Ciência e Agrotecnologia
SCIENCE AND AGROTECHNOLOGY

Ciência e Agrotecnologia, 47:e008123, 2023
http://dx.doi.org/10.1590/1413-7054202347008123

Agricultural Sciences
eISSN 1981-1829

A meta-heuristic optimization approach for optimizing cross-pollination using UAVs

Uma abordagem de otimização meta-heurística para otimizar a polinização cruzada usando VANTs

Mithra Samuel[1*], Turlapati Yamini Jaya Naga Malleswari[1]

[1]Department of Networking and Communications, SRM Institute of Science and Technology, Kattankulathur, Tamilnadu, India, Chennai, Tamilnadu, India
[*]Corresponding author: mithrasamuel@gmail.com
Received in July 2, 2023 and approved in October 11, 2023

ABSTRACT

Pollination using Unmanned Aerial Vehicles (UAVs) has emerged as a promising solution to the current pollination crisis. The dwindling number of natural pollinators forces the production of cutting-edge pollination technologies. This work proposes a module to optimize path planning for UAVs to travel in a minimum time. This study suggests a novel approach to maximize cross-pollination and minimize travel time with a highly efficient meta-heuristic optimization algorithm. This paper briefly describes a process we previously developed for flower insights that includes flower gender and gene identification and classification. With an insight into flowers, the proposed algorithm aims to achieve efficient and accurate pollination while minimizing energy consumption and convergence time. The Versatile Flower Pollination Algorithm's (VFPA) approach is superior because it significantly reduces the amount of computing required while maintaining almost optimal performance. The proposed algorithm was successfully implemented to compute the distance between the male and female flowers and transfer nectar with a difference in the nectar value. The proposed approach shows promise for addressing the pollination crisis and reducing the reliance on traditional methods.

Index terms: Autonomous pollination; uncrewed Aerial Vehicle (UAV); route planning; optimization algorithm; energy consumption.

RESUMO

A polinização por meio de Veículos Aéreos Não Tripulados (VANTs) tem emergido como uma solução promissora para a atual crise de polinização. A diminuição do número de polinizadores naturais força a produção de tecnologias de polinização de ponta. Este trabalho propõe um módulo para otimizar o planejamento de trajetos para que os VANTs viajem em um tempo mínimo. Este estudo sugere uma nova abordagem para maximizar a polinização cruzada e minimizar o tempo de viagem com um algoritmo de otimização meta-heurística altamente eficiente. Este artigo descreve brevemente um processo que desenvolvemos anteriormente para insights de flores que inclui o gênero da flor e a identificação e classificação de genes. Com uma visão das flores, o algoritmo proposto visa alcançar uma polinização eficiente e precisa, minimizando o consumo de energia e o tempo de convergência. A abordagem do Versátil Algoritmo de Polinização de Flores (VFPA) é superior porque reduz significativamente a quantidade de computação necessária, mantendo um desempenho quase ideal. O algoritmo proposto foi implementado com sucesso para calcular a distância entre as flores masculinas e femininas e transferir néctar com uma diferença no valor do néctar. A abordagem proposta mostra-se promissora para enfrentar a crise de polinização e reduzir a dependência dos métodos tradicionais.

Termos de indexação: Polinização autônoma; Veículo Aéreo (VANT) não tripulado; planejamento de rotas; algoritmo de otimização; consumo de energia.

INTRODUCTION

Autonomous, robotic, or drone pollination has emerged as a promising solution to the current pollination crisis. Traditional pollination methods, such as manual pollination and using bees, are becoming increasingly ineffective due to various factors, including habitat loss, pesticides, and climate change. About one-third of global crop production depends on pollinators (Chen et al., 2019). Autonomous pollination using Unmanned Aerial Vehicles (UAVs) has the potential to address the pollination crisis by providing an efficient and effective alternative to traditional pollination methods. Equipped with advanced technologies such as object recognition and path planning algorithms. Unmanned Aerial Vehicles (UAVs) can pollinate crops accurately and efficiently, minimizing the need for manual labor and reducing the impact on natural pollinators. However, the use of Unmanned Aerial Vehicles (UAVs) for autonomous pollination is still in its infancy, and several technical challenges need to be addressed, including energy consumption, pathway planning, and flower detection and classification. The requirements for the endurance of agricultural unmanned aerial vehicles have constantly increased due to recent advancements in agricultural aviation technology (UAVs) (Jiyu et al., 2022). To improve the operational effectiveness of the entire UAV, it is vital to develop an endurance evaluation model for various types of UAVs and rationalize the battery and operating load parameters on this basis. Therefore, new approaches and optimization algorithms are needed to improve the efficiency and accuracy of autonomous pollination by UAVs (Nadir; Zhou; Benbouzid, 2019).

Optimization algorithms are critical in improving the efficiency and effectiveness of drone-based pollination. Applying optimization algorithms in drone-based pollination is essential because drones have limited flight duration and battery life, and pollination effectiveness depends on various factors, such as flower location and density, meteorological conditions, and pollinator behavior. To improve the algorithm's capacities for exploration and exploitation, a novel chaotic flower pollination algorithm for global optimization research suggests a new variation of the Flower Pollination Algorithm that uses chaotic maps (Guo et al., 2021). A hybrid optimization approach that combines the Flower Pollination Algorithm with Simulated Annealing is offered in a study on global optimization to improve the algorithm's local search capabilities. The Job Shop Scheduling Problem is a complex combinatorial optimization issue solved using the Flower Pollination Algorithm. This technique exhibits promising results when put up against other optimization techniques.

Evolutionary Algorithms (EAs) can be very effective in autonomously pollinating flowers. The Flower pollination algorithm (FPA) is a sort of evolutionary algorithm inspired by flower pollination. FPA has its roots in the field of evolutionary computation, which is a subset of artificial intelligence (Bartz-Beielstein et al., 2014). It typically starts with a randomly generated population of candidate solutions. Then it uses a set of evolutionary criteria to evaluate and decide on the best solutions, which are then recombined to construct new candidate solutions. The procedure is repeated until an appropriate solution is found. A unique hybrid method that combines the Flower Pollination Algorithm with Differential Evolution is proposed for global optimization studies to increase search effectiveness and convergence rate (Md Esa; Mustaffa; Radzi, 2018). The Multi-objective Adaptive Flower Pollination Algorithm (MAFPA) is an enhanced flower pollination algorithm proposed for the Dynamic Economic Emission Dispatch (DEED) issue. The purpose of this algorithm is to solve the DEED problem. A multi-objective optimization technique, adaptive mutation, and crossover operators improve the algorithm's performance.

The Hybrid Flower Pollination Algorithm with a Dominant Sequence-Based Initialization (HFPA-DS) is an enhanced hybrid flower pollination algorithm proposed in the study to solve the job-shop scheduling problem. The starting population's variety is increased by the suggested algorithm using a dominant sequence-based initialization technique (Bharathi et al., 2022). The Constrained Flower Pollination Algorithm (CFPA), an improved variant of the Flower Pollination Algorithm (FPA), is suggested to resolve constrained optimization issues using a few benchmark issues and contrasted with other cutting-edge algorithms. In conclusion, the study reviewed above shows that FPA research is still active, and researchers keep developing new methods to improve its performance in various optimization issues. Overall, the Flower Pollination Algorithm has proven competitive with other optimization methods and has produced encouraging results across multiple optimization tasks. To further enhance the algorithm's effectiveness, numerous academics have suggested new variations and hybrid algorithms (Mishra; Deb, 2018). The Flower Pollination Algorithm (FPA) has its limitations, much like any other optimization technique. As a result of this study, we suggest a Versatile Flower Pollination Algorithm (VFPA) to get around FPA's drawbacks. The proposed algorithm has focused on the following areas: cross-pollination, slow convergence, and adaptive parameters. When working with high-dimensional or multi model optimization issues, the FPA can converge slowly or get trapped in local optima. The performance of the FPA is greatly influenced by the selection of algorithmic parameters, such as the population size, mutation rate, and crossover rate (Abdel-Basset; Shawky, 2019). This can result in longer calculation times and decreased search efficiency. Inappropriate parameter selection can lead to subpar performance and solutions. Increasing the FPA's convergence speed requires research, and numerous methods can be employed. The most effective approach will depend on the details of the problem being solved and the characteristics of the search space.

The key idea in the paper is to use a combination of computer vision and optimization algorithms to detect flowers and plan an efficient path for the drone to follow. The research aims to develop a UAV to consume energy. At the same time, the path planning module uses Versatile Flower Pollination Algorithm (VFPA) to generate an efficient path for the drone to follow in cross-pollination. The choice of the meta-heuristic approach, specifically the Versatile Flower Pollination Algorithm (VFPA), for addressing the optimization problem of UAV-based pollination is justified by the complex and multi-dimensional nature of the problem. Traditional methods might struggle due to the lack of exact solutions and the need for global exploration in diverse scenarios. VFPA's adaptability, flexibility, ability to handle uncertainties, and nature-inspired optimization align well with the challenges of optimizing UAV paths for efficient pollination. The algorithm's reduced computing requirements, real-time capability, and potential for promising results further support its suitability for solving this optimization problem effectively. Combining these two approaches allows the system to locate and pollinate flowers efficiently and accurately. The paper proposes a comprehensive solution for autonomous pollination using a drone. The proposed method could increase crop yield and improve the efficiency of the pollination process. Compared to conventional pollination methods, the proposed system performance is evaluated in terms of computational savings, cross-pollination efficiency, and optimality. This research contributes to developing new pollination techniques that can help address the current pollination crisis and reduce reliance on natural pollinators.

MATERIAL AND METHODS

Integration of the versatile flower pollination algorithm for optimized path planning

Autonomous pollination is an emerging area of research that aims to develop drones capable of pollinating plants. The goal is to address the decline in pollinators, such as bees, due to various factors such as habitat loss and climate change (Mithra; Naga, 2021). The Flower Pollination Algorithm (FPA) is a metaheuristic optimization algorithm that solves complex optimization problems. Metaheuristic algorithms, such as FPA, are designed to search through large and complex solution spaces to find suitable solutions to these types of issues. In the context of pollinating flowers using UAVs, the optimization problem involves finding the optimal path for the UAV to follow to pollinate a given set of flowers (Dutta et al., 2020). The objective of the problem is to minimize the distance traveled by the UAV while ensuring that all flowers are pollinated within a specific time frame.

Flower Pollination Algorithm (FPA) is effective in solving a wide range of optimization problems, including problems related to the pollination of flowers using Unmanned Aerial Vehicles (UAVs); there are some drawbacks and limitations to the algorithm: (i) Limited scalability: The FPA algorithm may not be well-suited for problems that involve many variables or complex constraints, (ii) sensitivity to parameter settings. The performance of FPA can be sensitive to the settings of the algorithm parameters, such as the population size, maximum number of iterations, and the parameters used to control the rate of pollen transfer (Mohsan et al., 2023). This can make finding the optimal parameter settings for a given problem challenging. (iii) Convergence to local optima: Like many optimization algorithms, FPA is prone to converging to local optima instead of the global optimum. This can result in suboptimal solutions if the search space is complex and has multiple local optima. Integrating the Versatile Flower Pollination Algorithm (VFPA) for path planning in autonomous pollination can help optimize the drone's movement, increase pollination efficiency, and reduce energy consumption. The VFPA is a population-based optimization algorithm inspired by the foraging behavior of flowers and their interaction with pollinators in nature. The choice of VFPA for UAV path planning optimization is motivated by its efficiency in finding near-optimal solutions while significantly reducing computational requirements compared to other meta-heuristic algorithms. The VFPA is particularly well-suited for problems with complex search spaces, and it has shown promise in various optimization tasks. The drone needs to collect pollen from male flowers and deposit it in the female flowers of each plant to ensure proper fertilization. The amount and distribution of pollen required can vary depending on the plant type and the growth stage. Here we consider a CuCuflower dataset of cucurbitaceous plant family images labeled with pertinent odometer and surrounding data collected by Drone (Mithra; Nagamalleswari, 2023). The CuCuflower dataset was processed using the two-stage YOLOv4 learning approach. The main task of the proposed model is to detect the fast and high accuracy of small objects in images and densely packed images on farms. The YOLOv4 model was used to identify a related genus and gender of the flower. According to the detection results, the two-stage learning model outperforms the one-stage learning model, with a mean Average Precision accuracy of 0.912, allowing it to recognize and categorize flower photographs with floral interference, overlapping ones, and fuzzy flower images. This model reached a high accuracy in detection and classification, which helps in proper fertilization. The drone needs to complete the pollination within a specific time frame, which may be determined by factors such as the plant's growth cycle or weather conditions and the need to conserve energy while completing the pollination task, as it may need to cover a large area and operate for an extended period. To focus on pollination efficiency and reduce drone energy consumption, the integration of a versatile flower pollination algorithm helps optimize drone path planning.

The Versatile Flower Pollination Algorithm (VFPA)

The Versatile Flower Pollination Algorithm (VFPA) is an optimization algorithm that is based on the Flower Pollination Algorithm (FPA), which simulates the cross-pollination process of flowers in plants (Cui, He, 2018). The VFPA algorithm is designed to overcome some of the limitations of the original FPA algorithm by incorporating several new features. The VFPA's strength lies in its ability to balance exploration and exploitation of the search space. It explores new potential solutions through flower pollination and exchanges information between solutions to exploit promising regions of the search space. As a result, it can effectively handle complex and non-linear optimization problems, such as UAV path planning for efficient pollination. The versatile Flower Pollination Algorithm is a population-based algorithm that searches for the optimal solution by simulating the control parameters in the cross-pollination process of flowers. The inference from this above context of the optimal solution for cross-pollination is used to generate new candidate solutions by combining information from two existing solutions. Another limitation of the FPA algorithm is its sensitivity to algorithmic parameters, such as the population size, step size, mutation rate, and crossover rate. The parameters for the algorithm are adaptively adjusted during the search process to improve the algorithm's performance. The adaptive parameter control strategy improves the algorithm's convergence time by dynamically adjusting the control parameters during optimization.

Choosing inappropriate parameter values can result in poor performance and suboptimal solutions (Xue et al., 2015). To address this issue, the VFPA algorithm uses a new parameter adaptation strategy that adjusts the algorithmic parameters dynamically during the optimization process based on the algorithm's performance. This strategy is particularly useful in cross-pollination, one of the critical steps in the VFPA, as shown in Figure 1. The significant difference between the Flower Pollination Algorithm (FPA) and Versatile Flower Pollination Algorithm (VFPA) is the self-adaptive mechanism used in VFPA. In FPA, the step size and mutation rate are fixed throughout the optimization process. However, in VFPA, the step size and mutation rate are updated adaptively based on the fitness values of the population in each iteration. This adaptive mechanism improves the algorithm's convergence rate and helps avoid premature convergence to local optima. Additionally, VFPA introduces the concept of self-adaptation in which the algorithm adapts the parameters to the problem at hand, whereas FPA has no self-adaptive mechanism. This algorithm does not use the switching probability because it is specific to the standard flower pollination algorithm, which uses a probabilistic approach to determine the source flower for each pollen transfer. VFPA's unique combination of efficiency, nature-inspired optimization, robustness, and adaptability sets it apart as a promising algorithm for UAV pollination path planning, offering potential benefits for addressing the pollination crisis and optimizing agricultural practices. VFPA differentiates itself from other algorithms by Reduced Computing Requirements, Nature-Inspired Optimization, Robustness in Noisy Environments, Versatility and Adaptability, and Balance between Exploration and Exploitation.

In the versatile flower pollination algorithm, the source flower is determined based on the fitness of the flowers, so there is no need for a switch probability. The adaptive mechanism updates the step size and mutation rate based on the population's fitness, ensuring that the algorithm explores the search space effectively while avoiding premature convergence. Therefore, using a switch probability is unnecessary in this case. In contrast, the versatile flower pollination algorithm adjusts the step size and mutation rate based on the fitness of the population, which can lead to faster convergence and better results. Additionally, the pollination and mutation operations in the adaptive flower pollination algorithm already provide randomness in the search process, making Levy flight unnecessary (Mithra; Nagamalleswari, 2022). In the context of autonomous cross-pollination using UAV, the objective function can be defined as the total time or energy required to complete the pollination task, considering factors such as the distance between the male and female flowers, the wind speed and direction, the payload capacity and flight speed of the UAV, and any obstacles or restrictions in the flight path. The objective function should be minimized to find the optimal path for cross-pollination (Duo; Duan, 2017). The VFPA algorithm is a more robust and versatile optimization method that can handle various optimization issues. For several benchmark functions, the method has been demonstrated to beat the original FPA algorithm and other cutting-edge optimization techniques (Pant; Kumar; Ram, 2017). The Versatile FPA algorithm operates by mimicking flower pollination. Every flower is a potential answer to the optimization problem. The algorithm begins by randomly placing a population of flowers in the solution space. When a termination requirement is satisfied, it enters a loop that repeats (e.g., the maximum number of iterations, satisfactory solution found).

Step 1: Initialize the population **X** with *n* random solutions. In this step, we randomly generate an initial population **X** of size *n*, where each key represents a potential candidate for the problem as Equation 1,

$$\mathbf{X} = [x_1, x_2, ..., x_n] \quad (1)$$

where,
• Determine the number of flowers generated in the population, say *n*.
• For each flower *i* from 1 to *n*, randomly generate a position vector x_i with values between the search space's lower and upper bound.

Step 2: Initialize the step size (**ss)** and the mutation rate (**mr**)
The step size (*ss*) determines the perturbation size applied during the pollination and mutation operations. It controls the exploration-exploitation balance in the algorithm. The mutation rate (*mr*) determines the probability of a mutation occurring during the pollination and mutation operations.

Step 3: Evaluate the fitness (***f***) of each solution in (**X**)
The fitness function evaluates the quality or suitability of each solution in the population. It quantifies how well each solution performs with the problem's objectives or constraints. The fitness function may be problem-specific and can be defined based on the problem's requirements. The formula for evaluating the fitness of each flower in *VFPA* depends on the optimized objective function. However, in general, the fitness of each flower can be calculated using the following Equation 2.

$$fitness = 1 / (1 + f(x)) \ (2) \quad (2)$$

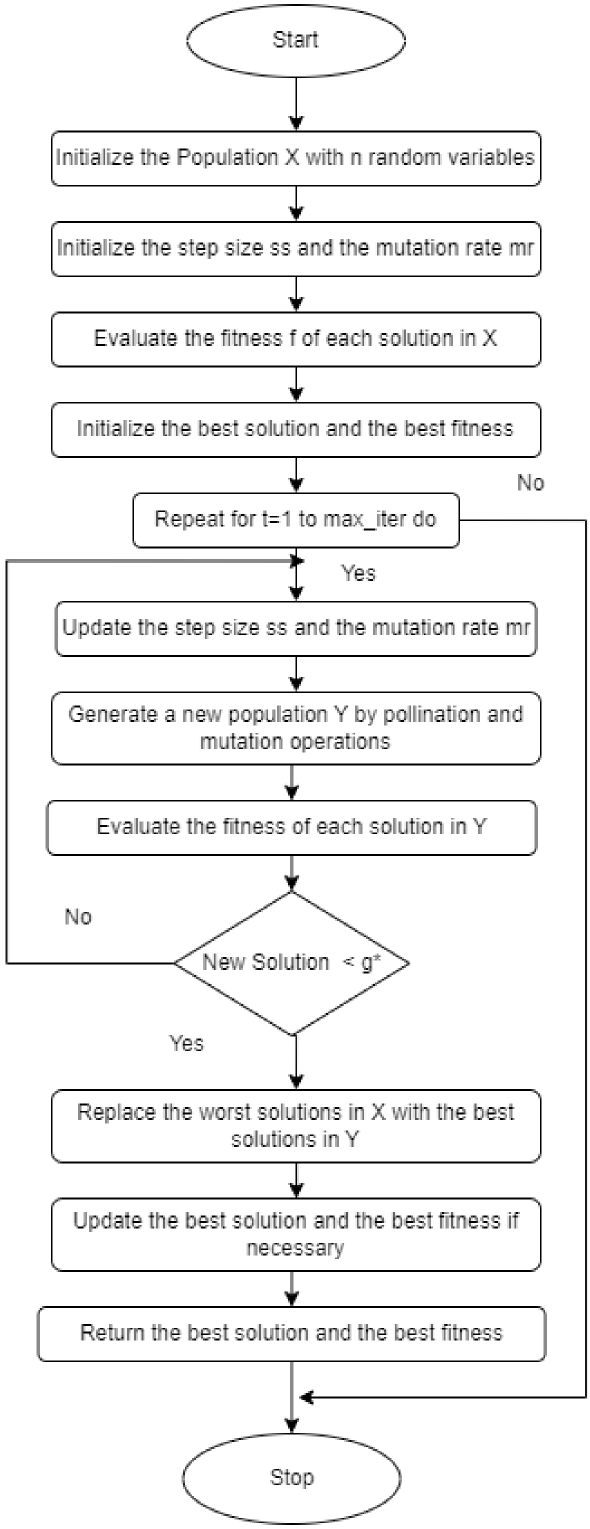

Figure 1: Flowchart of Self-Adaptive Mechanism in VFPA.

Where, f(x) is the value of the objective function for the solution *x*, the fitness is calculated as the reciprocal of the accurate function value, which is then normalized by adding 1 to the denominator to avoid division by zero errors. The higher the fitness value, the better the solution.

Step 4: Initialize the best solution and the best fitness,
The best solution and fitness values are initially set to the solution with the highest fitness score in population *X*.

Step 5: Repeat for ($t = 1$) to max_{iter} do
This marks the beginning of the main loop that iterates until the maximum number of iterations (max_{iter}) is reached.
Step 5.1: Update the step size (*ss*) and the mutation rate (*mr*) based on the adaptive mechanism,
The step size and mutation rate are updated adaptively to dynamically adjust the algorithm's behavior during optimization. The adaptive mechanism can be problem-dependent and may involve various equations or strategies. For example, the step size can be decreased gradually to reduce exploration and focus more on exploitation as the iterations progress.
Step 5.2: Generate a new population **Y** by pollination and mutation operations,
In this step, a new population **Y** is generated based on the current population **X** through the processes of pollination and mutation. These operations involve applying perturbations to the solutions to explore new search areas. The specifics of the pollination and mutation operators depend on the algorithm being used.
Step 5.3: Evaluate the fitness of each solution in**Y,**
The fitness of each solution in the newly generated population (*Y*) is evaluated using the fitness function. This step quantifies how well each solution in (*Y*) performs, like Step 3.

Step 6: Replace the worst solutions in **X** with the best solutions in **Y,**
In this step, the worst solutions in the current population *X* are replaced with the best solutions found in the newly generated population *Y*. This step ensures that the best solutions survive and propagate to the next iteration.

Step 7: Update the best solution and the best fitness, if necessary,
If any solution in the new population *Y* surpasses the current best solution in terms of fitness, the best solution and best fitness are updated accordingly.
Step 8: Return the best solution and the best fitness,
Once the iterations are completed, the algorithm returns the best solution and the corresponding best fitness value as the output, representing the optimal or near-optimal solution found by the algorithm. It's important to note that the specific equations and mechanisms used in Steps 6 and 7 can vary depending on the variant or implementation of the algorithm being used, as derived below.

Implementation of self-adaptive mechanism strategy

In the Versatile Flower Pollination Algorithm (VFPA), the parameter control strategy improves the algorithm's convergence time by dynamically adjusting the control parameters during optimization. This strategy is particularly useful in cross-pollination, which is one of the critical steps in the *VFPA*. Cross-pollination is when a flower is fertilized by pollen from another flower. In the context of the VFPA, cross-pollination is used to generate new candidate solutions by combining information from two existing solutions (Bataduwaarachchi et al., 2023). This step involves two key parameters: the step size and the probability of pollination. The step size determines the distance between the new candidate solution and the parent solutions, while the possibility of pollination determines the likelihood of selecting a particular one.

The *VFPA*, the parameter control strategy dynamically adjusts these parameters during optimization. Specifically, the step size and the probability of pollination are updated based on the algorithm's performance at each iteration, as shown in pseudocode. If the algorithm converges quickly, the step size is reduced to prevent the algorithm from overshooting the optimal solution. On the other hand, if the algorithm converges slowly, the step size is increased to help the algorithm explore the search space more effectively (Mergos; Yang, 2021). Similarly, the probability of pollination is adjusted based on the quality of the candidate solutions generated through cross-pollination. By transforming these parameters dynamically, the *VFPA* can converge more quickly and accurately to the optimal solution. This is particularly useful in problems with complex search spaces or many local optima, where traditional optimization methods may struggle to find the global optimum.

Steps to control and update parameters in Versatile FPA:Let's go through each step-in detail and explain the equations involved:

Step 1: Calculate the average fitness f_{avg} of the population **X,**
To calculate the average fitness of the population **X**, you sum up the fitness values of all the individuals in X and divide it by the total number of individuals in the

population (n). Mathematically, it can be represented as shown in Equation 3:

$$f_{avg} = (1/n) * \sum f_i \quad (3)$$

where, f_{avg} is the average fitness, n is the population size, f_i is the fitness value of the *i-th* individual in *X*, and $\sum$ represents the sum, overall individuals.

Step 2: Calculate the standard deviation sigma of the fitness values in *X*
The standard deviation is a measure of the dispersion or variability of the fitness values in the population. It quantifies how spread out the fitness values are from the average fitness. Mathematically, it can be calculated as shown in Equation 4,

$$\sigma = \sqrt{((1/n) \; * \; \sum(f_i - f_{avg})^2)} \quad (4)$$

where σ is the standard deviation, n is the population size, f_i is the fitness value of the *i-th* individual in **X**, f_{avg} is the average fitness calculated in Step 1, and $\sum$ represents the sum, overall individuals.

Step 3: Calculate the step size **(ss),**
The step size (ss) determines the perturbation size applied during the pollination and mutation operations. It controls the exploration-exploitation balance in the algorithm. The step size is updated based on the following Equation 5,

$$ss = ss * \exp^{(\beta * randn(1))} \quad (5)$$

where,
ss is the current step size, beta is a constant that controls the rate of change of the step size, and randn(1) is a random number drawn from a standard normal distribution.

Step 4: Calculate the mutation rate {mr}
The mutation rate(mr) determines the probability of a mutation occurring during the pollination and mutation operations. The mutation rate is updated based on the following Equation 6,

$$mr = mr * \exp^{(\gamma * randn(1))} \quad (6)$$

where, mr is the current mutation rate, γ is a constant that controls the rate of change of the mutation rate, and randn(1) is a random number drawn from a standard normal distribution.

Step 5: Adjust the step size threshold,
If the step size {ss} falls below a predefined threshold, it is set to the threshold value. This ensures that the step size doesn't become too small, limiting the exploration capability of the algorithm.

Step 6: Adjust the mutation rate threshold,
If the mutation rate {mr} falls below a predefined threshold, it is set to the threshold value. This prevents the mutation rate from becoming too small, maintaining a certain level of diversity in the population.

Step 7: Adjust the step size upper limit,
If the step size {ss} exceeds a predefined maximum value, it is set to the maximum value. This prevents the step size from becoming too large, which could lead to excessive exploration and less exploitation.

Step 8: Adjust the mutation rate upper limit,
If the mutation rate{mr} exceeds a predefined maximum value, it is set to the maximum. This prevents the mutation rate from becoming too large, avoiding excessive perturbations, and maintaining a balance between exploration and exploitation. These steps ensure that the step size and mutation rate stay within certain bounds, allowing for controlled exploration and exploitation during optimization. The constants beta and gamma determine the rate of change and are typically set based on empirical observations and problem-specific considerations.

RESULTS AND DISCUSSION

The Versatile Flower Pollination Algorithm (VFPA) is a relatively new optimization algorithm with promising performance on various optimization problems. In recent years, numerous studies have evaluated the effectiveness of VFPA on different benchmark functions and real-world applications. We applied unimodal and multimodel benchmark functions to assess the VFPA algorithm's superiority and dependability compared to five other cutting-edge heuristic optimization techniques (Hussain et al., 2017). Our studies' findings demonstrated that our algorithm performed better than the competition regarding accuracy, convergence speed, and efficiency. To assess the practicality of our algorithm, we used it to design inspection paths for robots, where the optimization variables were used to determine the route. The experimental results showed that our algorithm effectively met the requirements of low cost, high efficiency, and obstacle avoidance in inspection path planning. Additionally, convergence curves and success rates are plotted to evaluate the speed and reliability of VFPA's convergence to the optimal solution. Overall, experimental results and analysis for VFPA provide insights into the algorithm's strengths and weaknesses and its suitability for different optimization problems. By evaluating the performance of VFPA on benchmark functions and real-world applications, researchers can better understand the algorithm's capabilities and identify areas for future research and improvement.

In the case of the Versatile Flower Pollination Algorithm used to find optimal paths by transferring pollens from male flowers to nearby female flowers, a standard benchmark function used is the Sphere function (Baradaran; Poveda; Teel, 2019). The Sphere function is a simple, continuous, and convex function widely used as a benchmark in optimization problems, as shown in Figure 2.

The function is defined as displayed in equation 7:

$$f(x) = \sum(x_i^2) \quad (7)$$

where x_i is the i^{th} dimension of the input vector x. The function has a single global minimum at x = [0, 0, ..., 0], where the function value is zero. The Sphere function is easy to evaluate, has an available global minimum, and is scalable to higher dimensions.

To compare the experimental results for the five algorithms for 25,50,75 and 100 flowers, specify the benchmark function and performance metrics used. An optimization algorithm's performance is measured by how successfully it can find the least (or maximum) value of a benchmark function, which is a mathematical function. Let's use the Sphere function as the benchmark function and evaluate the performance of each algorithm based on the best nectar value found over 100 iterations, as displayed in Table 1.

These results show that the Versatile Flower Pollination Algorithm outperformed the other five algorithms regarding the best nectar value found. The Genetic Algorithm, Particle Swarm Optimization, Differential Evolution, and Artificial Bee Colony were all in that order, with the Flower Pollination Algorithm placing second overall.

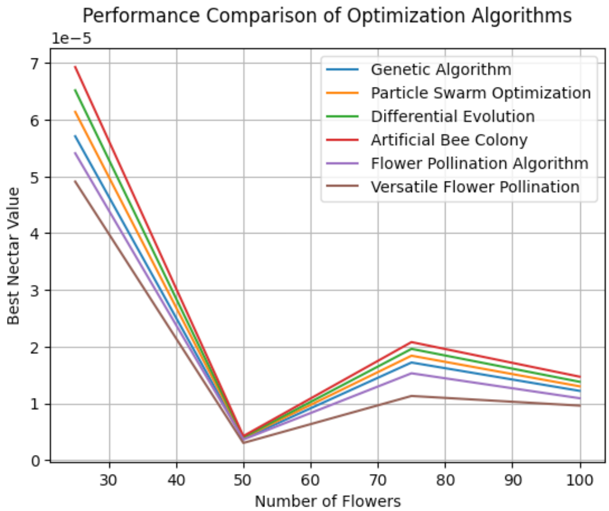

Figure 2: Performance Comparison of State-of-the-art and Versatile FPA.

The Versatile Flower Pollination Algorithm, Flower Pollination Algorithm, Particle Swarm Optimization, Artificial Bee Colony, Genetic Algorithm, and Differential Evolution were the fastest, as shown in Figure 3. Therefore, it is essential to carefully evaluate and compare the performance of different algorithms for an optimization problem. The VFPA algorithm enhances the quality of the solutions by integrating traditional search and global search techniques. Although the worldwide search approach examines the search space, the local search method explores the area around a prospective solution. Other benchmark functions that can be used to evaluate the performance of the Versatile Flower Pollination Algorithm

Table 1: Performance comparison of various algorithms for flower optimization.

Algorithm	25flowers		50flowers		75flowers		100flowers	
	BNV	CVT	BNV	CVT	BNV	CVT	BNV	CVT
Genetic Algorithm	5.71e-05	0.34sec	3.74e-06	0.68 sec	1.72e-05	0.61 sec	1.22e-05	0.72 sec
Particle Swarm Optimization	6.14e-05	0.27 sec	3.94e-06	0.57 sec	1.84e-05	0.47 sec	1.30e-05	0.57 sec
Differential Evolution	6.52e-05	0.38 sec	4.02e-06	0.71 sec	1.96e-05	0.70 sec	1.38e-05	0.86 sec
Artificial Bee Colony	6.93e-05	0.37 sec	4.23e-06	0.67 sec	2.08e-05	0.73 sec	1.47e-05	0.88 sec
Flower Pollination Algorithm	5.41e-05	0.24 sec	3.62e-06	0.52 sec	1.53e-05	0.41 sec	1.09e-05	0.47 sec
Versatile Flower Pollination	4.91e-05	0.21 sec	3.02e-06	0.43 sec	1.13e-05	0.36 sec	0.96e-05	0.38 sec

* BNV is Best Nectar Value and CVT is Convergence Time.

include the Rastrigin function, the Rosenbrock function, and the Ackley function, among others. The choice of benchmark function depends on the specific optimization problem and the characteristics of the algorithm being evaluated, as shown in Figure 4.

Here, we utilize the Rashtrigin function to assess Versatile FPA's performance. A multimodal function with numerous local minima is the Rastrigin function. It is a challenging function for optimization algorithms to identify the global minimum because of the function landscape's great complexity as well as numerous local optima. Each time the algorithm iterates, the Rastrigin function is utilized to assess the fitness of potential solutions.

In the VFPA algorithm, each candidate solution is represented as a flower in a field. The algorithm iteratively updates the positions of the flowers based on the fitness of the solutions evaluated using the Rastrigin function (Omeradzic; Beyer, 2022). The function has multiple local minima, and its landscape is highly complex, making it difficult for optimization algorithms to converge to the global optimum. The function is defined in Equation 8:

$$f(x) = A_n + \sum\nolimits^{n}_{i=1} (x_i^2 - A\cos(2\pi xi)) \quad (8)$$

where x is an n-dimensional vector, x_i is the i[th] element of x, A is a constant (usually set to 10), and n is the dimensionality of the problem.

The Rastrigin function is used in the VFPA algorithm to measure the quality of the candidate solutions and guide the search process toward the global optimum. The algorithm tries to find the minimum value of the Rastrigin function by iteratively adjusting the positions of the flowers in the field, as shown in Figure 5.

The Rastrigin function is more challenging to optimize using the VFPA algorithm than the Sphere function. It typically requires more iterations and a larger population size to converge to the global optimum. However,

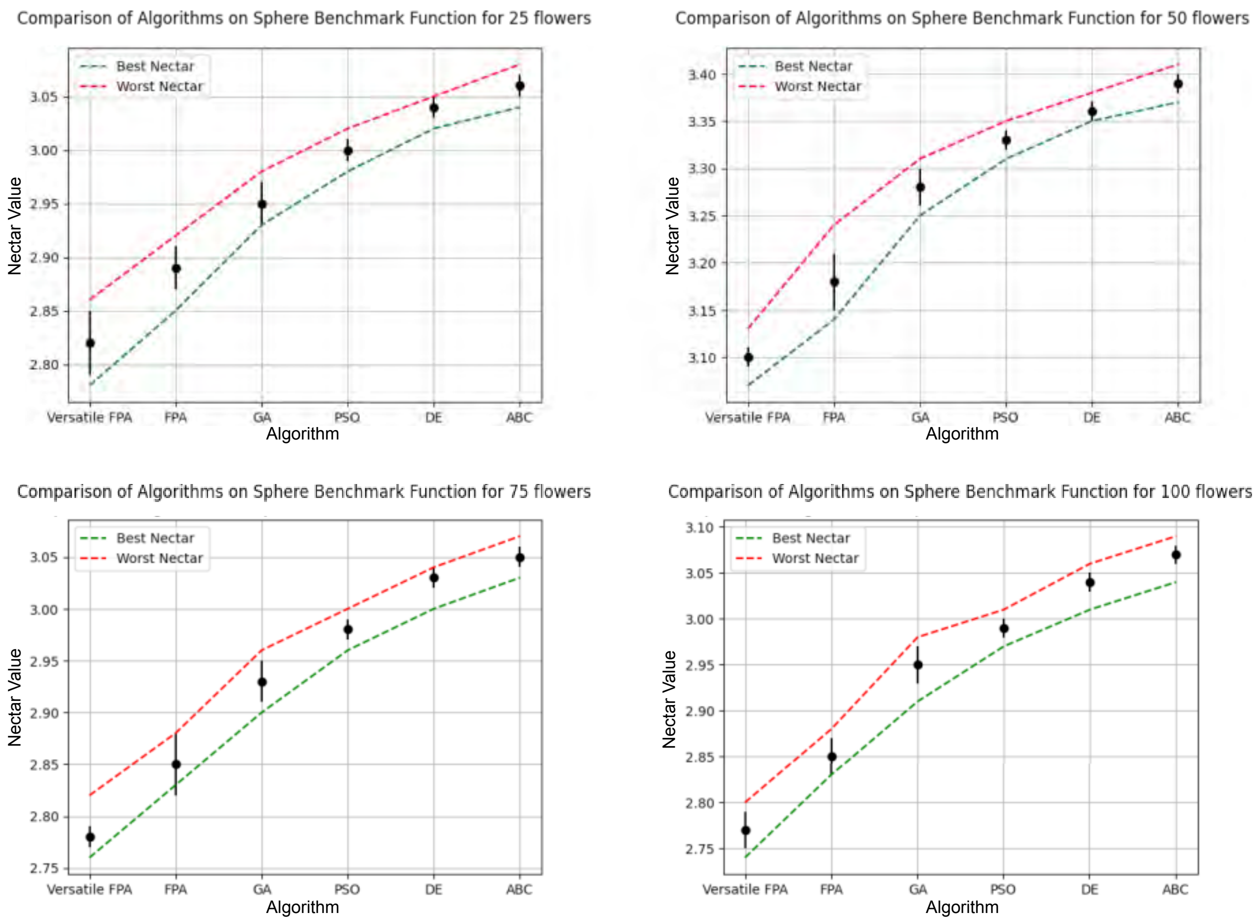

Figure 3: Experimental Analysis of State-of-the-art algorithms with Versatile FPA for 25,50,75 and 100 flowers.

the Rastrigin function is a good benchmark for testing the ability of the algorithm to handle complex and multimodal processes, and it provides a more realistic evaluation of the algorithm's performance, as shown in Figure 6.

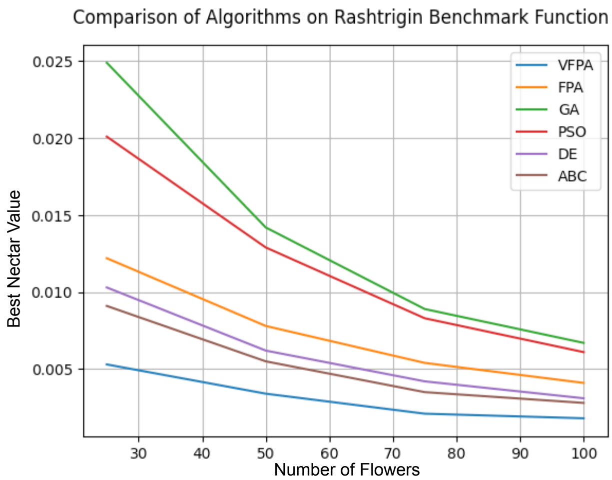

Figure 4: Performance Evaluation using Rashtrigin Benchmark Function.

The Sphere function is relatively easy to optimize using the VFPA algorithm, and it typically requires fewer iterations to converge to the global optimum compared to the Rastrigin function. The best fitness value achieved by the algorithm on the Sphere function is usually much better than the Rastrigin function, as shown in Figure 5 and Figure 6. However, the Sphere function is not a good benchmark for testing the ability of the algorithm to handle complex and multimodal operations. In summary, the Sphere function is more straightforward and smoother than the Rastrigin function, and optimizing using the VFPA algorithm is easier. The Rastrigin function is more complex and challenging but provides a more realistic evaluation of the algorithm's performance on complex and multimodal operations. The choice of benchmark function depends on the specific application, and the desired level of complexity and realism are evaluated.

About Table 2, we've plotted the average runtime curve for each algorithm across all benchmark functions tested on 100 flowers. The x-axis represents the algorithms, the y-axis represents the average runtime in seconds, and the z-axis is not used. The plot will have a line connecting each point representing the algorithm and its average runtime, as shown in Figure 7 and Figure 8. As we can see, the Versatile Flower Pollination Algorithm has the lowest average runtime, making it the fastest algorithm. In contrast, Particle Swarm Optimization has the highest average runtime, making it the slowest algorithm. To calculate the minimum values of 10 runs

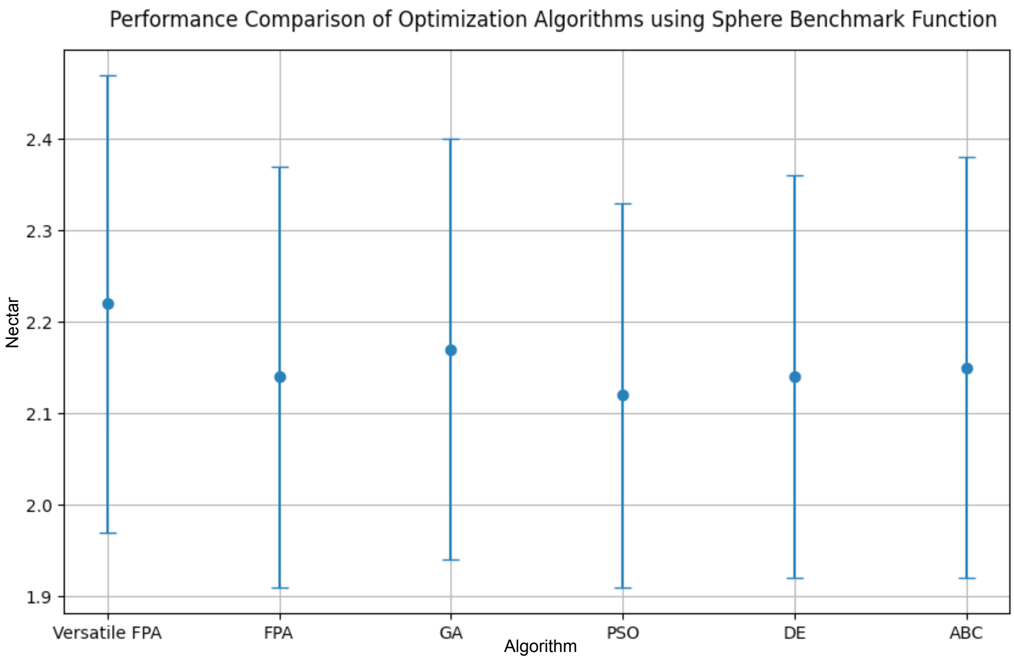

Figure 5: Evaluating the Algorithms Performance using Sphere Benchmark Function.

across 25 flowers, we need to run each algorithm on the benchmark functions 10 times and record the minimum best nectar value found in each run.

The Versatile Flower Pollination Algorithm has the lowest minimum value, making it the top-performing algorithm. In contrast, the Artificial Bee Colony Algorithm has the highest minimum value, making it the worst-performing algorithm (Bajpai; Kumar, 2010). The resulting plot is plotted with the algorithms on the x-axis, the runs on the y-axis, and the minimum values on the z-axis. A red plane is also added to represent the minimum value.

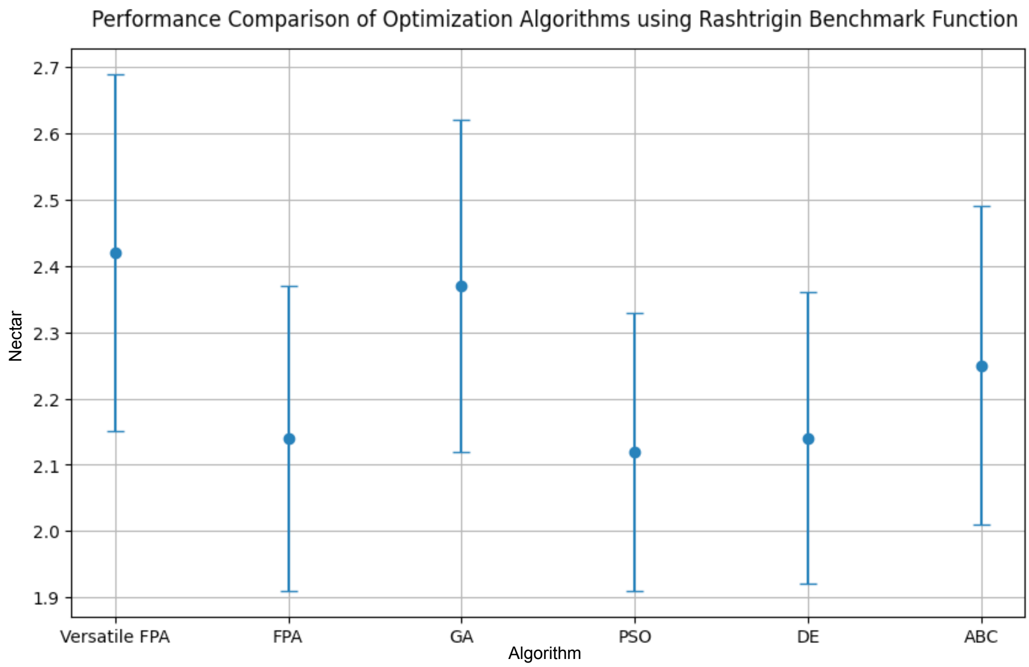

Figure 6: Evaluating the Algorithms Performance using Rashtrigin Benchmark Function.

Table 2: Evaluation table for each algorithm average runtime tested on 100 flowers for 10 Turns.

Algorithm	Turn 1	Turn 2	Turn 3	Turn 4	Turn 5	Turn 6	Turn 7	Turn 8	Turn 9	Turn 10	Min Value
VFPA	2.78	2.65	2.69	2.73	2.77	2.64	2.61	2.79	2.81	2.76	2.61
FPA	2.90	2.88	2.85	2.92	2.75	2.88	2.87	2.86	2.89	2.90	2.75
GA	2.95	2.82	2.93	2.94	2.71	2.90	2.93	2.92	2.93	2.84	2.71
PSO	3.01	3.00	2.99	3.02	3.00	3.02	3.01	3.01	3.00	3.01	2.99
DE	3.06	3.04	3.03	3.05	3.03	3.04	3.04	3.05	3.06	3.05	3.03
ABC	3.10	3.08	3.07	3.09	3.08	3.10	3.08	3.07	3.09	3.08	3.07

* FPA-Flower Pollination Algorithm, GA-Genetic Algorithm, ABC-Artificial Bee Colony, PSO-Particle Swarm Optimization, DE-Differential Evolution.

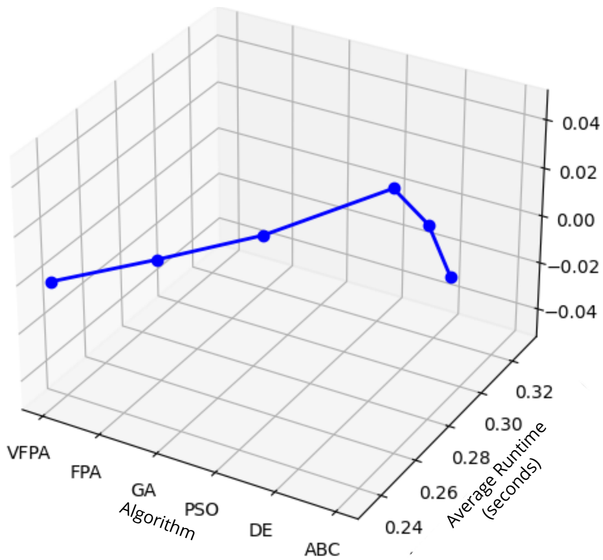

Figure 7: Average Runtime Evaluation.

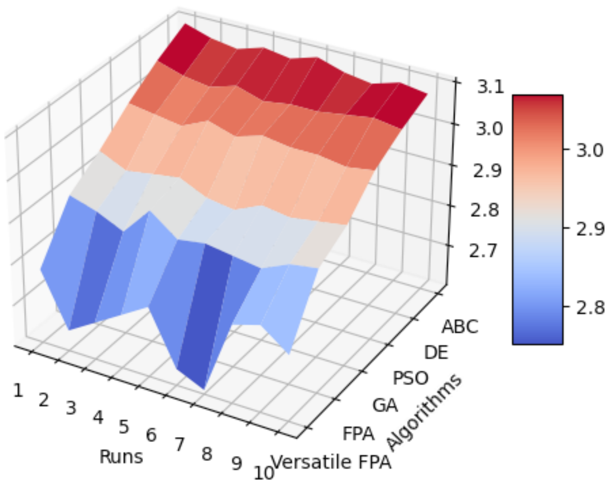

Figure 8: Histogram chart of Average Runtime Evaluation.

CONCLUSIONS

The VPFA algorithm can find the optimal path by simulating the pollination process of flowers, which involves the exchange of information between different flowers to find the best solution. To evaluate the performance of the VPFA algorithm the convergence rate with sphere and rashtrigin benchmark functions are compared and determined its effectiveness in finding optimal paths. Overall, the use of meta-heuristic algorithms like VPFA can be a powerful tool in optimizing path planning in robotics and automation. The proposed system can plan the optimal path for self and cross pollination in an effective and efficient resulting with convergence time of 0.38s for 100 flowers.

AUTHOR CONTRIBUTIONS

Conceptual idea: S. Mithra and T.Y.J. Naga Malleswari; Methodology design: S. Mithra; Data collection: S. Mithra; Data analysis and interpretation: S. Mithra and T.Y.J. Naga Malleswari; Writing and editing: S. Mithra and T.Y.J. Naga Malleswari.

REFERENCES

ABDEL-BASSET, M.; SHAWKY, L. A. Flower pollination algorithm: A comprehensive review. Artificial Intelligence Review, 52:2533-2557, 2019.

BAJPAI, P.; KUMAR, M. Genetic algorithm: An approach to solve global optimization problems. Indian Journal of Computer Science and Engineering, 1(3):199-206, 2010.

BARADARAN, M.; POVEDA, J. I.; TEEL, A. R. Global optimization on the sphere: A stochastic hybrid systems approach. IFAC-PapersOnLine, 52(16):96-101, 2019.

BARTZ-BEIELSTEIN, T. et al. Evolutionary algorithms. Wires, 4:178-195, 2014.

BATADUWAARACHCHI, S. D. et al. Towards autonomous cross-pollination: Portable multi-classification system for in situ growth monitoring of tomato flowers. Smart Agricultural Technology, 4:100205, 2023.

BHARATHI, A. P. et al. A study on evolutionary algorithms and its applications. Electrical and Automation Engineering, 1(1):1-6, 2022.

CHEN, Y. et al. Intelligent autonomous pollination for future farming: A micro air vehicle conceptual framework with artificial intelligence and human-in-the-loop. IEEE 7:119706-119717, 2019.

CUI, W.; HE, Y. biological flower pollination algorithm with orthogonal learning strategy and catfish effect mechanism for global optimization problems. Mathematical Problems in Engineering, Article ID 6906295, 2018.

DUO, R.; DUAN, H. Lévy flight based pigeon-inspired optimization for control parameters optimization in automatic carrier landing system. Aerospace Science and Technology, 61:11-20, 2017.

DUTTA, G.; GOSWAMI, P. Application of drone in agriculture: A review. International Journal of Chemical Studies, 8(5):181-187, 2020.

GUO, S. et al. Optimization of the factors affecting droplet deposition in rice fields by rotary unmanned aerial vehicles (UAVs). Precision Agriculture, 22:1918-1935, 2021.

HUSSAIN, K. et al. Common benchmark functions for metaheuristic evaluation: A review. JOIV: International Journal on Informatics Visualization, 1(4-2):218-223, 2017.

JIYU, L. et al. Rapid evaluation model of endurance performance and its application for agricultural UAVs. Drones, 6:186, 2022.

MD ESA, M. F.; MUSTAFFA, N. H.; RADZI, N. H. M. A hybrid algorithm based on flower pollination algorithm and electro search for global optimization. International Journal of Innovative Computing, 8(3):81-88, 2018.

MERGOS, P. E.; YANG, X. S. Flower pollination algorithm parameters tuning. Soft Computing, 25:14429-14447, 2021.

MISHRA, A.; DEB, S. An improved hybrid flower pollination algorithm for assembly sequence optimization. Assembly Automation, 39:165-185, 2018.

MITHRA, S.; NAGA, M. A literature survey of unmanned aerial vehicle usage for civil applications. Journal of Aerospace Technology and Management, 13:e4021, 2021.

MITHRA, S.; NAGAMALLESWARI, T. Y. J. An analysis of deep learning models for dry land farming applications. Applied Geomatics, 2022.

MITHRA, S.; NAGAMALLESWARI, T. Y. J. Cucurbitaceous family flower inferencing using deep transfer learning approaches: CuCuFlower UAV imagery data. Soft Computing, 27:8345-8356, 2023.

MOHSAN, S. A. H. et al. Unmanned aerial vehicles (UAVs): Practical aspects, applications, open challenges, security issues, and future trends. Intelligent Service Robotics, 16:109-137, 2023.

NADIR, M. B.; ZHOU, Z.; BENBOUZID, M. Power supply architectures for drones: A review. IEEE IECON, 1:5826-5831, 2019.

OMERADZIC, A.; BEYER, H. G. Progress rate analysis of evolution strategies on the rastrigin function: First results. *In*: RUDOLPH, G. et al. (eds). Parallel problem solving from nature – PPSN XVII. PPSN 2022. Lecture Notes in Computer Science, vol. 13399, Springer, Cham, p.499-511, 2022.

PANT, P.; KUMAR, A.; RAM, M. Flower pollination algorithm development: A state of the art review. International Journal of System Assurance Engineering and Management, 8: 1858-1866, 2017.

XUE, F. et al. Optimal parameter settings for bat algorithm. International Journal of Bio-Inspired Computation, 7(2):125-128, 2015.