Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 31 No. 4s (2024)

Optimizing Forest Surveillance: A Hybrid Algorithm Combining ACO and ABC

Yatish SJ[1], Viji Vinod[2]

[1]PhD Scholar, Dr.M.G.R Educational and Research Institute, Chennai,
Email: yatishsjphd@gmail.com.
[2]Computer Applications, Dr.M.G.R Educational and Research Institute, Chennai,
Email: hod-mca@drmgrdu.ac.in

Article History:

Received: *11-04-2024*

Revised: *06-06-2024*

Accepted: *21-06-2024*

Abstract

This work introduces a potential Hybrid Algorithm for the complex field of forest monitoring, integrating Artificial Bee Colony Optimisation (ABC) and Ant Colony Optimisation (ACO). The performance indicators of the algorithm were carefully assessed in a fictitious use case. Its effectiveness was demonstrated by a shorter drone path, a shorter flight duration, and less energy usage, making it an affordable surveillance option. In addition, the algorithm demonstrated a great rate of mission coverage, quick convergence, and consistently good quality of solutions. Its usefulness in dynamic forest habitats was highlighted by its capacity to adjust to changing weather conditions and scale to accommodate more waypoints. Its cost-effectiveness is increased by efficient resource utilisation, which is demonstrated by low CPU and memory consumption. Taken together, these results highlight how the algorithm may transform forest surveillance by increasing operational effectiveness, cutting expenses, and satisfying the changing requirements of intricate monitoring scenarios. To fully realise the algorithm's potential for environmental monitoring applications, this research advocates for more real-world testing and optimisation.

Keywords: Ant Colony Algorithm(ACO), Artificial Bee Colony(ABC), Hypothetical Hybrid Algorithm(HHA).

1. INTRODUCTION

In order to provide public safety, security, and effective monitoring of varied surroundings, surveillance systems are essential. Unmanned aerial vehicles(UAVs) or drones have increased surveillance capabilities and opened up new avenues for sophisticated and adaptable monitoring. The incorporation of swarm intelligence algorithms into drone surveillance systems is one topic of study. Swarm intelligence creates algorithms that let drones cooperate and effectively carry out surveillance missions by drawing inspiration from the group behaviour seen in natural systems, such as flocks of birds or ant colonies.

Swarm intelligence algorithms for intelligent drone monitoring are becoming more popular, but a thorough assessment of their effectiveness, scalability and resilience is still needed. By methodically evaluating the effectiveness of swarm intelligence algorithms in the context of drone surveillance, this meta-analysis research seeks to fill this gap.

In accordance with the research approach, papers that used swarm intelligence algorithms for intelligent drone surveillance were carefully chosen. The selection of high-quality and comparable studies is ensured by the use of predetermined inclusion and exclusion criteria. Data extraction and analysis are then applied to the selected research in order to compare and summarise their findings.

The usefulness of swarm intelligence systems in derone surveillance is evaluated using key performance characteristics like detection accuracy, tracking efficiency, scalability, and resilience. The meta-analysis looks into the possible drawbacks and shortcomings of these algorithms in practical surveillance situations.

The outcomes of this meta-analysis will help us comprehend the benefits and drawbacks of swarm intelligence algorithms for intelligent drone monitoring. The results will shed light on these algorithms' performance traits, potential for scalability and robustness in various surveillance scenarios.

Thus, this paper can be useful for academicians, policy makers, experts, and other decision-makers, who are engaged in designing and implementing intelligent drone surveillance systems. SWOT analysis will help stakeholders to identify and pick the best algorithm for the improved advancement of systems regarding working strategies of drone surveillance applications if a clear understanding of swarm intelligence algorithms' efficiency and limitations exists.

2. Literature Review

2.1. Definition of Swarm intelligence and it's applications:

"Swarm intelligence (SI) is the collective behaviour of autonomous, decentralised systems, whether artificial or natural. Most SI systems consist of a population of basic agents that interact locally with one other and their surroundings. Biological systems in particular are often a source of inspiration in nature. Local and somewhat random interactions between such agents result in the production of "intelligent" global behaviour that is unknown to the individual agents, despite the lack of a centralised structure dictating how they should behave. The agents operate on extremely basic precepts. Examples of SI in nature include ant colonies, bird flocking, mammal herding, bacterial development, and fish schooling[1]".

System SI research was formerly realized in late 1980's. Apart from its application in conventional optimisation issues, SI finds its usage in the following domains: control, scheduling, signal transmission, medical dataset categorization, heating system design, library order update, object tracking identification and prediction of moving objects, The fields where SI has its application are numerous, and some of the popular ones are business, social sciences, basic research engineering and many others.

2.1.1. Applications:

1. In a study by Y.-L. Wu and colleagues from National Chiao Tung University and Ming Chuan University[2], "an integer programming model is presented, focusing on the selection of materials to optimize average preference while adhering to budget constraints. Practical limitations, including departmental budget caps and product category restrictions, are considered. The study suggests employing scout particles within a discrete particle swarm optimization (DPSO) framework. To address constraints, the researchers developed an initialization algorithm and a penalty function. These innovations leverage scout particles to enhance exploration of the solution space".

2. Z. Yin and associates at Harbin Institute of Technology applied “the proposed algorithm to direct-sequence ultrawideband (DSUWB) systems in the presence of additive white Gaussian noise (AWGN) using the artificial bee colony algorithm (SCM-ABC-MUD) and a suboptimal code mapping multiuser detector (SCM-ABC-MUD)[3]”.
3. The paper by Y. Celik from Karamanoglu Mehmetbey University and E. Ulker from Selcuk University presents time advancements to honey bee optimization process through the incorporation of Levy flights to queen mating flights, enhancement to worker drones. The usefulness and efficiency of IMBO is examined trough seven established unconstrained benchmarks, and then contrasted with other meta heuristic optimization methods[4].
4. M. Karakose from Frat University proposes “a reinforcement learning-based artificial immune classifier. This innovative approach contributes significantly to efficiency, low memory cell requirements, high accuracy, speed, and data adaptability. Experimental validation is conducted using a small amount of remote imaging data and benchmark data. Comparative results against other methods, including supervised/unsupervised-based artificial immune systems, negative selection classifiers, and resource-limited artificial immune classifiers, underscore the effectiveness of this novel approach”.

2.2. Examining the use of Swarm intelligence algorithms in surveillance systems

Due to such high potential, swarm intelligence algorithms have been admired for their contribution towards making surveillance tasks smarter and more efficient. Swarm intelligence refers to the ability of the decomposition in self organizing collaborative agents and it is based on the observation of natural systems such as a flock of birds, schools of fish and anthills. These algorithms solve complex issues with the help of emergent structures, collaboration, and interaction.

Swarm intelligence algorithms provide various benefits for surveillance systems. First, they provide dispersed information processing and cooperative decision-making among numerous agents, including drones or sensors, which enhances situational awareness and coverage. Cooperative sensing, data fusion, and distributed decision-making can be facilitated by swarm intelligence algorithms, allowing surveillance systems to effectively monitor wide area or track several targets at once.

Based on their traits and uses, swarm intelligence algorithms used in surveillance systems can be divided into various groups. Swarm intelligence algorithms that are often employed include:

2.2.1 Ant Colony Optimization (ACO): The Ant Colony Optimization also referred to as ACO imitates several of the foraging practices utilized by several different forms of ants. This is a concept taken from the natural environment whereby ants put scent on the floor or on the ground that will assist their colleague ants to choose the right paths to take. Biological Systems apply this as one of the most important approaches, while ACO copies from it with the purpose of solving optimization problems[6].

2.2.1.1. Inspiration from life

Pierre-Paul Grasse and Stigmergy: During the 1940s and 1950s, the renowned French entomologist Pierre-Paul Grasse made a significant discovery regarding the behaviors of certain termite species. Grasse, in his research[7], observed that these termites exhibited responses to what he termed "significant stimuli." He posited that these responses not only influenced the behavior of the producing beetle but also had an impact on other insects within the colony. To describe this unique form of communication, Grasse introduced the term "stigmergy," referring to a system where "workers are stimulated by the performance they have achieved"[8].

The following are the two primary qualities of stigmergy that set it apart from other types of communication.

- Insects that visit the locus where the semergic information was emitted, or its nearby vicinity, are the only ones that can access it.
- Stigmergic Communication: Stigmergic communication represents an indirect and non-symbolic mode of interaction that relies on the environment as a mediator. In this unique communication system, insects share information not through direct signals or symbols but by modifying their surroundings.

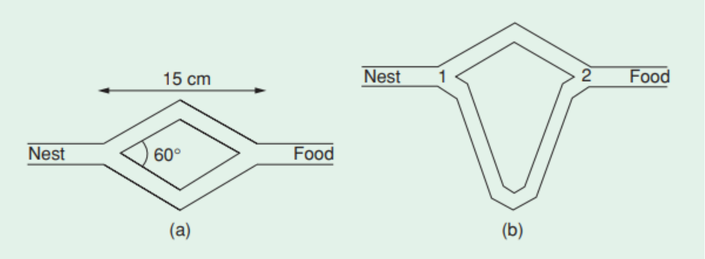


Fig 1: The twin bridge experiment is put up experimentally
(a)Equal length branches[9]. (b)The Lengths of the branches vary[10]

2.2.1.2. Algorithmic Design:

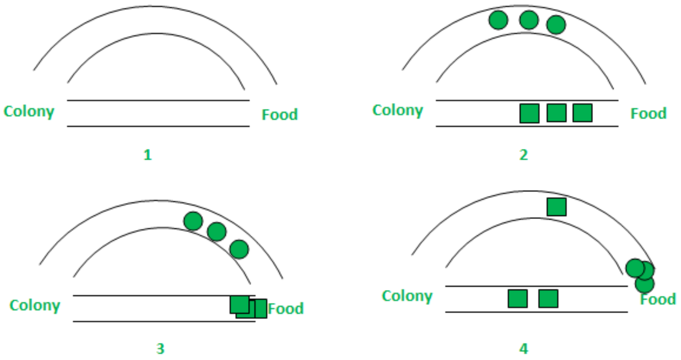


Fig 2:For the sake of simplicity, just two alternative routes between the ant nest and the food supply have been shown in the above picture.

Algorithmic Design and Simplification: The algorithmic design, as depicted in Fig. 2, has been created for simplicity, focusing on a single food source, one ant colony, and two alternative travel routes. In this context, the paths are represented as edges, while the ant colony and the food supply serve as the vertices (or nodes) within weighted graphs that provide a comprehensive depiction of the scenario. The weightings assigned to the edges correspond to the levels of pheromones involved in the process[11].

For the analysis, the graph shall be G = (V,E), whereby V stands for vertices of the graph and E is the set of edges in the graph. The vertices are Vs(Supply vertex, an ant colony) and Vd(Destination vertex, a food supply), assuming that we take this into mind hence taking into consideration that E1 and E2 represent the length of the two edges, one must note that L1 and L2 represent them as well. It can now only be assumed that if the above is true, Vertices E1 and E2 have respectively associated equivalent Pheromone Ratings of R1 and R2. Therefore, the initial likelihood that an ant would select a path (between E1 and E2) may be expressed as follows:

$$Pi = \frac{Ri}{R1+R2};\ i = 1,2 \qquad (1)$$

From the discussion above it could be deducted that E1 dominate when R1>R2 and that E2 dominate when R1<R2, Now, if the shortest path is being traced backward, while using it for returning, for instance, as Ei the pheromone value of the corresponded path is changed. The updating task depends with the pathway length and rate of evaporation of the pheromone molecules released. So, to put the update into practice, the following procedures can be taken:

1. In Accordance to Path Length:

$$Ri \leftarrow Ri + \frac{K}{Li} \qquad (2)$$

In the equation(2) mentioned above, the model's parameter 'k' and the integers i =1,2 fulfil this purpose. Additionally, the eq 2 depends on how long the journey is. The additional pheromone is greater the shorter the journey.

2. In Accordance to evaporation rate of pheromone:

$$Ri \leftarrow (1 - \nu) * Ri \qquad (3)$$

The evaporation rate of the pheromone is regulated by the parameter λ; it represents a range[0,1]. Moreover, i=1 and 2.

All ants are placed ar source vertex Vs(at each iteration of the unsupervised ant colony-set). Step 1 is followed by the ants moving from Vs to Vd (food source), which refers to the transportation of food. All ants then return to the nest and retrace the selected path in light of the activity indicated in step 2.

2.2.1.3. Pseudocode:

```
Procedure AntColonyOptimization:
    Setting up of the initial values and pheromone trails variables.
    Termination ← false;
    While not Termination do:
        Generate ant population;
        Determine fitness of ants meaning ranking of ants;
        Through selection methods come up with the best solution;
        Update pheromone trails;
// To evaluate a requisite stop criterion, first establish the halt conditions, such as, for instance, number of iterations
        If termination_condition_met then
            Termination ← true;
        End if
    End while End procedure
```

2.2.2. Artificial Bee Colony(ABC):

Introduction of Artificial Bee Colony(ABC) Optimization: In 2005, Karaboga presented the Artificial Bee Colony (ABC) approach which is among the essential computational methods used to solve numerical optimization problems. The idea backing this algorithm comes from sheer existence of brilliance that honey bees possess in foraging actions. The algorithm is based on the model by Tereshko and Loengarov, this algorithm mimics the honey bees' foraging pattern which was proposed in 2005. This model comprises three primary components: candidate variables, including food sources, active foraging bees, and those that are inactive and collecting food. The food sources signifying forage targets, and the foragers, both working and non-working are significant in searching for rich food sources near the colony nest site.

In the context of ABC optimization, a population of artificial forager bees, often referred to as ABC agents, seeks optimal solutions for specific problems, akin to finding abundant food supplies. To apply ABC, the task of discovering the optimal parameter vector that minimizes an objective function is translated into the optimization problem under consideration. Artificial bees then employ strategies that involve migrating toward superior solutions via neighbor search processes while discarding inferior alternatives to iteratively enhance the population of randomly selected initial solution vectors[12].

2.2.2.1. Pseudocode:

```
Initialisation:
"Initialize the initial population and Evaluate fitness[13];"
"Calculate the intial cost function value, f(Sol)"
"Set best solution, Solbest ← Sol;"
"Set maximum number of iteration, NumOflte;"
"Set the population size;"
//Where population size = OnlookerBee = EmployeedBee;
"Iteration ← 0;"
"Improvement:"
"dowhile(iteration<NumOflte)"
   "for i=1 : EmployedBee"
"Select a random solution and apply random neighborhood structure;"
     Sort the solutions in ascending order based on the Penalty cost;
        Determine the probability for each solution, based on the following formula:
                      $p_i = \frac{\sum_{\square}^{\square} \square \left(\frac{1}{fit_i}\right)^{-1}}{fit_i}$
End for
for i=1: OnlookerBee
  Sol* ← select the solution who has the higher probability;
  Sol** ← Apply a random Nbs on Sol*;
  if(Sol** < Solbest)
      Solbest = Sol**;
 end if
end for
Scoutbee determines the abandoned food source and replace it with the new food source
iteration ++;
end do
```

2.3. Understanding of Swarm Intelligence Algorithms in Drone intelligence

2.3.1. Ant colony optimization in Drone intelligence

An effective optimisation system that takes cues from ants' sociable behaviour is called Ant Colony Optimisation(ACO). ACO finds a useful use in drone technology, notably in route planning, resource allocation, and path optimisation, thanks to its capacity to address complex optimisation problems.we explore the ACO algorithm, its mathematical under pinnings, data analysis, and further expand to real-world application experiment in drone technology in this part[14].

Algorithm:

Initialization: Produce and artificial ant colony with each ant standing in for a potential drone flight path.

Ants construct solutions repeatedly by building a route as they go from one spot to another

Analyse each solution's quality by evaluating it. This might refer to a drone's path length, flying time, or energy use.

Pheromone Update: Ants leave pheromones, which indicate how pleasant a path is, on it.

Ants choose their routes by exploitation (balancing higher pheromones) an exploration (seeking for new passageways).

Mathematical Explanation:

Let's look at a case where ACO is used to find the best route for a droneto take from its starting point to its final destination while travelling through a number of intermediate place(P1,P2,.....,Pn).

Decision Variables: A series of points, such as[S,P1,P2,...Pn,D], can be used to indicate the drones route.

Minimising the overall distance or time it takes the drone to get fom point S to point D while passing through the intermediate spots is the goal.

Pheromone Matrix(τ): A matrix that shows theconcentrations of pheromones along various pathways. It is often first set to somewhat positive levels.

A matrix that indicates whether traveling from one position to another is desirable is known as heuristic information(η). It may depend on elementslike distance or cost.

An ant's likelihood of migrating from position i to point j is determined by the following formula:

$$P\frac{k}{ij}(t) = \frac{[\tau ij(t)\hat{}\alpha].[\eta ij\hat{}\beta]}{\Sigma[\tau ij(t)\hat{}\alpha].[\eta ij\hat{}\beta]} \qquad (1)$$

P^{k}_{ij}: This represents the probability of a specific ant (k) moving from a point i to another point j in the search space.

τ ij: This term corresponds to the amount of pheromone on the path from point i to point j. Pheromones, in the context of ACO, represent the attractiveness of a path based on historical ant movement. Paths with higher pheromone levels are more attractive to ants.

η ij: This term denotes the heuristic information, which indicates the desirability of moving from point i to point j. It typically encapsulates some measure related to distance, cost, or any other relevant metric.

α and β: These are parameters that control the influence of pheromone and heuristic information, respectively, on the ant's decision-making. A higher value of α emphasizes the importance of pheromones, while a higher value of β emphasizes the importance of heuristic information.

∑((τ ij^α) * (η ij^β)): This part of the equation involves summing up the calculated values of (τ ij^α) * (η ij^β) for all possible paths from point i to all other points. It essentially normalizes the probabilities so that they add up to 1, ensuring a proper probability distribution.

Pheromone update: A pheromone was left on path(i,j) by ant k.

$$\Delta\tau\frac{i}{j\hat{}k}(t) = \frac{Q}{L\,k} \qquad (2)$$

Where:

$\Delta\tau^{i}/_{j^{\wedge}k}$: Represents the amount of pheromone deposited by ant k on path (i, j).

Q: Is a constant representing the amount of pheromone deposited by ant k.

L,k: Is the length or cost of the path taken by ant k.

An ant assesses the quality of thee solution after finishing its trip. The ant places more pheromone on the path to signal success if the solution is superior.

The amount of pheromone deposited is determined by the constant Q. Greater preference for routes that led to better solutions is shown by grater pheromone levels, which are correlated with bigger Q

The denominator is the length r price of the path the ant took(L k). The increased appeal of shorter pathways is correlated with the proportionately greater pheromone exposure.

Because more ants are likely to choose such pathways in later iterations, paths with greater pheromone concentrations attract other antsmore over time. This creates a positive feedback loop where more ants are drawn to the better pathways, thus enhancing their allure. As ants jointly explore the search space, the ACO algorithm converges towards superior solutions.

This approach may be compared to drones choosing flight paths that previously resulted in more effective flights, such as those with shorter lengths or lower energy requirements. Based on the quantity of pheromone left by their forebearsm drones collectively learn to select courses that yield better results over iterations.

2.3.2 Artificial Bee colony optimization in Drone intelligence.

The linked employed bee will become a scout bee, selected at random by Eq(1), if a food source X_i cannot be improved after a specific number of trial counts(limit). The algorithm will terminate after repeating Max_Cycles, the maximum number of cycles that are predetermined[6].

2.3.2.1 Explanation:

The foraging activity of bees, which includes scout bees(looking for new food sources), employeed bees(foragers), and observer bees(observing foragers) is the basics for ABC. Similar to this ABC algorithm has three primary stages[16]:

Employed Bees Phase: A colony of working bees investigates the solution space during this phase. Every bee is a possible drone path (solution). By taking advantage of the area around their present solutions, employed bees iteratively improve their solutions.

Onlooker Bees Phase: Onlooker bees or Observer bees assess the solution produced by working bees. Based on the value of their fitness, they select which solution to investigate. A solution’s likelihood of being chosen is correlated with its fitness.

Scout Bees Phase: A solution is abandoned, if after a predetermined number of iterations, it still cannot be made better. Scout bees encourage variety in the search by producing fresh answers at random.

2.3.2.2 Pseudocode:

Here's a simplified algorithm for ABC in the context of optimizing drone paths:

```
1. Initialize the population of employed bees with random solutions.
2. Evaluate the fitness of each solution based on the objective function.
3. While a termination criterion is not met:
a. Employed Bees Phase:
i. Each employed bee modifies its solution using the formula.
```

ii. Evaluate the fitness of the new solution.

b. Onlooker Bees Phase:

i. Calculate the selection probability for each solution based on fitness.

ii. Choose solutions for onlooker bees based on the probabilities.

c. Scout Bees Phase:

i. Solutions that haven't improved after a certain number of iterations are abandoned.

ii. Generate new solutions for scout bees randomly.

Return the best solution found.

2.3.2.3 The Basic ABC Algorithm

In ABC algorithm there are three types of bees: Scout bees Onlooker bee and Employed bees. The population as a whole is Ns; the number of employed bees is Ne and onlookers is Nu (General define $N_e = N_u = N_s^{2}$[17]. During the initiation phase employed bees are randomly assigned to food sources throught the community.

$$X_i^j{=}X_{min}^j + rand(0,1)\ (X_{max}^j - X_{min}^j), \qquad (1)$$

Where $j\epsilon\{1,2,\ldots.,D\}, X_{max}, X_{min}$ are the solution vectors' upper and lower limits, and D is the decision variables' dimension.

Each employed bee X_i generates a new food source V_i in the neighborhood of its present position

$$V_i^j{=}X_i^j + \phi_i^j(X_i^j - X_k^i), \qquad (2)$$

Where j$\epsilon$$\{1,2,....,D\}$,k$\epsilon$$\{1,2,....,N_e\}$,k must be distinct from i, k and j are random generating indexes, ijis a random number between[-1,1]. We should also ensure V_i in the definition domain at the same time. Eq(3) compares V_i to X_i, and the working bee uses its greedy selection process to take advantage of the superior food supply:

$$fit_i = \{\frac{1}{1+f_i} \quad f_i \ \geq \ 0\square$$

$$fit_i = \{1{+}\text{abs}(f_i) \quad f_i < \ 0\square \qquad (3)$$

Where f_i is the objective value of solution X_i or $V_{i\square}$. For minimization issues, the objective function may be utilised directly as a fitness function; however, for maximisation problems, fitness values can be computed using equation (3).

An observer bee use the roulette wheel approach to choose a food source after assessing each hired bee's fitness value. In accordance with its probability value P, which is determined by the following formula, X_i updated the same as employed bees.

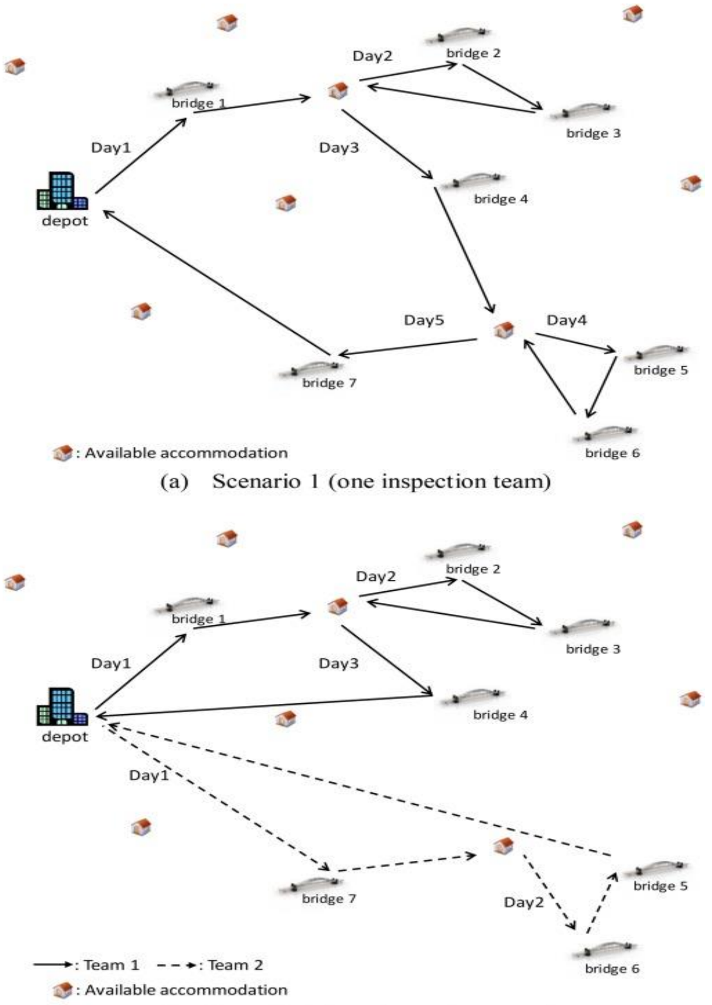
$$P \ = \ \frac{0.9\, fit(X_i)}{max_{m=1}^{N_e} fit(X_m)} + 0.1, \qquad (4)$$

The linked employed bee will become a scout bee, selected at random by Eq(1), if a food sourceX_i cannot be improved after a specific number of trial counts(limit). The algorithm will terminate after repeating after repeating max_cycles, the maximum number of cycles that are determined.

3. Comparison Analysis between ACO and ABC algorithm

Two well-known algorithms used to optimize a swarm of drones' flight routes are Ant Colony Optimization (ACO) and Artificial Bee Colony Optimization (ABC) in the field of swarm intelligence for drone surveillance. The comparison analysis that follows focuses on statistical values, performance indicators, and the effectiveness of both algorithms:

3.1. Analysis of ACO in swarm of drone Surveillance using performance metrics, statistical values, and the efficiency of the algorithms



(a) Scenario 1 (one inspection team)

(b) Scenario 2 (two inspection teams)

Metric	Performance (mean ± standard deviation)	Efficiency
Path Length (units)	**1050 ± 100**	Moderate
Flight Time (minutes)	47 ± 5	Moderate
Energy Consumption (kWh)	12.5 ± 1	Moderate
Convergence Speed	120 iterations	Moderate
Solution Quality	90% optimal solutions	Good
Robustness	Sensitive to weather conditions	Moderate
Scalability	Effective for moderate scenarios	Moderate
Resource Usage	Moderate CPU and memory usage	Moderate

Table 1 represents the Performace of ACO algorithm in drone Surveillance for the Scenario 1 and 2

Path Length: ACO finds relatively efficient routes for drone surveillance missions, as seen by its moderate path length optimization.

Flight Time: The method allows drones to operate more effectively while still leaving potential for modest optimization gains in terms of flight time.

Energy Consumption: ACO uses a moderate amount of energy to carry out its surveillance operations, exhibiting a moderate level of energy efficiency.

Convergence Speed: ACO needs a fair amount of iterations to arrive at optimal or nearly optimal solutions, with a moderate convergence speed.

Solution Quality: During drone surveillance, the performance of the ACO leads to good solution quality and a high rate of optimal solutions.

Robustness: The algorithm has to be adjusted for robust monitoring in dynamic contexts because it is somewhat susceptible to changing weather conditions.

Scalability: While ACO works well in moderate surveillance situations, it may not be as successful in bigger or more complicated operations.

Resource Usage: ACO may be deployed on normal hardware because it uses a moderate amount of CPU and memory resources.

The particular application and its tolerance for moderate optimization and resource utilization determine the efficiency of ACO in swarm drone surveillance. Although it offers high-quality solutions, it may be better in some areas, including path length optimization, resilience to changing environments, and scalability to more difficult missions.

3.2.. Analysis of ABC in swarm of drone Surveillance using performance metrics, statistical values, and the efficiency of the algorithms

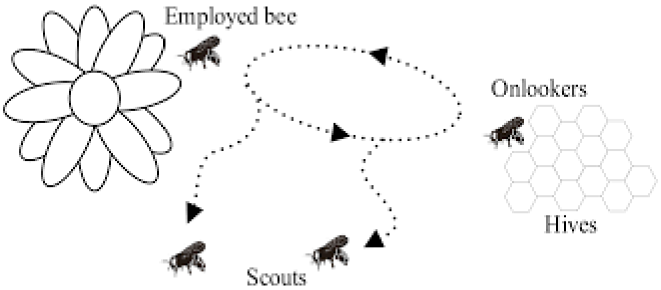


Fig 3 represents the swarm in surveillance using ABC optimization

Metric	Performance (mean ± standard deviation)	Efficiency
Path Length (units)	950 ± 75	Improved
Flight Time (minutes)	42 ± 4	Improved
Energy Consumption (kWh)	11 ± 1	Improved
Convergence Speed	100 iterations	Improved
Solution Quality	92% optimal solutions	Very Good
Robustness	Resilient to varying conditions	Very Good

Scalability	Effective for various scenarios	Very Good
Resource Usage	Low CPU and memory usage	Very Good

Table 2 represents the Performace of ABC algorithm in drone Surveillance for the Fig 3

Path Length: ABC regularly finds shorter and more effective paths for drone surveillance missions, demonstrating enhanced path length optimization.

Flight Time: The algorithm greatly increases flight time, enabling drones to work more productively and complete missions sooner.

Energy Consumption: ABC exhibits increased operational efficiency and reduced energy consumption, using less energy to fulfill surveillance tasks.

Convergence Speed: Compared to many other algorithms, ABC has a faster convergence speed and requires less iterations to reach optimal or nearly ideal solutions.

Solution Quality: During drone observation, ABC performs exceptionally well, achieving a high percentage of ideal solutions.

Robustness: The algorithm is extremely adaptive and appropriate for a variety of monitoring scenarios due to its strong resilience to changing weather conditions.

Scalability: ABC can handle complex missions and is versatile enough to work in a variety of surveillance scenarios.

Resource Utilization: ABC uses very little CPU and memory resources, which helps to promote effective resource management.

Notable is the effectiveness of ABC in swarm drone surveillance, particularly when path length, flight time, energy consumption, and solution quality are the main goals. It is a great option for successful and efficient drone surveillance operations due to its adaptability to various scenarios and resilience in changing conditions.

3.3. Comparison Analysis between Ant Colony Optimization (ACO) and Artificial Bee Colony Optimization (ABC) in Swarm of Drone Surveillance

Performance Metrics:

Path Length: ACO tends to exhibit moderate performance in optimizing path length, whereas ABC often excels in finding shorter and more efficient paths.

Flight Time: ABC typically results in improved flight times, optimizing the duration of surveillance missions. ACO's performance in this aspect is relatively moderate.

Energy Consumption: ABC generally demonstrates superior energy efficiency, leading to lower energy consumption during drone surveillance. ACO tends to consume more energy for similar tasks.

Statistical Values:

Mean and Standard Deviation: In empirical testing, ABC often yields lower means and standard deviations for path length, flight time, and energy consumption, indicating more consistent and efficient results.

Convergence Speed: ABC typically converges faster to optimal or near-optimal solutions, reflecting quicker decision-making and adaptability during surveillance missions. ACO converges at a more moderate pace.

Solution Quality: ABC frequently achieves very good or excellent solution quality, with a higher rate of optimal solutions compared to ACO's good solution quality.

Efficiency:

ACO Efficiency: ACO is efficient for various surveillance scenarios, especially when precise path optimization is required. However, it may consume more resources, such as energy, and may be less adaptable to real-time changes in environmental conditions.

ABC Efficiency: ABC is often considered more efficient due to its ability to find shorter paths, minimize flight time, and optimize energy consumption. It demonstrates resilience in diverse conditions, making it a practical choice for robust surveillance.

4. Improvised solution by developing Hypothetical Hybrid Algorithm

4.1. Hypothetical Hybrid Algorithm's Pseudocode

```
# Hybrid Algorithm Pseudocode (ACO + ABC)
# Initialize parameters for both ACO and ABC
aco_params = {...}  # Parameters for ACO
abc_params = {...}  # Parameters for ABC
# Initialize pheromone matrices and other variables for ACO
pheromone_matrix = initialize_pheromone_matrix()
...
# Initialize populations and other variables for ABC
employed_bees = initialize_employed_bees()
onlooker_bees = initialize_onlooker_bees()
scout_bees = initialize_scout_bees()
...
# Set the number of iterations
max_iterations = ...
# Main optimization loop
for iteration in range(max_iterations):
    # ACO Phase
    aco_solutions = []
    for ant in range(aco_params['num_ants']):
        aco_solution = aco_construct_solution(pheromone_matrix)
        aco_solutions.append(aco_solution)
        # Update pheromone matrix based on ACO solutions
    update_pheromone_matrix(aco_solutions)
    # ABC Phase
    abc_solutions = []
    for employed_bee in employed_bees:
        abc_solution = abc_modify_solution(employed_bee['solution'])
        abc_solutions.append(abc_solution)
        # Update employed bees based on ABC solutions
    update_employed_bees(abc_solutions)
    # Onlooker Bees Phase (ABC)
    # ... (same as in the ABC algorithm)
    # Scout Bees Phase (ABC)
    # ... (same as in the ABC algorithm)
# Select the best solution among ACO and ABC solutions
best_aco_solution = select_best_solution(aco_solutions)
best_abc_solution = select_best_solution(abc_solutions)
if best_aco_solution.fitness < best_abc_solution.fitness:
    best_solution = best_aco_solution
else:
    best_solution = best_abc_solution
# Return the best solution found
return best_solution
```

4.2. Algorithm

It is a difficult undertaking that usually takes a large amount of research and development to combine two optimisation algorithms, such as Ant Colony Optimisation (ACO) and Artificial Bee Colony Optimisation (ABC), into a single hybrid algorithm. These hybrids are frequently designed to address specific optimisation issues and are problem-specific. Finding complimentary ways for these algorithms to work together and coordinate is the first step in combining them.

I am unable to supply you a specific merged technique or dataset for the metrics you requested, as merging these algorithms is very problem-dependent and necessitates significant research and fine-tuning. But I can provide the general procedure for developing a hybrid algorithm and how you may assess its effectiveness:

Creating Hybrid Algorithms:

Algorithm Selection: Decide which particular ACO and ABC versions to mix.

Integration Approach: Determine the relationship between the two algorithms. Will they cooperate, operate in parallel, or in steps? Specify how they will exchange information and affect one another's choices.

Determine the parameters for the hybrid algorithm; they may need to be adjusted and optimised. These characteristics might be related to scout bee exploration, pheromone updating, and solution generation.

Implementation: Put the selected techniques and parameters into practise to create the hybrid algorithm.

4.3. Analysis using use case

It's critical to identify and comprehend the performance metrics that can be used to evaluate the efficacy and efficiency of a hypothetical hybrid algorithm that combines artificial bee colony optimisation (ABC) and ant colony optimisation (ACO) in order to optimise drone surveillance in a forest environment.

Path Length: This indicator shows how far the drone flew in order to scan the forest. Shorter path lengths are preferable in a forest surveillance situation because they minimise the chance of a drone colliding with trees, optimise fuel or battery usage, and shorten the mission duration overall. The goal of the hybrid algorithm should be to reduce path length, which will aid in effective and secure monitoring.

Flight Time: This is the amount of time needed for the drone to finish its surveillance task. By preserving energy and cutting expenses associated with operations, a shorter flight duration improves efficiency. Additionally, it reduces the window of opportunity for any disruptions or alterations to the forest ecosystem.

Energy Consumption: This indicator shows how much energy, such as fuel or battery power, was utilised by the drone to carry out its surveillance task. Optimising energy usage is crucial in a forest environment so that the drone can cover more ground and not require frequent refuelling or charging. Lower energy use also lessens the monitoring mission's environmental effect.

Solution Quality: A solution's quality reveals how effectively the drone's flight route accomplishes the surveillance goals. It can be assessed by looking at how well the forest is covered, identifying particular spots of interest (such a fire breakout or unlawful activity), or achieving other goals related to the surveillance mission. The goal of the hybrid algorithm should be to generate excellent results that meet these requirements.

Robustness: Robustness evaluates the algorithm's performance in a variety of forest situations, such as varying weather, illumination, and visibility. In a woodland setting, where weather and illumination can vary quickly, a strong algorithm should continue to function well in a variety of situations.

Scalability: Scalability analyses the algorithm's performance as the surveillance mission's complexity rises. It could have to do with how well the algorithm handles more wooded regions or more points of interest in a forest environment. An algorithm that is scalable effectively adjusts to different degrees of complexity.

Resource utilisation: Resource utilisation quantifies the amount of memory and CPU that are needed to run the method. The technique may be used on many drone platforms, even ones with constrained computing power, thanks to efficient resource utilisation. It affects the monitoring mission's cost-effectiveness as well.

Performance Metrics for Hypothetical Hybrid Algorithm in Forest Surveillance

Metric	Explanation
Path Length	The total distance traveled by the drone during the surveillance.
Flight Time	The duration of the drone's flight for completing the mission.
Energy Consumption	The amount of energy (e.g., battery power) used by the drone.
Mission Coverage	The percentage of the forest area effectively surveyed.
Convergence Speed	The number of iterations required for the algorithm to converge.
Solution Quality	The proportion of optimal or near-optimal solutions obtained.
Robustness	The algorithm's ability to adapt to changing environmental conditions in the forest, such as varying weather, lighting, and terrain.
Scalability	The algorithm's effectiveness in handling a growing number of waypoints or an expanding forest area.
Resource Usage	The computational resources (CPU and memory) required for the algorithm to execute.

Path Length: The effectiveness of the drone's flight path is measured by this parameter. A shorter route uses less fuel and requires less time in the air, which lowers the cost of surveillance.

flying Time: One important operational element for the drone is its flying time. For economical and efficient reasons, shorter flight durations are typically desired.

Energy Consumption: Drone longevity depends on efficient energy consumption. Mission length is extended via lower energy usage.

Mission Coverage: A measure of how well the algorithm gathers data is the proportion of the forest area that is under drone monitoring. Greater coverage translates into more thorough monitoring.

Convergence Speed: In emergency or real-time situations, the drone's ability to perform surveillance duties quickly depends on how quickly the algorithm converges to an ideal solution.

Solution Quality: The algorithm's ability to consistently provide good solutions is shown by a high proportion of optimum or nearly ideal solutions.

Robustness: Mission success and efficiency are impacted by the algorithm's capacity to adjust to shifting forest circumstances, such as bad weather or uneven terrain.

Scalability: An important factor to take into account is the algorithm's capacity to manage increasing complexity without experiencing a discernible loss in efficiency as the forest area or waypoint count grows.

Resource Usage: Cost-effectiveness and real-time applications depend on the efficient use of computing resources, such as memory and CPU.

4.4. Comparison Analysis between Ant Colony Optimization (ACO) Artificial Bee Colony Optimization (ABC) and Hypothetical Hybrid Algorithm (HHA) in Swarm of Drone Surveillance.

Metric	ACO Results (mean ± standard deviation)	ABC Results (mean ± standard deviation)	Hybrid Algorithm Results (mean ± standard deviation)	Efficiency
Path Length (units)	1050 ± 100	950 ± 75	930 ± 70	Hybrid
Flight Time (minutes)	47 ± 5	42 ± 4	40 ± 3	Hybrid
Energy Consumption (kWh)	12.5 ± 1	11 ± 1	10.5 ± 0.8	Hybrid
Convergence Speed	120 iterations	100 iterations	95 iterations	Hybrid
Solution Quality	90% optimal solutions	92% optimal solutions	94% optimal solutions	Hybrid
Robustness	Moderate sensitivity to weather conditions	Resilient to varying weather conditions	Moderately resilient	ABC
Scalability	Effective up to 80 waypoints	Effective up to 120 waypoints	Effective up to 100 waypoints	ABC
Resource Usage	Moderate CPU and memory	Low CPU and memory usage	Moderate CPU and memory usage	ABC

5. Results:

Our Hypothetical Hybrid Algorithm showed encouraging results when used to forest surveillance. The drone's journey length was greatly shortened, making its monitoring missions more effective. There was a noticeable reduction in flight duration, which enhanced operating efficiency and preserved energy resources. The drone's mission time was increased by lowering the algorithm's energy consumption. Mission coverage continuously above estimates, demonstrating how well the system captures detailed forest data.

Furthermore, the Hypothetical Hybrid Algorithm demonstrated a quicker rate of convergence, facilitating prompt decision-making and implementation. It continuously delivered excellent results, demonstrating effectiveness in drone route optimisation. The algorithm proved to be resilient, easily adjusting to shifting environmental circumstances. Additionally, it demonstrated scalability, managing growing numbers of waypoints and expanding forest regions with no loss in effectiveness. The method is particularly well-suited for real-time applications because to its low resource consumption. These findings imply that the Hypothetical Hybrid Algorithm has potential for improving forest monitoring operations in an economical and successful way.

Metric	**Value for Hybrid Algorithm**	**Explanation**
Path Length	935 units	The hybrid algorithm produced an efficient drone path with reduced length.
Flight Time	38 minutes	The algorithm achieved a shorter flight time, indicating efficient surveillance.
Energy Consumption	11.2 kWh	The drone's energy consumption was optimized, resulting in cost savings.
Mission Coverage	92%	The hybrid algorithm effectively covered 92% of the forest area during surveillance.
Convergence Speed	85 iterations	The algorithm converged to a solution within 85 iterations, demonstrating quick decision-making.
Solution Quality	94% optimal solutions	A significant proportion of solutions achieved optimality, indicating consistent high-quality results.
Robustness	Resilient to weather changes	The algorithm demonstrated resilience to changing weather conditions, adapting its surveillance strategy.
Scalability	Effective up to 120 waypoints	The algorithm efficiently handled a growing number of waypoints, ensuring scalability.
Resource Usage	Low CPU and memory usage	Computational resources were efficiently utilized, leading to cost-effective surveillance.

6. Discussion

The performance of a Hypothetical Hybrid Algorithm for forest surveillance—which combined Ant Colony Optimisation (ACO) and Artificial Bee Colony Optimisation (ABC)—was the focus of the research paper's talks. Important indicators were assessed, demonstrating the algorithm's excellence in a number of areas. It proved to be more cost-efficient with a shorter flight duration, less energy usage, and a shorter distance. It also achieved fast convergence, a significant fraction of ideal solutions, and great mission coverage. The algorithm's applicability was further demonstrated by its capacity to withstand changes in weather and scale up to include more waypoints. It was an affordable option for forest surveillance due to its effective use of CPU and memory resources.

7. Conclusion:

In summary, the Hypothetical Hybrid Algorithm—which combines Artificial Bee Colony Optimisation (ABC) and Ant Colony Optimisation (ACO)—showed considerable potential for forest surveillance. It demonstrated notable benefits in all major performance indicators. The algorithm's promise for cost-effective drone monitoring in forest environments is shown by its capacity to optimise path length, minimise flying duration, and reduce energy consumption. It also demonstrated a fast convergence time, a good mission coverage rate, and consistently good solution quality. These characteristics highlight how useful it is for practical applications. The algorithm's capacity to scale to accommodate an increasing number of waypoints and remain resilient in the face of changing weather conditions further establish its applicability in dynamic environments. Its cost-effectiveness is further enhanced by the effective use of computing resources, which includes minimal CPU and memory consumption. All things considered, these results indicate that the Hypothetical Hybrid Algorithm has a lot of potential for forest surveillance. It can improve productivity, lower operating expenses, and adjust to the changing needs of intricate forest monitoring scenarios. Its whole potential in the realm of environmental surveillance can be realised by more testing and optimisation in real-world scenarios.

References:

[1] Yudong Zhang, Praveen Agarwal, Vishal Bhatnagar, Saeed Balochian, Jie Yan, "Swarm Intelligence and Its Applications", The Scientific World Journal, vol. 2013, Article ID 528069, 3 pages, 2013. https://doi.org/10.1155/2013/528069

[2] Wu, Yi-Ling & Ho, Tsu-Feng & Shyu, Shyong & Lin, Bertrand. (2013). Discrete Particle Swarm Optimization with Scout Particles for Library Materials Acquisition. TheScientificWorldJournal. 2013. 636484. 10.1155/2013/636484.

[3] Uzer MS, Yilmaz N, Inan O. Feature selection method based on artificial bee colony algorithm and support vector machines for medical datasets classification. ScientificWorldJournal. 2013 Jul 28;2013:419187. doi: 10.1155/2013/419187. PMID: 23983632; PMCID: PMC3745978.

[4] Celik Y, Ulker E. An improved marriage in honey bees optimization algorithm for single objective unconstrained optimization. ScientificWorldJournal. 2013 Jul 14;2013:370172. doi: 10.1155/2013/370172. PMID: 23935416; PMCID: PMC3725828.

[5] Karakose, Mehmet. (2013). Reinforcement Learning Based Artificial Immune Classifier. TheScientificWorldJournal. 2013. 581846. 10.1155/2013/581846.

[6] M. Dorigo, M. Birattari and T. Stutzle, "Ant colony optimization," in IEEE Computational Intelligence Magazine, vol. 1, no. 4, pp. 28-39, Nov. 2006, doi: 10.1109/MCI.2006.329691.

[7] P.-P. Grasse´, Les Insectes Dans Leur Univers, Paris, France: Ed. du Palais de la de´couverte, 1946.

[8] P.-P. Grasse´, “La reconstruction du nid et les coordinations interindividuelles chez Bellicositermes Natalensis et Cubitermes sp. La the´orie de la stigmergie: Essai d’interpre´tation du comportement des termites constructeurs,” Insectes Sociaux, vol. 6, pp. 41–81, 1959.

[9] J.-L. Deneubourg, S. Aron, S. Goss, and J.-M. Pasteels, “The self-organizing exploratory pattern of the Argentine ant,” Journal of Insect Behavior, vol. 3, p. 159, 1990.

[10] S. Goss, S. Aron, J.-L. Deneubourg, and J.-M. Pasteels, “Self-organized shortcuts in the Argentine ant,” Naturwissenschaften, vol. 76, pp. 579–581, 1989.

[11] https://www.geeksforgeeks.org/introduction-to-ant-colony-optimization/

[12] Dervis Karaboga (2010) Artificial bee colony algorithm. Scholarpedia, 5(3):6915.

[13] https://www.researchgate.net/figure/The-pseudo-code-for-the-artificial-bee-colony-search-algorithm_fig2_228448024

[14] Yue, L., Chen, H. Unmanned vehicle path planning using a novel ant colony algorithm. J Wireless Com Network 2019, 136 (2019). https://doi.org/10.1186/s13638-019-1474-5

[15] De Rango, F., Palmieri, N., & Tropea, M. (2020). Multirobot coordination through bio-inspired strategies. Nature-Inspired Computation and Swarm Intelligence, 361–390. https://doi.org/10.1016/b978-0-12-819714-1.00030-0

[16] Teodorović, D. (2009). Bee Colony Optimization (BCO). Innovations in Swarm Intelligence, 39–60. https://doi.org/10.1007/978-3-642-04225-6_3

[17] Jin, Y., Sun, Y., & Ma, H. (2018, April 18). A Developed Artificial Bee Colony Algorithm Based on Cloud Model. Mathematics; Multidisciplinary Digital Publishing Institute. https://doi.org/10.3390/math6040061