inspired by the flashing behavior of firefly insects. In this algorithm, fireflies practice the flashing behavior to be the center of attention of other fireflies, commonly for dispatching signals to the opposite gender. Nonetheless, in the mathematical model, utilized inside Firefly Algorithm, the fireflies are simply of both sexes, and whatever firefly sex it can attract another firefly [18].

The connection between fireflies is related to their radiance and for any couple of fireflies, the shining one will attract the other, so the less bright one is moving across the brighter one. and that is carried out for any binary combination in the population, for every iteration of the algorithm [19].

### E. Invasive Weed Optimization (IWO)

Alireza Mehrabian and Caro Lucas, in 2006, proposed a nature-inspired metaheuristic, inspired by the spreading strategy of weeds, namely Invasive Weed Optimization (IWO), and Based on the r/K selection theory [20], the artificial weeds (solutions) employ the r-selection strategy at the start, and progressively they transform to the K-selection strategy, as the algorithm keeps running [21] [22].

## III. Experiment and Analysis Results

To compare the performance of algorithms used, the three benchmark functions presented in Table I, F1, F2, and F3, and namely respectively; Ackley N. 4, Sphere, and Schaffer N. 4, all of which are optimized separately.

The surface plot of a 2-variable for Ackley N. 4, Schaffer N. 4, and Sphere functions are shown in Figs 1–3.

The algorithms parameters are given in Tables II–VI.

TABLE I. Benchmark Functions Description.

| | F1 | F2 | F3 |
|---|---|---|---|
| **Function** | **Ackley N. 4** | **Sphere** | **Schaffer N. 4** |
| **Input Domain** | [-100,100]<br>[-1,1] | [-10,10] | [--100,100] |
| **Dimensional space** | 60 | 30 | 100 |
| **Global Minima** | f (x∗)= at x∗= | f (x∗)= at x∗=(0,…,0) | f (x∗)= at x∗= |

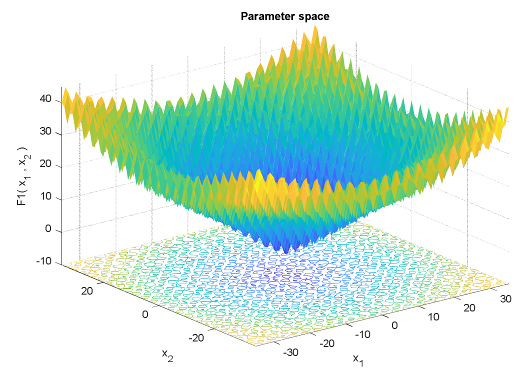


Fig. 1. Surface plot with a contour plot of a 2-variable for Ackley N. 4.

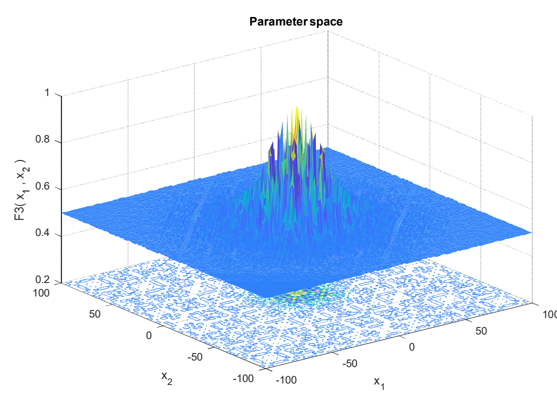


Fig. 2. Surface plot with a contour plot of a 2-variable for Schaffer N. 4.

For validating the performance of five algorithms, computer experiments are executed. The convergence curves are shown in Figs 4–8.

For F1, the optimal values obtained are between -166 to -160 for D = 60, while IWO gives the best score in the input domain [-1,1], and when D = 100 in the input domain [-100,100], the optimal values obtained are between -229.1 to 103.8, in this case, the GA gives the best solution, and the best score of GA increased by 43.18% of improvement, while IWO couldn't search for the best score in this input domain.

For F2, the optimal values obtained are between e-5 to e-43, the optimization performance of GWO for a unimodal function is proved, and GWO grants the perfect minimum value of 0.

For F3, the optimal values obtained are between 0.2926 to 0.48, in this case, the five algorithms are offered good results.

From Fig. 8, it is not possible to get good results by IWO for a larger input domain.

Subsequently, for more accuracy regarding optimization precision of GA, IWO, GWO, PSO, and FA, each algorithm for every test function is turned on fifty times, then the best, the std, the average value, and computing times for F1, F2, and F3 are listed in Tables VII–X, the maximum iteration number is iter $_{max}$= 500.

Computer configurations utilized are symbolized as follows:

Processor (CPU): Core i5-7200U CPU @ 2.50 GHz 2.71 GHz.

Memory (RAM): 4 GB.

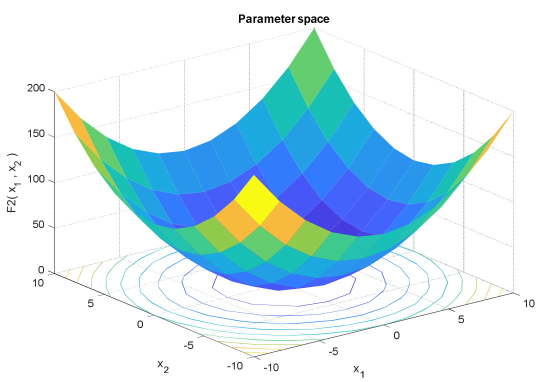


Fig. 3. Surface plot with a contour plot of a 2-variable for Sphere.

TABLE II. THE GA PARAMETERS.

| Parameter | Value |
|---|---|
| Population size | 100 |
| Iterations | 500 |
| Mutation probability | 0.01 |
| Crossover operator | 0.5 |

TABLE III. THE GWO PARAMETERS.

| Parameter | Value |
|---|---|
| Population size | 100 |
| Iterations | 500 |
| a | [0,2] |
| $r_1$ and $r_2$ | [0,1] |
| Population size | 100 |
| Iterations | 500 |

TABLE IV. THE PSO PARAMETERS.

| Parameter | Value |
|---|---|
| Maximum particle velocity $v_{max}$ | 6 |
| Maximum inertia weight $w_{max}$ | 0.9 |
| Minimum inertia weight $w_{min}$ | 0.2 |
| Cognitive component $c_1$ | 2 |
| Social component $c_2$ | 2 |
| Population size | 100 |
| Iterations | 500 |

TABLE V. THE FA PARAMETERS.

| Parameter | Value |
|---|---|
| Light Absorption Coefficient γ | 1 |
| Attraction Coefficient Base Value β | 2 |
| Randomization α | 0.2 |
| Population size | 100 |
| Iterations | 500 |

TABLE VI. THE IWO PARAMETERS

| Parameter | Value |
|---|---|
| Population size | 100 |
| Iterations | 500 |
| Initial Population Size Iterations | 10 |
| Minimum Number of Seeds | 0 |
| Maximum Number of Seeds | 5 |
| Variance Reduction Exponent | 2 |
| Initial Value of Standard Deviation | 0.5 |

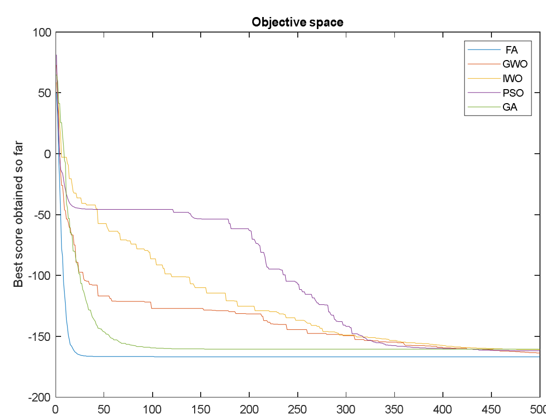


Fig. 4. Comparison of convergence curves of F1 for ((Input Domain =[--1,1]), 60-dimension).

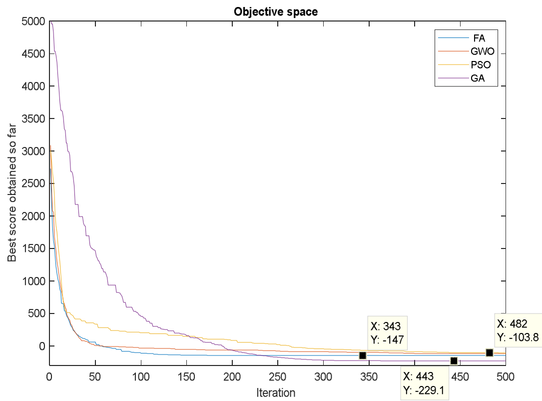


Fig. 5. Comparison of convergence curves of F1 for ((Input Domain = [-100,100]), 60-dimension).

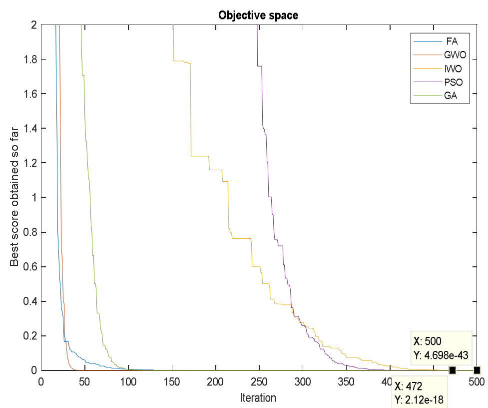


Fig. 6. Comparison of convergence curves of F2 for D=30.

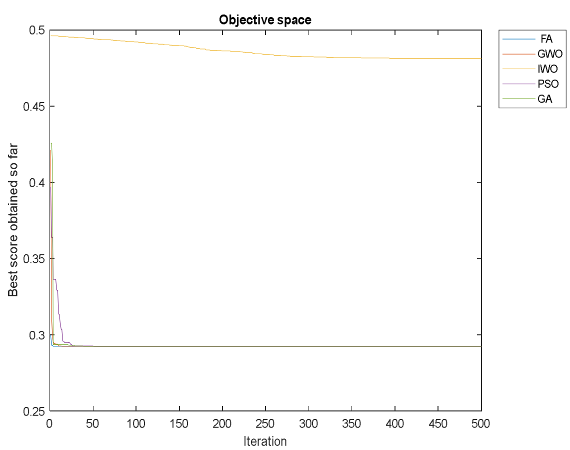


Fig. 7. Comparison of convergence curves of F3 for D=100.

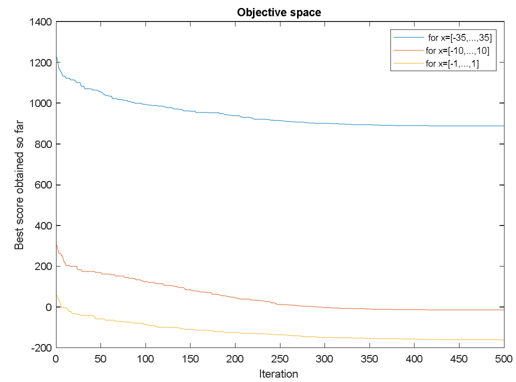


Fig. 8. Comparison of convergence curves of F1 for different input domains using IWO.

The robustness and searching precision of F1, F2, and F3 are presented in Figs 9–11.

It is conspicuous from the simulation curves, that GA has the best searching precision than PSO, GWO, FA, and IWO, for the most part when we have great value with a wider field for solutions searching, and this was confirmed by function F1. GWO and PSO, all are faster, while FA presents a strong optimization precision but is need a big time for searching.

TABLE VII. THE ROBUSTNESS ANALYSIS RESULTS ( D=60, INPUT DOMAIN = [-1,1]).

| Algorithms | F1 | | |
|---|---|---|---|
| | **AVG** | **STD** | **MIN** |
| **FA** | -137.06 | 11.7 | -150.74 |
| **GA** | -147.31 | 7.98 | -161.03 |
| **IWO** | -159.28 | 5.30 | -166.32 |
| **GWO** | -114.17 | 14.07 | -136.82 |
| **PSO** | -107.65 | 22.55 | -146.26 |

TABLE VIII. THE ROBUSTNESS ANALYSIS RESULTS, D=30.

| Algorithms | F2 | | |
|---|---|---|---|
| | **AVG** | **STD** | **MIN** |
| **FA** | 6.56e-10 | 4.50e-10 | 8.29e-11 |
| **GA** | 1.73e-19 | 1.29e-19 | 1.17e-20 |
| **IWO** | 8.03e-5 | 1.45e-5 | 4.25e-05 |
| **GWO** | 5.01e-43 | 6.53e-43 | 5.066e-44 |
| **PSO** | 2.62e-7 | 3.72e-07 | 5.10e-10 |

TABLE IX. THE ROBUSTNESS ANALYSIS RESULTS, D=100.

| Algorithms | F3 | | |
|---|---|---|---|
| | AVG | STD | MIN |
| FA | 0.2926 | 3.68e-15 | 0.2926 |
| GA | 0.2926 | 2.64e-6 | 0.2926 |
| IWO | 0.2926 | 0.0581 | 0.2926 |
| GWO | 0.2926 | 4.51e-8 | 0.2926 |
| PSO | 0.2926 | 2.04e-16 | 0.2926 |

TABLE X. THE AVERAGE COMPUTING TIMES FOR F1, F2, AND F3.

| Algorithms | Times (s) | | |
|---|---|---|---|
| | F1 | F2 | F3 |
| FA | 2960 | 2390 | 2654.24 |
| GA | 114.34 | 100 | 113 |
| IWO | 73.35 | 63.79 | 93.78 |
| GWO | 47.68 | 19.66 | 53.06 |
| PSO | 28.90 | 8.53 | 21.23 |

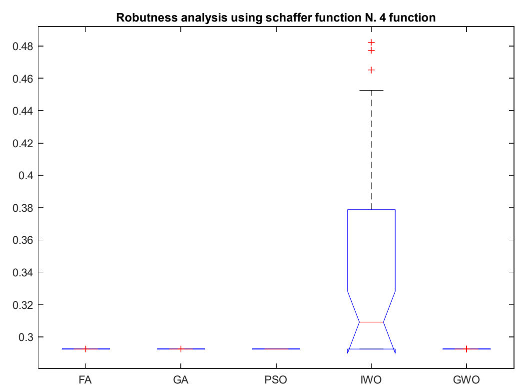


Fig. 9. Boxplot distribution solutions for F3.

Boxplot distribution shows that (GA) presents a better convergence and strong robustness, although (PSO) has a faster convergence.

It is apparent from simulation, the optimization precision and convergence velocity of GWO, FA, and PSO, are all in a good range, but GA is the best.

## IV. CONCLUSION

For achieving the best choice of the algorithm used in terms of optimization problems, a comparative study between different metaheuristics is proposed in this paper. Computer experiments are accomplished by taking on three function optimization issues with different descriptions. Simulations show that FA has strong robustness. For multimodal function, GA has the best searching precision, in particular when you have a higher dimension and a larger input domain.

Moreover, GA has an optimal performance for complex problem solving, while IWO has been effective only in the small input domains.

The convergence speed of PSO and GWO are best, as though, the five metaheuristics are giving perfect results for unimodal function.

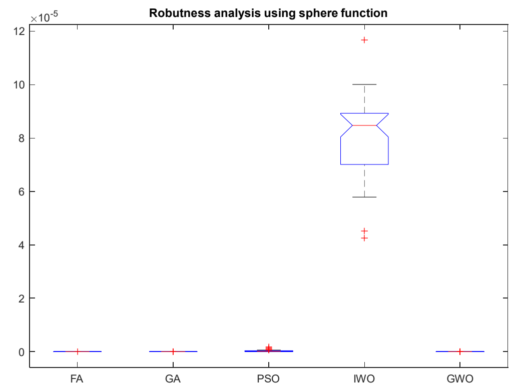


Fig. 10. Boxplot distribution solutions for F2.

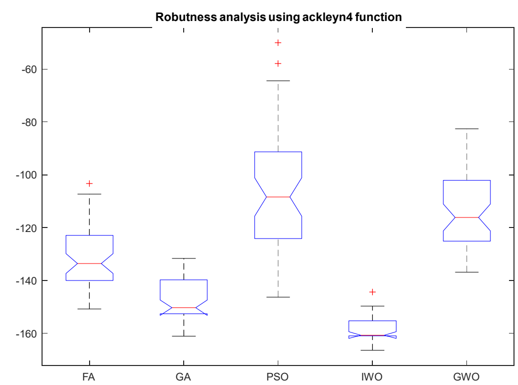


Fig. 11. Boxplot distribution solutions for F1.

## References

[1] I.E. Hajjami, B. Benhala, and H. Bouyghf, “Optimal Design of RF Integrated Inductors via Differential Evolution Algorithm,” *in IEEE 2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*, 2020, pp. 1–6.

[2] M. Crepinsek, S. H. Liu, and M. Mernik, “Exploration and exploitation in evolutionary algorithms: A survey,” *in ACM Computing Surveys,* vol. 45, no. 3, 2013.

[3] R. Storn and K. Price, “Differential evolution: A simple and efficient heuristic for global optimization over continuous spaces,” *in J. Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[4] M. Mitchell, “ An introduction to genetic algorithms, ” *in The MIT Press*, 1998.

[5] B. Benhala and O. Bouattane, “GA and ACO techniques for the analog circuits design optimization”, *in Journal of Theoretical and Applied Information Technology (JATIT)*, vol. 64, no. 2, pp. 413–419, 2014.

[6] I.E. Hajjami, B. Benhala, and H. Bouyghf, “Comparative Study between the Genetic Algorithm and the Artificial Bee Colony Technique: RF Circuits Application,” *in 2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS),* 2020, pp. 1–5.

[7] E. Rashedi, H. Nezamabadi, and S. Saryazdi, “GSA: A gravitational search algorithm,” *in Information Sciences*, vol. 179, no. 13, pp. 2232–2248, 2009.

[8] B. Benhala, “Sizing of inverted current conveyors by an enhanced ant colony optimization technique,” *in 2016 Conference on Design of Circuits and Integrated Systems (DCIS)*, Granada, 2016, pp. 1–5.

[9] S. Mirjalili, “Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems,” *in Neural Computing and Applications*, vol. 27, no. 4, pp. 1053–1073, 2016.

[10] S. Mirjalili, S. M. Mirjalili, and A. Lewis, “Grey wolf optimizer,” *in Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.

[11] K. M. Passino, “Biomimicry of bacterial foraging for distributed optimization and control,” *in IEEE control systems Magazine*, vol. 22, no. 3, pp. 71–67, 2002.

[12] J. Kennedy and R. Eberhart, “Particle swarm optimization, in Neural Networks, ” *in Proceedings IEEE International Conference*, 1995, pp. 1942–1948.

[13] J. S. W. Eason and S. H. Lie, “An Improved Grey Wolf Optimizer Based on Differential Evolution and Elimination Mechanism,” *in Sci Rep*, vol. 9, pp. 71–81, 2019.

[14] M. Melanie, “An Introduction to Genetic Algorithms,” *Cambridge, MA: MIT Press,* 1996.

[15] I.E. Hajjami, B. Benhala, and H. Bouyghf, “Shape Optimization of Planar Inductors for RF Circuits using a Metaheuristic Technique based on Evolutionary Approach,” in *Advances in Science, Technology and Engineering Systems Journal*, vol. 5, no. 5, pp. 426–433, 2020.

[16] P. Majumder and T.I. Eldho, “Artificial Neural Network and Grey Wolf Optimizer Based Surrogate Simulation-Optimization Model for Groundwater Remediation,” in *Water Resources Management*, vol. 34, pp. 763–783, 2021.

[17] B. S. G. de Almeida and V.C. Leite, “Particle Swarm Optimization: A Powerful Technique for Solving Engineering Problems*,” in Swarm Intelligence - Recent Advances, New Perspectives, and Applications*, IntechOpen Book Series, 2019.

[18] X. S. Yang, “Chapter 9 - Firefly Algorithms,” in *Nature-Inspired Optimization Algorithms (Second edition)*, pp. 123–139, 2021.

[19] M. K. Heris, *Firefly Algorithm (FA) in MATLAB*, Yarpiz, 2015. [Online]. Available: https://yarpiz.com/259/ypea112-firefly-algorithm

[20] R. Oizumi, T. Kuniya, Y. Enatsu, “ Reconsideration of r/K Selection Theory Using Stochastic Control Theory and Nonlinear Structured Population Models,” *in PLoS ONE,* vol. 11, no. 6, 2016.

[21] M. Misaghi and M. Yaghoobi, “Improved invasive weed optimization algorithm (IWO) based on chaos theory for the optimal design of PID controller,” *in Journal of Computational Design and Engineering*, vol. 6, pp. 284–295, 2019.

[22] M. K. Heris, *Invasive Weed Optimization (IWO) in MATLAB*, Yarpiz, 2015. [Online]. Available: https://yarpiz.com/251/ypea119-invasive-weed-optimization