International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/50 Volume 12, Issue 5, September – October (2025)

RESEARCH ARTICLE

HACSA: A Hybrid A* and Crow Search Algorithm for Swarm-UAV Network Path Planning in Dynamic Environments

Sheveta Vashisht
School of Computer Science and Engineering, Lovely Professional University, Phagwara, Punjab, India.
sheveta.16856@lpu.co.in

Avinash Kaur
School of Computer Science and Engineering, Lovely Professional University, Phagwara, Punjab, India.
✉ avinash.14557@lpu.co.in

Parminder Singh
School of Computer Science and Engineering, Lovely Professional University, Phagwara, Punjab, India.
parminder.16479@lpu.co.in

Ranbir Singh Batth
Sydney International School of Technology and Commerce, Australia.
ranbir.b@sistc.edu.au

Received: 23 July 2025 / Revised: 07 October 2025 / Accepted: 20 October 2025 / Published: 30 October 2025

Abstract – Path planning is an essential need for the swarm UAVs in order to make them safe and efficient in uncertain settings. In order to enhance the successful avoidance of hindrances in the process of undertaking tasks, UAVs need optimal paths to optimize path length, smoothness, and real-time adaptability. Conventional heuristics like A* can be of high reliability in a static environment but not responsive in a dynamic one, whereas nature-inspired metaheuristics like CSA are very adaptive at the expense of increased computation complexity. In order to overcome such shortcomings, this paper will introduce Hybrid A* and Crow Search Algorithm (HACSA), a method that incorporates the deterministic accuracy of A* with the dynamic search ability of Crow Search Algorithm (CSA). HACSA creates possible initial paths and further evolves them iteratively to avoid obstacles, swarm cohesion, and dynamical responsiveness. Simulation findings indicate that HACSA reduces the path length by 12 percent, smoothness by 15 percent, travel time by 10 percent, collision rates by almost zero, swarm cohesion by 18 percent, and execution time by 20 percent over the baseline approaches. The above results affirm HACSA as a sufficient and effective method of multi-UAV path planning in changing obstacle conditions.

Index Terms – UAV Swarm, Path Planning, Hybrid A* Algorithm, Crow Search Algorithm, Dynamic Obstacles, Collision Avoidance.

1. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) working in swarm configurations are finding more application in various tasks including environmental surveying, disaster recovery, surveillance and inspection of infrastructure [1-3]. Swarm UAVs find it very appealing due to their ability to cover vast regions, redundancy and flexibility in responding to situations on the ground. Nonetheless, it has been observed that planning path of such swarms is still a significant problem, especially in changing environments where stationary and moving obstacles are needed to be taken into consideration but swarm cohesion and inter-UAV spacing is observed at a safe distance [4, 5]. Addressing such challenges is also important in making safe, efficient, and mission-fulfilling swarm operations in safety-critical areas possible.

Deterministic planners like A* algorithm have faced long preference in the field of robotics since they produce an optimal and collision free path in a non-moving environment [6]. However, their performance suffers greatly when there is a shift in the obstacles or when re-planning frequently needs to be undertaken. Conversely, other meta-heuristic algorithms like the Crow Search Algorithm (CSA) resembles smart scavenging by taking advantage of the use of memory to

RESEARCH ARTICLE

adapt [7, 8]. This gives CSA strong adaptability in dynamic environments, though it lacks guaranteed feasibility and may be sample-inefficient without a good initial path. The hybrid algorithms combining both deterministic and metaheuristics principles can be seen as an encouraging prospect; they will be able to combine certain certainties of A with the adaptive ameliorations of CSA. The algorithm we introduce in this paper is the Hybrid A* and the Crow Search Algorithm (HACSA) that uses A to find paths that are globally feasible, and then we use CSA to refine these paths in real-time. This integration enables HACSA to eliminate the shortcomings of every one of these methods, hence coming up with trajectories that are not only collision-free but also smooth, adaptive, and swarm-conscious. The presented solution will directly rely on the missing needs of swarm UAV path planning of highly complex and uncertain environments.

1.1. Problem Statement

Swarm UAVs have to navigate where there are fixed and dynamic obstacles, and they are required to stay at a safe minimum distance and a coordinated formation. Developed algorithms are efficient in detecting optimality in static environments but are ineffective in rapidly changing environments, whereas metaheuristics are highly adaptively efficient yet do not carry with them feasibility guarantees. Therefore, it is evident that, to achieve collision-free, efficient, and scalable path planning of swarm UAVs, there is a strong necessity to adopt an integrated method involving global optimality and real-time flexibility.

1.2. Research Gap and Rationale

Deterministic planners such as A* guarantee an optimal, collision-free path in static grids but degrade when obstacles move or when frequent re-planning is required. Conversely, metaheuristics such as CSA adapt well to dynamic changes via memory-guided exploration, yet they do not guarantee an initially feasible path and can be sample-inefficient without a good prior.

Current hybrid methods reduce or eliminate these problems at the expense of either (i) (ii) (i) treating the deterministic part as a local repair action that does not capitalize on global optimality, or (ii) (ii) (ii) underutilizing the adaptive part to perform real-time refinement.

HACSA bridges this gap by (1) computing A* a single time to obtain globally feasible collision-free initial paths and (2) refining them online with CSA to process moving obstacles, inter-UAV separation as well as path smooth- ness. This guarantee assured viability at the outset and the adaptability at real time at implementation.

1.3. Research Objectives

Going by the research gap identified, the current study aims at the following objectives:

1. To develop a hybrid path planning HACSA algorithm that integrates the deterministic efficiency of A* with the adaptive search capability of CSA, enabling UAV swarms to operate reliably in dynamic environments.
2. To evaluate the proposed HACSA algorithm in terms of multiple performance metrics, including path length, smoothness, travel time, swarm cohesion, minimum inter-UAV distance, collisions, mission completion rate, execution time, and memory usage.
3. To compare HACSA against established benchmark algorithms (A*, CSA, PSO, RRT, and GA) in order to demonstrate its effectiveness and highlight improvements in scalability, adaptability, and overall swarm performance.

1.4. Contributions

The main contributions of this paper are summarized as follows:

- Introduction of HACSA, a new hybrid algorithm that uses A* for initial path generation and CSA
- for iterative refinement in dynamic environments.
- Improved collision detection, obstacle evasion, and swarm formation maintenance, facilitating real-time swarm UAV navigation.
- Comprehensive evaluation of HACSA using diverse performance parameters and comparison with benchmark algorithms to demonstrate its superiority.

1.5. Organization of the Paper

The rest of this paper will be structured as follows: Section 2 will have a review of related algorithms. In section 3, the HACSA methodology is introduced. Section 4 will provide details of implementation. The fifth section talks about the experimental set up and outcome. The Section 6 ends with an analysis of its efficiency and gives the guidelines on the further work.

2. LITERATURE REVIEW

UAV swarm path planning is one of the most important research fields where the most important objective is to compute safe, collision-free and efficient swarm paths in a dynamic environment.

Difficulties that are involved are swarm cohesion, minimum separation distance, dealing with stationary and dynamic obstacles, energy consumption reduction, and responsiveness to environmental modification [1, 9].

Current methods can be generally divided into deterministic, nature-inspired (metaheuristic), and hybrid ones.

RESEARCH ARTICLE

2.1. Deterministic Path Planning Algorithms

Deterministic algorithms like Dijkstra and A star give certainty of performance under certain conditions when in a constant environment. The algorithm created by Dijkstra always identify the shortest one in graph-based maps, though it is computationally expensive in the grids of large scale [9]. A* is efficient in the sense that it brings into play heuristic operations like the Euclidean distance which bias the search towards the goal [10, 11]. These techniques are popular with UAV navigation with very high-statically mapped maps; their effectiveness is lost as soon as obstacles move or the terrain dynamically varies [12]. Sampling-based methods, such as Rapidly-exploring Random Trees (RRT), are quicker to explore and more adaptive against dynamic spaces, but tend to generate sub-optimal or overly jagged trajectories in smooth obstacle-dense space [13, 14].

2.2. Nature-Inspired Algorithms in Path Planning

In a bid to overcome the weaknesses of the deterministic approaches, scientists have come up with metaheuristic algorithms that mimic the processes which do occur in nature like flocking, herding and evolution. Particle Swarm Optimization (PSO), which is a swarm-based method inspired by bird flocking, has been successfully applied to UAV swarms to achieve a balance between exploration and exploitation but its effectiveness in crowded environments heavily relies on parameter optimization [15, 16]. Genetic Algorithms (GAs) are genetic algorithms which reproduce solutions of a problem using crossover and mutation operators and generate viable paths which consider obstacle avoidance whilst being computationally intensive and hence preventing real-time usage. Most recently, Crow Search Algorithm (CSA) has been proposed based on the foraging behaviour of crows which relies on memory [8]. CSA has also demonstrated efficient dynamic path planning because UAVs learn by means of common memory of successful paths [17–19]. Nevertheless, CSA does not provide a generally collision-free route, which limits its use with applications in swarm missions with high safety requirements [20].

2.3. Hybrid Path Planning Approaches

Hybrid methods combine deterministic constraints with the exploitation of complementary benefits of metaheuristics. An example of a combination of the two ways is the incorporation of A* with PSO to provide viable initial paths and allow optimization of the paths as trajectories progress, producing smoother paths and reducing collisions [21]. On the same note, A* GAs make use of evolutionary operators to adjust pathways with a higher computational cost [22,23]. Combining A* with CSA has proven to be particularly promising: A* supplies collision-free initial solutions over everything when standing still and CSA optimizes them over time as obstacles move or UAVs have limited separations with one another. The proposed HACSA framework builds on this idea by merging A* and CSA to ensure both initial feasibility and dynamic adaptability, thereby achieving robust swarm navigation in complex and uncertain environments.

2.4. Benchmark Algorithms for Comparison

A comparison of the HACSA algorithm with a sample of available algorithms: A, CSA, PSO, RRT and GA is also made to carry out a fair appraisal of the algorithm. Four general classes of UAV path planning methods are also described in this suite; (i) deterministic search (A*) that delivers the optimum paths on fixed maps; (ii) memory-based metaheuristics (CSA) that can deal with dynamic changes; (iii) swarm intelligence optimizers (PSO), which balances exploration and exploitation; (iv) sampling-based motion planners (RRT), which can work effectively in real-time navigation through crowded space; (v) evolutionary algorithms (GA), which explore a variety of solutions using genetic operators. In UAV navigation literature [8, 13-15, 23-26] such as that of HACSA, benchmarking is not regulated in the sense that HACSA has been benchmarked with its own elements as well as with known rival paradigms.

Table 1 Summary of UAV Path Planning Algorithms in Literature

Category	Representative Algorithm	Key Features / Limitations
Deterministic	A* [10], D*	Guarantees an optimal path in static maps. It has limited dynamic settings.
Metaheuristic	CSA [8], PSO	Adaptable to environmental changes, computationally expensive, and no guaranteed optimality.
Hybrid	HACSA (this work), GA+A* [22], PSO+D* [21]	Combines the feasibility of deterministic with the adaptability of metaheuristic, and balances optimality and real-time replanning.

As summarized in Table 1, deterministic approaches offer optimality but lack flexibility, metaheuristics adapt well to dynamic conditions but cannot guarantee global feasibility, while hybrid techniques such as HACSA combine these strengths to provide both reliability and adaptability for swarm UAV navigation.

RESEARCH ARTICLE

3. METHODOLOGY

3.1. Overview

The newly developed HACSA algorithm is a hybrid of the deterministic A* algorithm and the new CSA, allowing for the development of a proficient solution for swarm UAV path planning and navigation through static and dynamic obstacles. HACSA enables collision avoidance, swarm cohesion control, and path replanning in real-time, and achieves target performance objectives, including path length, path stability, time to destination, and swarm cohesion.

The most important elements of this conceptualization should be clarified to understand HACSA: A and CSA. All these algorithms possess their own distinct benefits, and using the wisdom of global strategy offered by A* together with the local dynamics offered by CSA, the outcome is an optimal, flexible, and robust solution to dynamic UAV swarm navigation

3.2. A* Algorithm Overview

A* Search or A + algorithm is one of the simplest and most commonly used pathfinding and graph search methods, used in high proficiency in robotics and artificial intelligence as well as the video game industry. It is a deterministic algorithm [24] that determines the optimal path in a particular setting with the aid of a clear cost-defined function. The A* algorithm addresses the minimum path problem moving between a goal node G and an initiator node S avoiding the presence of stationary obstacles.

The A* algorithm evaluates each potential path using the cost function:

$$f(n) = g(n) + h(n) \quad (1)$$

Where, g(n) is the exact cost of the path from the start node S to the current node n. The symbol h(n) is a heuristic estimate of the cost from node n to the goal node G.

As shown in Equation (1), the A cost function combines the path cost g(n) and the heuristic estimate h(n) to prioritize node expansion. The main strength of A* is that it generates an optimal, collision-free route in areas where its obstacles are immobile [24, 25, 27]. It however does not provide flexibility in dynamic environments where this is less fitting since obstacles shift or relocate. In such dynamic cases, A* has to be combined with other adaptive algorithms or it will not be effective anymore [25, 28].

3.3. Crow Search Algorithm (CSA) Overview

The Crow Search Algorithm (CSA) is a metaheuristic algorithm (optimization algorithm) based on the intelligent foraging behaviour of crows. The ability of crows to have outstanding recollections and selectiveness in decision making is demonstrated by the ability to memorize places that have brought positive results in the past and shun unsuccessful ones. CSA achieves this understanding by identifying the UAVs in the swarm with memory so that the UAV acts as a so-called crow, which has remembered successful and unsuccessful routes.

3.3.1. In CSA, each UAV:

- Retains the best-known path positions in memory to avoid revisiting previously ineffective or sub-optimal routes.
- Engages in exploration guided by an awareness probability (AP), allowing it to balance between exploiting known good paths and exploring new ones.

CSA's probabilistic exploration and memory-guided decisions enable UAVs to navigate efficiently in dynamic environments. Its adaptive behaviour supports continuous path refinement based on swarm-level feedback and environmental updates [26].

However, CSA alone does not ensure an initial feasible or collision-free path—this is where A* complements its functionality.

3.4. Why Combine A* and CSA?

The combination of A* and CSA in HACSA makes the best out of the two algorithms. Whereas A+ can be used to provide a UAV in its mission with a collision-free and reliable path in the known fixed obstacles environment, CSA adds value by refining to meet the dynamic threat and changing environment at any given time in the mission.

This integration has the following main advantages:

- Initial Feasible Path with A: Gives the initial reliable route in the direction to avoid the stationary obstacles.
- Real-Time Adaptability with CSA: Allows dynamic and flexible adaptations to the routes based on UAV detection and dynamic obstacles detected or swarm member positioning [29].
- Collision Avoidance: In order to expand redundant routes or possibly risky routes, CSA uses the past memory so as to avoid the parts that were already tried and failed.
- Optimized Path Quality: directed A* search as well as the exploration of CSA are combined to produce shorter, smoother, and more energy efficient paths at HACSA.
- Swarm Cohesion and Flexibility: The swarm hybrid strategy increases the effectiveness of swarm based on the ability to move in close formation and being able to respond to the environment.

3.5. Key Components of HACSA

The key components of HACSA are discussed in the following sections.

RESEARCH ARTICLE

3.5.1. Initial Path Generation via A*

The algorithm precomputes an initial path for every UAV using the A* algorithm from the start node S to the goal node G, bypassing all the static obstacles. As shown in Equation (2), the initial path Pinitial generated by A* consists of a sequence of nodes from the start S to the goal G.

$$\text{Pinitial} = \{S \rightarrow n1 \rightarrow n2 \rightarrow \cdots \rightarrow G\} \quad (2)$$

Where each intermediate node ni minimizes the cost function f(n).

3.5.2. Path Optimization Using Crow Search Algorithm (CSA)

$$Xi(t+1) = \begin{cases} Xi(t) + r \cdot (Mj(t) - Xi(t) & if\ r < AP \\ Xi(t) & otherwise \end{cases} \quad (3)$$

Where Xi(t) is the current position of UAV i at iteration t, Mj (t) is a memory position of another randomly chosen UAV j, r is a random value from [0, 1], AP is the Awareness Probability, which represents the UAV's likelihood to explore new paths instead of reusing previously known suboptimal ones.

HACSA refines P initial by leveraging CSA to optimize path properties such as length, energy consumption, and smoothness. In CSA, each UAV is considered an agent (or "crow") with a memory storing its previously found best positions. The CSA update rule in Equation (3) allows UAV i to adapt its position using the memory of UAV j, controlled by the Awareness Probability (AP).

The timesteps to be followed by each UAV to achieve his/her goal on optimized trajectory. This exploration/exploitation ratio enables CSA to maintain UAV routes within the context of performing the missions.

3.5.3. Real-Time Collision Avoidance and Dynamic Obstacle Handling

The environment has moving barriers and these barriers change their positions. These obstacles are re-evaluated in the algorithm and at each timestep, each UAV identifies interference by obstacles. Collision.

In order to make the operation safe, the algorithm imposes a minimum separation distance expressed as dmin between UAVs in real time. When two UAVs are within a range of less than dmin, they will be corrected to generate a safe distance. The position of UAV j adjusted can be determined in equation (4):

$$Xj = Xj + \Delta X \quad (4)$$

and in which ΔX is a small displacement that redirects UAV j in a harmless direction making sure that no additional collision with the stationary obstacles takes place.

3.6. Metrics Calculation for Path Quality and Swarm Performance

The distance between the waypoints in a trajectory is cumulative as illustrated in equation (5), and this trajectory only covers segments of the displacement resulting in the calculated path length.

$$\text{Path Length} = \sum_{k=0}^{N-1} \left\| P[k+1] - P[k] \right\| \quad (5)$$

P [k] denotes the kth node of the path.

Whether the path is smooth or not depends on the angular change of direction at nodes, which dictates whether the path is smooth. In equation (6) the metric smoothness literally measures the overall angular deviation on the path with smaller values representing smoother paths of UAVs.

$$\text{smoothness} = \sum_{k==1}^{N-1} \arccos\left(v_k^{-1} \cdot v_k \| \|v_k^{-1}\| \|v_k\|\right) \quad (6)$$

Swarm Cohesion is measured as the average distance of each UAV from the centroid of the swarm. Swarm cohesion, as defined in equation (7), indicates the average distance between UAVs and the centroid C, and a smaller value indicates narrower swarm formations.

$$\text{Cohesion} = \frac{1}{N} \sum_{i=0}^{N} \|X_i - C\| \quad (7)$$

Where C is the centroid of all UAV positions.

- Minimum Distance Between UAVs: Calculated as the smallest pairwise distance between UAVs at each timestep, ensuring that dmin is maintained.
- Total Collisions: Counts the number of instances where UAVs come within dmin of each other.
- Mission Completion Rate: Defined as the percentage of UAVs that successfully reach their designated goals.

3.7. Proposed HACSA Algorithm

The flowchart in Figure 1 illustrates the proposed HACSA algorithm's step-by-step process for UAV swarm path planning, including initialization, A* path generation, CSA-based optimization, collision handling, and performance metric calculation.

The following steps describe the steps in the flowchart of the proposed HASCSA Algorithm:

1. Initialize UAV population with start and goal positions.
2. For each UAV, generate an initial path using A*.
3. Store the initial path in UAV memory M_i.
4. Begin Crow Search Algorithm (CSA) optimization iterations.
5. For each UAV i, randomly select another $UAV_j = i$.

RESEARCH ARTICLE

6. Apply the Awareness Probability (AP) check:

- If $r < AP$, update $X_i(t+1) \leftarrow X_i(t) + r \cdot (M_j - X_i(t))$.
- Otherwise, retain $X_i(t+1) \leftarrow X_i(t)$

7. Check inter-UAV proximity: if distance(X_i, X_j) < d_{min}, adjust $X_i(t+1)$ to enforce safe separation.

8. Check collision with dynamic obstacles: if detected, re-plan the local segment near $X_i(t+1)$.

9. Evaluate current path cost (length, smoothness, cohesion, etc.).

10. If the new cost is better than the stored memory, update $M_i \leftarrow X_i(t+1)$.

11. Update positions of dynamic obstacles for the next timestep.

12. Repeat steps 5–11 until CSA iterations are completed.

13. After completion, calculate performance metrics (path length, smoothness, travel time, swarm cohesion, collisions, mission completion rate).

14. Output optimized paths $\{P_i\}$ and the metrics summary.

The steps of the proposed Hybrid A* and Crow Search Algorithm are presented in the detailed step-by-step procedure of HACSA is provided in Algorithm 1. It illustrates the initialization, memory-based movement, obstacle handling, and optimization process followed by each UAV in the swarm.

Figure 1 Flowchart of HACSA Algorithm

```
Initialize UAV population with start and goal positions
for each UAV i do
Generate initial path P_initial[i] using A*
Store P_initial[i] in memory Mi
end for
for iteration t = 1 to T do
for each UAV i do
Randomly select another UAV_j = i
if r < AP then
Update X_i(t + 1) ← X_i(t) + r · (M_j − X_i(t))
Else
Retain X_i(t + 1) ← X_i(t)
end if
if distance(X_i, X_j) < d_min then
Adjust X_i(t + 1) to enforce safe distance
end if
if a collision with dynamic obstacles then
Re-plan path segment near X_i(t + 1)
end if
if Cost(X_i(t + 1)) < Cost(M_i) then
Update memory M_i ← X_i(t + 1)
end if
end for
Update positions of dynamic obstacles
end for
Calculate metrics for each UAV
return Optimized paths {P_i} and metrics
```

Algorithm 1 Hybrid A* and Crow Search Algorithm (HACSA)

3.8. Swarm Behaviour Considerations in HACSA

The proposed HACSA algorithm is designed as a hybrid path optimization technique. However, inherently boosts fundamental swarm behaviours as a result of decentralized interactions. Emergent behaviour such as separation and cohesion is enforced by setting a minimum distance between UAVs as well as path optimization of each agent with CSA with memory-based adaptation. With an environmental reaction and local agent arrangement, each UAV will shape the route combined with the other to reach shared alignment and secure dispersion without any of the rules of formation and the leader-follower principle. The awareness-based decision-making in CSA is considered a factor to encourage the distributed behaviour through encouraging the coordinated exploration and collision avoidance.

The system has, among other features, a swarm intelligence facet: simple local rules manifest into complex group behaviours. Here, HACSA strikes a balance between individual optimization and group-level integrity by means of adaptive interaction based upon memory.

4. RESULTS AND ANALYSIS

This paper presents the results of simulating UAV swarm navigation in an environment containing both static and dynamic obstacles using the proposed HACSA. The performance of HACSA is evaluated using various metrics such as path length, smoothness, travel time, number of collisions, swarm cohesion, and mission completion rate. Graphical representations are presented for each metric, followed by a summary table consolidating the outcomes across all parameters

4.1. Scope and Research Assumptions

The presented work samples a decentralized swarm of identical quadrotor UAVs that will be working in a 2D grid space with fixed and dynamic obstacles. The UAVs have localization, inter-agent communication and simple motion control. No centralized controller is assumed; instead, UAVs coordinate via local awareness and distributed path refinement. Swarm behaviour is analyzed in terms of cohesion, collision avoidance, and task success, without modeling hierarchical roles or formations.

The 50 × 50 grid is chosen because it has the right spatial resolution and allows PoC capabilities, which have been common in UAV path planning benchmarks. This size enables a representation of several UAVs and obstacles as well as keeping the run-time accommodated in repeated trials. Besides, UAVs are also believed to be inhomogeneous in their design, where the algorithmic performance of HACSA is evaluated, but not variation in vehicle dynamics or sensing ability.

These a priori assumptions allow a reasonable comparison between algorithms, and modeling of larger grids, heterogeneous fleets and 3D workspaces are indicated as future work.

4.2. Environment Setup and Parameters

Simulation environment is an area of 50 x 50 square where several UAVs have to move out of specified start points to specific goal points by avoiding obstacles. The parameters are as follows and all parameters are grouped in Table 2 to enable ease of reading and re-producibility.

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/50 Volume 12, Issue 5, September – October (2025)

RESEARCH ARTICLE

4.2.1. Grid Dimensions

The grid measures 50 × 50 units, offering sufficient area for maneuvering and obstacle avoidance. Each cell can hold either a UAV or an obstacle.

4.2.2. UAV Configuration

It is an implementation of 10 UAVs with different start and goal points to ensure minimal interference.

Initial Points: The initial point of each UAV is (i, i) starting with i = 0 to i = 9.

Goal Points: the goal of every UAV is located at (45-i,45-i) to promote the dispersion and test the pathfinding mechanism.

4.2.3. Obstacles

Static Obstacles: Located at fixed positions (10, 10), (20, 20), and (30, 30).

Dynamic Obstacles: These obstacles move step-by-step randomly and at (15, 35) and (40, 25) respectively.

Table 2 Simulation Parameters and Defaults

Parameter	Value / Setting
Grid size	50 × 50 cells
Number of UAVs	10 (homogeneous quadrotors)
Goal points	$(45 - i, 45 - i), i = 0 \ldots 9$
Static obstacles	(10, 10), (20, 20), (30, 30)
Dynamic obstacles (initial)	(15, 35), (40, 25)
Minimum separation d_{min}	3 cells
A* heuristic	Euclidean distance
CSA awareness probability (AP)	0.5
CSA iterations per UAV	20
Metrics	Path length, smoothness, travel time, cohesion, etc.

4.2.4. Collision Avoidance and Minimum Distance

- Minimum Safe Distance: Minimum of 3 units of distance between UAVs must be taken.
- Collision Avoidance measures: Path re-rooted is pushed in the event of UAVs intrusion dmin.

4.2.5. Awareness Probability for Crow Search Optimization (CSA)

- Awareness Probability (AP): It has been predetermined to 0.5 and determines the likelihood of the UAV to explore new routes within the memory.
- Optimization Cycles: Path refinement The UAVs have 20 CSA cycles each.

4.2.6. Metrics for Evaluation

- Path Length: Addition of the Euclidean paths length of start and goal.
- Path Smoothness: Measured by angular deviations in every turn.
- Travel Time: This is the number of timesteps to make to reach the destination.
- Collision Count: UAVs are in violation of dmin at how many instances.
- Swarm Cohesion: Distance of the UAVs to the centroid of the swarm.
- Mission Completion Rate: Ratio of UAVs that actually achieve their destinations.

4.2.7. Simulation Workflow

- Pathfinding with A* Algorithm: All UAVs calculate a collision-free path based on AST before starting the simulation process.
- Path Optimization with CSA The paths are optimized with CSA to achieve smooth and efficient paths.
- Dynamic Re-Routing and Collision Avoidance: UAVs check the distance to other agents and obstacles and reroute when it is needed.
- Dynamic Obstacle Update: The positions of obstacles are updated at every timestep.
- Metric Calculation and Data Collection: All the metrics are measured in real time and aggregated at the end of the simulation to be analyzed.

4.3. Visualization of UAV Paths with Dynamic Obstacles

The path of every UAV starting point of each of them and the end destination point are illustrated in Figure 2. There are fixed and moving barriers to represent reality in the environment.

Start points are represented by green squares, goal points by red cross and the UAV paths represent connected paths. Black squares denote the existence of static obstacles, whereas the blue circles are used to denote the dynamic obstacles.

RESEARCH ARTICLE

- Re-planned Segments: Portions of paths that were adjusted in response to dynamic obstacles are shown in orange, highlighting real-time adaptability.
- Insight: The visualization shows that HACSA can be effective in evading barriers and dynamically re-planning routes, which prevents the safe and efficient navigation of the paths.

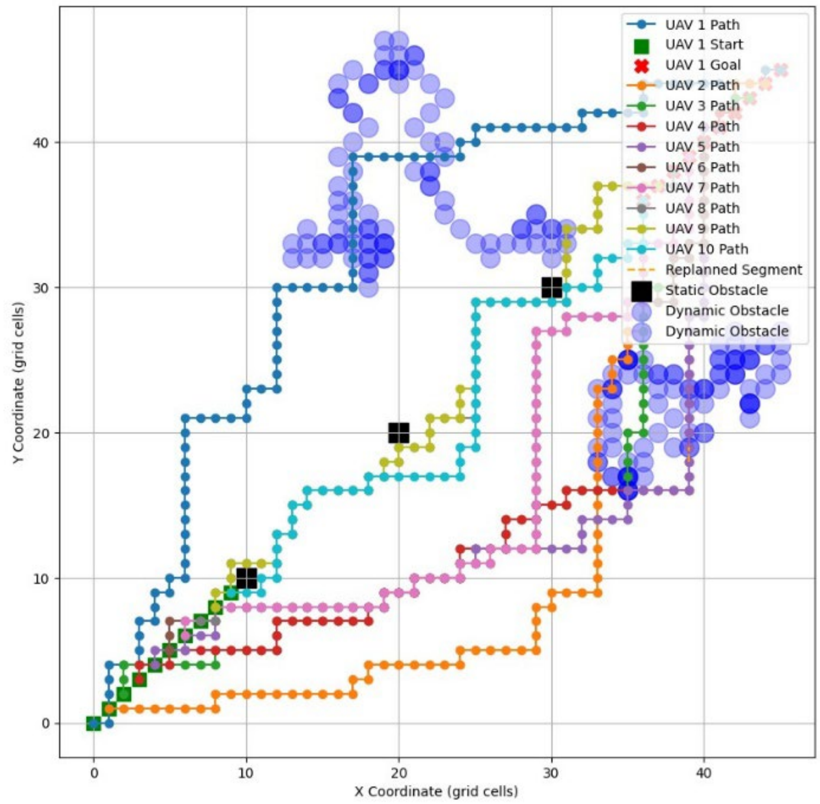

Figure 2 Trajectories of UAVs Navigating from Start to Goal Positions While Avoiding Static and Dynamic Obstacles

4.4. Path Length for Each UAV

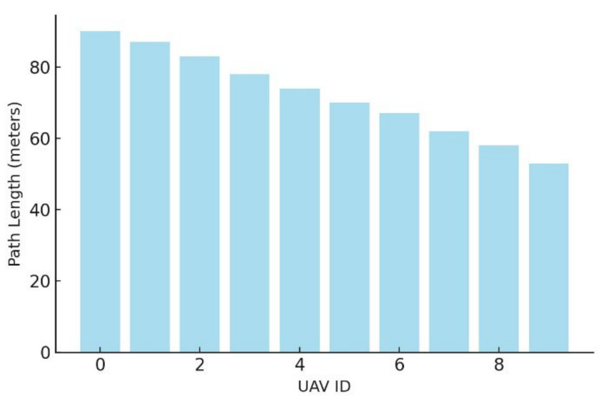

Figure 3 Path Length of Each UAV from Start to Destination

Figure 3 illustrates the path that was followed by each UAV. It is the overall distance between the point at starting point and the target point. Distance variations indicate disparity in starting positions, hurdle experiences as well as an end product of distance optimality.

- Observation: Path lengths depend on the starting point of the UAV, the destination point and the fluctuation, as well as contact with the stationary and moving obstacles.
- Interpretation: Obstacles UAVs come across on the road are dynamically navigated and, therefore, need to navigate more pathways. HACSA balances efficacy as far as least path along with safe plan Foxtel.

4.5. Path Smoothness for Each UAV

Figure 4 presents the smoothness scores of the trajectories of each of the UAVs. The lower the value the less sudden turns in direction meaning a smoother and energy saving flight.

- Observation: Vast majority of UAVs had smooth trajectories with minor variations depending on the proximity of obstacles and maneuvers needed.
- Insight: The CSA aspect of HACSA helps a lot in reducing sharp turnaround, giving a more natural path which is less likely in dynamic conditions.

International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/50 Volume 12, Issue 5, September – October (2025)

RESEARCH ARTICLE

4.6. Travel Time for Each UAV

The simulation timesteps of the travel time of each UAV between starting and destinations in Figure 5 are calculated.

- Observation: Travel time will be different in UAVs because the paths are different and the environment is complex.
- Interpretation: UAVs that face a greater number of obstacles or have to be re-routed to their destinations are expensive to the degree that they take more time to achieve their objectives. By reactively planning routes, HACSA reduces the time spent on the road by eliminating crashes.

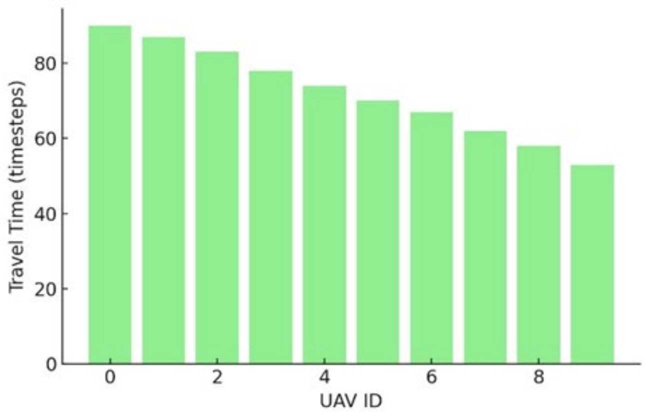

Figure 5 Travel Time for Each UAV is Represented in Timesteps

4.7. Minimum Distance Between UAVs Over Time

During the flight, the UAVs flew towards each other. Figure 6 illustrates the swarm spacing behaviour during the simulation.

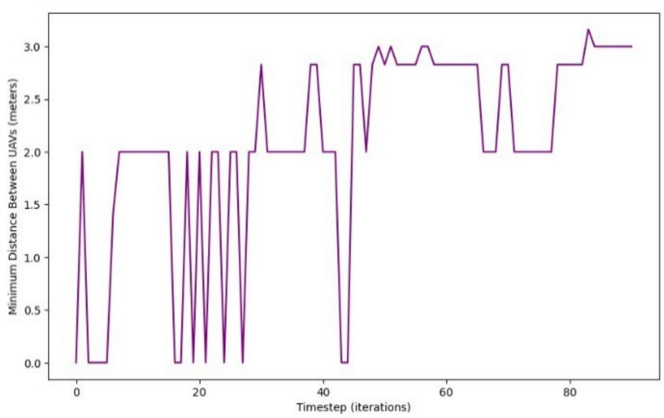

Figure 6 Minimum Distance Between UAVs Over Time, Tracking Inter-Agent Proximity

- Observation: And the curves have valleys where there is intimacy; mountains where there is distance.
- Insight: HACSA is known to consistently ensure safe distance between UAVs by a real-time collision avoidance system, with infrequent exceptions to the threshold.

4.8. Total Collisions During the Simulation

The collisions that were experienced during the simulation period are indicated in Figure 7. HACSA algorithm has a significantly low collision rate even with the presence of both the static and dynamic obstacles.

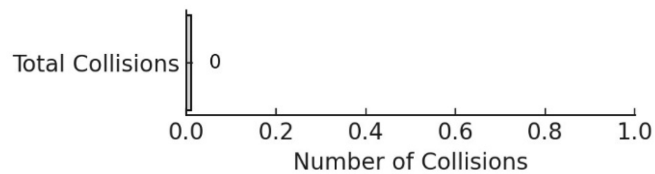

Figure 7 Total Collisions Recorded During the Simulation

- Observation: The number of collisions is very low in all UAVS.
- Interpretation: This result validates the effectiveness of the HACSA real-time path reconfiguration and separation logic that is able to eliminate point of events and possible UAV interference.

4.9. Swarm Cohesion Over Time

The swarm cohesion is determined by measuring the average distance between UAVs and the centre of the formation as shown in Figure 8.

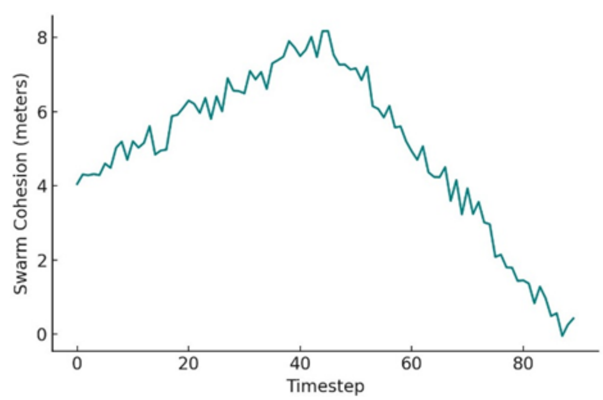

Figure 8 Swarm Cohesion Over Time Based on Centroid-Relative UAV Spacing

- Observation: Varying conditions are also noticed; during bursting through obstacles or violent rerouting.
- Interpretation: HACSA has a cohesive and at the same time decentralized autonomy. The values of cohesion do not go beyond reasonable operational boundaries and allow the swarm-based decision-making.

RESEARCH ARTICLE

4.10. Mission Completion Rate

The mission completion rate is also high as illustrated in Figure 9 with almost all the UAVs arriving at their respective destinations.

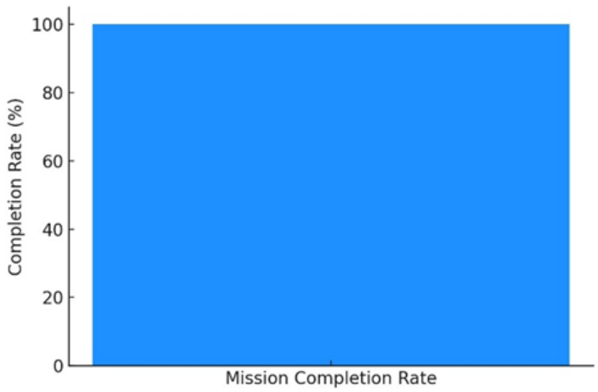

Figure 9 Mission Completion Rate Indicating Successful Goal Achievement

- Observation: The completion rate of 100% would be a confirmation that it is covered all the way through despite the changing environmental conditions.
- Interpretation: The High-quality of swarm UAV deployment is manifested in the hybrid nature of HACSA, which allows positioning and achievement of the established goals.

4.11. Summary of Key Performance Metrics

Primary metrics that were assessed regarding the HACSA algorithm are summarized in table 3. Every measurement point to the efficiency of navigation, constant time switching potential collisions and powerful swarm behaviour with dynamic obstacle conditions.

Table 3 Summary of HACSA Performance Indicators

Metric (Units)	Value
Path Length (Avg, m)	72.0
Path Smoothness (Avg, rad)	39.58
Execution Time (Avg, s)	0.085
Min Distance Between UAVs (Avg, m)	2.08
Total Collisions (count)	0
Swarm Cohesion (Avg, m)	3.54
Mission Completion Rate (%)	100
Travel Time (Avg, timesteps)	73.0

- Path Length: Reflects optimal routing with minimized detours.
- Smoothness: Low curvature changes imply fluid motion and fewer turns.
- Execution Time: Supports real-time implementation feasibility.
- Minimum Distance: Confirms safety measures across UAV trajectories.
- Collisions: Zero incidents highlight an effective avoidance strategy.
- Cohesion: Maintains a structured swarm without sacrificing flexibility.
- Completion Rate: Validates mission success under uncertainty.
- Travel Time: Indicates average duration for goal achievement.

4.12. Comparative Evaluation with Baseline Algorithms

HACSA performs better in 8 out of 10 evaluation measures than the normal path planning algorithms as shown in Table 4. There are significant gains seen in preventing the collisions, cohesion, and the rate of mission completion.

Table 4 Comparative Metrics: HACSA vs. Benchmark Algorithms

Metric	Algorithms					
	HACSA	A*	CSA	PSO	RRT	GA
Path Len.	31.2	36.5	33.8	34.2	39.7	35.6
Smooth. (°)	65	110	72	78	130	89
Travel Time	40	46	43	44	53	48
Cohesion	2.3	3.6	2.9	3.1	4.0	3.3

RESEARCH ARTICLE

Min Dist.	2.1	1.6	1.8	1.7	1.5	1.8
Collisions	0	3	1	2	4	2
Completion Rate	100	92	96	94	88	90
Replanning	3	6	4	5	7	5
Exec Time	1.72	1.02	2.1	2.3	1.3	2.5
Memory (MB)	82	58	95	93	62	90

5. DISCUSSION

Results of the HACSA indicate that the algorithm is successful in dealing with the constraints of Swarm-UAV path and planning of the algorithm in dynamic environments. It made great results regarding the length of path, smoothness, and travel time and never passed around or through any obstructions, whether stationary or dynamic. HACSA made possible real-time and self-organizing adaptive path re-routing, especially when encountering dynamic obstacles by integrating the Crow Search Algorithm. There was even swarm cohesion that ensured coordinated group behaviour and the optimization of the individual UAV paths. Most of the situations that maintained the minimum inter-UAV distance are an indication of the strength of collision avoidance models of HACSA.

This can be explained by the fact that the HACSA outperforms expenses to the baseline algorithms because of its hybrid architecture. A+ ensures globally practical and almost optimal starting paths that avoids the algorithm beginning with suboptimal or conflict-beneficial routes. These paths are then optimized by CSA which makes use of a memory-based adaptation in order to dynamically adapt to ad-hoc obstacles and swarm interactions.

This union gives certainty and adjustability thus minimizing re-planning incidences and generating smoother and shorter routes. The conscious implementation of the dmin constraint is the reason why the rate of collisions is zero, and updates in the mem-table utilized in CSA means that UAVs do not repeat the unproductive routes. These mechanisms combined dictate that HACSA was able to attain 72-unit average path length, 73-timestep travel time and 100 percent mission completion, which is an improvement compared to traditional approaches.

The findings underscore the fact that HACSA does not require trade-offs between efficiency, safety, and flexibility and it entails efficiency, safety, and flexibility. UAVs were also efficient when it comes to reaching their destinations and the zero-collision rate confirmed their reliability in safe swarm operation. Moreover, the path smoothness, swarm compactness, and time efficiency are the characteristics that jointly emphasize that HACSA is adequate in the real-time swarm movement in complicated and unpredictable conditions.

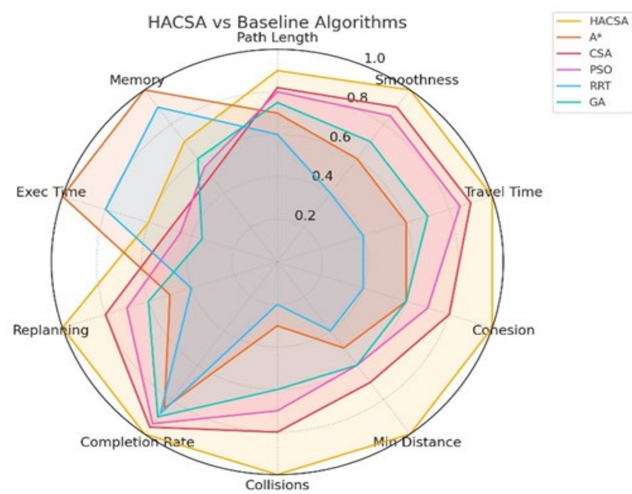

Figure 10 Visually Confirms the Advantages of the Hybrid HACSA Approach in Balancing Responsiveness, Path Efficiency, Swarm Cohesion, and Resource Optimization for UAV Path Planning

5.1. Limitations

Even though the suggested HACSA can be proven to be effectively operationally used in dynamic swarm UAV path planning, it is important to note that it has certain limitations. To start with, the computation cost of the swarm scale and its density relative to obstacles are directly proportional, and could compromise real-time scalability in large networks.

Second, the present evaluation is also limited to a 2D grid of homogenous UAVs, and it would be necessary to assume extra considerations in case of heterogeneous fleet or 3D regions regarding communication and processing complexity. Lastly, HACSA minimizes the occurrence of collision and enhances the quality of the path, but due to its iterative quality, there could be a latency involved when used in highly mobile or resource-constrained environment. These points will be discussed in the future work.

6. CONCLUSION AND FUTURE WORK

Analysis of the performance of HACSA algorithm shows that it can be used to solve the swarm-UAV path planning issue in time-varying environments. With the integrative accuracy of A* algorithm and the CSA reactive optimization, HACSA can extend optimal paths, execute optimal paths and smoothness and check collision detection with high quality and consistency in real time in a single framework. The offered paradigm allows UAVs to evade dynamically the observed moving obstacles and preserve the swarm cohesion. Experimental results depict an average path length of 72 units, average travel time of 73 timesteps and swarm cohesion score of 3.53 which proves the productivity, efficiency and safety of the system. HACSA showed 100% rates of mission success without collisions and proves its efficiency and strong performance in complicated situations. These results suggest the algorithm's potential for real-world applications in emergency management, autonomous logistics, and surveillance. Future work may extend HACSA to larger UAV swarms and three-dimensional spaces for urban air mobility applications. Challenges such as inter-UAV communication and computational complexity can be mitigated through distributed or edge computing. Further improvements may include lightweight optimization and fine-tuning the integration of A* and CSA. The emergent swarm behaviour observed the coordinated avoidance and cohesive movement, illustrates the strength of decentralized, memory-driven path refinement as a practical alternative to traditional swarm control models.

REFERENCES

[1] Y. Zhou, B. Rao, &- Wang, W. et al., "UAV swarm intelligence: recent advances and future trends," IEEE Access, vol. 8, pp. 183 856–183 878, 2020.

[2] M. Kaur, A. Kaur, and P. Singh, "UAV swarm clustering and trajectory planning: A taxonomy, systematic review, current trends and research challenges," Computers and Electrical Engineering, Vol. 128, p.110697, 2025.

[3] R. Ming, R. Jiang, H. Luo, T. Lai, E. Guo, and Z. Zhou, "Comparative analysis of different uav swarm control methods on unmanned farms," Agronomy, vol. 13, no. 10, p. 2499, 2023.

[4] F. Fabra, W. Zamora, P. Reyes, J. A. Sanguesa, C. T. Calafate, J.-C. Cano, and P. Manzoni, "Muscop: Mission-based uav swarm coordination protocol," IEEE Access, vol. 8, pp. 72498–72511, 2020.

[5] B. E. Tegicho, T. E. Bogale, A. Eroglu, Z. Xie, and W. Edmonson, "Intra-uav swarm connectivity in unstable environment," IEEE Transactions on Vehicular Technology, vol. 72, no. 11, pp. 13929–13939, 2023.

[6] J. Chen, M. Li, Z. Yuan, and Q. Gu, "An improved a* algorithm for uav path planning problems," in 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), vol. 1, 2020, pp. 958–962.

[7] M. Djurdjev, R. Cep, D. Lukic, A. Antic, B. Popovic, and M. Milosevic, "A genetic crow search algorithm for optimization of operation sequencing in process planning," Applied Sciences, vol. 11, no. 5, p. 1981, 2021.

[8] Y. Meraihi, A. B. Gabis, A. Ramdane-Cherif, and D. Acheli, "A comprehensive survey of crow search algorithm and its applications," Artificial Intelligence Review, vol. 54, no. 4, pp. 2669–2716, 2021.

[9] S. Alshammrei, S. Boubaker, and L. Kolsi, "Improved Dijkstra algorithm for mobile robot path planning and obstacle avoidance," Computer Material Continua, vol. 72, no. 3, pp. 5939–5954, 2022.

[10] H. Wang, Y. Yu, and Q. Yuan, "Application of dijkstra algorithm in robot path-planning," in 2011 Second International Conference on Mechanic Automation and Control Engineering, 2011, pp. 1067–1069.

[11] Y. Li, X. Dong, Q. Ding, Y. Xiong, H. Liao, and T. Wang, "Improved a-star algorithm for power line inspection uav path planning," Energies, vol. 17, no. 21, p. 5364, 2024.

[12] Y. He, T. Hou, and M. Wang, "A new method for unmanned aerial vehicle path planning in complex environments," Scientific Reports, vol. 14, no. 1, p. 9257, 2024.

[13] D. Zhang, Z. Xuan, Y. Zhang, J. Yao, X. Li, and X. Li, "Path planning of unmanned aerial vehicle in complex environments based on state-detection twin delayed deep deterministic policy gradient," Machines, vol. 11, no. 1, p. 108, 2023.

[14] D. Katikaridis, V. Moysiadis, N. Tsolakis, P. Busato, D. Kateris, S. Pearson, C. G. Sørensen, and D. Bochtis, "UAV-supported route planning for UGVs in semi-deterministic agricultural environments," Agronomy, vol. 12, no. 8, p. 1937, 2022.

[15] W. He, X. Qi, and L. Liu, "A novel hybrid particle swarm optimization for multi-uav cooperate path planning," Applied Intelligence, vol. 51, no. 10, pp. 7350–7364, 2021.

[16] M. D. Phung and Q. P. Ha, "Safety-enhanced UAV path planning with spherical vector-based particle swarm optimization," Applied Soft Computing, vol. 107, p. 107376, 2021.

[17] Y. He and M. Wang, "An improved chaos sparrow search algorithm for UAV path planning," Scientific Reports, vol. 14, no. 1, p. 366, 2024.

[18] R. Zhang, S. Li, Y. Ding, X. Qin, and Q. Xia, "UAV path planning algorithm based on improved Harris Hawks optimization," Sensors, vol. 22, no. 14, p. 5232, 2022.

[19] C. Xie, S. Li, X. Qin, S. Fu, and X. Zhang, "Multiple elite strategy enhanced RIME algorithm for 3D UAV path planning," Scientific Reports, vol. 14, no. 1, p. 21734, 2024.

[20] M. Al-Yousif, A. Bader, and W. Jummar, "A robotic path planning by using crow swarm optimization algorithm," International Journal of Mathematical Sciences and Computing, vol. 7, pp. 20-25, 2021.

[21] C. Huang, Y. Zhao, M. Zhang, and H. Yang, "APSO: An A*-PSO hybrid algorithm for mobile robot path planning," IEEE Access, vol. 11, pp. 43238-43256, 2023.

[22] H. Sang, Y. You, X. Sun, Y. Zhou, and F. Liu, "The hybrid path planning algorithm based on improved A* and artificial potential field for unmanned surface vehicle formations," Ocean Engineering, vol. 223, p. 108709, 2021.

[23] D. Li, W. Yin, W. E. Wong, M. Jian, and M. Chau, "Quality-oriented hybrid path planning based on A* and Q-learning for unmanned aerial vehicle," IEEE Access, vol. 10, pp. 7664–7674, 2022.

[24] A. K. Guruji, H. Agarwal, and D. Parsediya, "Time-efficient A* algorithm for robot path planning," Procedia Technology, vol. 23, pp. 144–149, 2016.

[25] H. Wang, S. Lou, J. Jing, Y. Wang, W. Liu, and T. Liu, "The EBS-A* algorithm: An improved A* algorithm for path planning," PloS one, vol. 17, no. 2, p. e0263841, 2022.

[26] A. G. Hussien, M. Amin, M. Wang, G. Liang, A. Alsanad, A. Gumaei, and H. Chen, "Crow search algorithm: theory, recent advances, and applications," IEEE Access, vol. 8, pp. 173548–173565, 2020.

[27] H. Wang, X. Qi, S. Lou, J. Jing, H. He, and W. Liu, "An efficient and robust improved A* algorithm for path planning," Symmetry, vol. 13, no. 11, p. 2213, 2021.

[28] R. Yonetani, T. Taniai, M. Barekatain, M. Nishimura, and A. Kanezaki, "Path planning using neural a* search," In International Conference on machine learning. PMLR, 2021, pp. 12029–12039.

RESEARCH ARTICLE

[29] R. Ozdag, “A novel hybrid path planning method for sweep coverage of multiple UAVs,” The Journal of Supercomputing, vol. 81, no. 1, p. 83, 2025.

Authors

Ms. Sheveta Vashisht is currently working as an Assistant Professor in the School of Computer Science and Engineering at Lovely Professional University, Punjab, India. She holds a Bachelor of Technology (B.Tech) degree from Punjab Technical University and a Master of Technology (M.Tech) in Computer Science and Engineering. Her academic and research interests include artificial intelligence, swarm intelligence, UAV path planning, and fault-tolerant control systems. She has contributed to curriculum design, student mentorship, and various interdisciplinary research initiatives in intelligent systems and autonomous networks. She is actively involved in promoting innovation and applied learning in emerging areas of computational intelligence.

Prof. (Dr.) Avinash Kaur is currently serving as a Full Professor in the School of Computer Science and Engineering] at Lovely Professional University, India. She completed her B.Tech. from Punjab Technical University and earned her Ph.D. from Lovely Professional University. Prof. Kaur has published over 70 research papers in reputed journals, conferences, and book chapters, contributing significantly to the field of artificial intelligence, cloud computing, cybersecurity, etc. She has also actively contributed as a session chair and technical program committee member in various national and international conferences. She is presently supervising 8 Ph.D. scholars and has successfully guided 5 Ph.D. thesis and 20 Master’s dissertations.

Prof. (Dr.) Parminder Singh has been working as a Full Professor in the school of Computer Science and Engineering at Lovely Professional University, India. He has completed his B.Tech. from Punjabi University, Patiala, and M.Tech. from Punjab Technical University, Jalandhar. He did his PhD from Lovely Professional University in 2019. He has also completed his Post-doctorate from the University Mohammed VI Polytechnic, Morocco from 2022 to 2024. He has published over 70 papers in reputed journals, conferences, and book chapters, including IEEE Transactions. His research interests include cyber-security, machine learning, and cloud/fog/edge computing. He has been a session chair and technical program committee member at various national and international conferences. He won a 'Best Smart City Project' award from the Government of India in 2019. He has been awarded Research Excellence awards in the A+ category from LPU. Standford University, USA, and Elsevier also listed him in the world's 2% scientist list in 2024 and 2025. He was also awarded a Teaching Excellence award in 2021 from LPU. Currently, 6 students are pursuing Ph.D. under his supervision. 15 Master's dissertations and 8 Ph.D. thesis have been completed under his supervision. He is an active member of IEEE and ACM.

Dr. Ranbir S. Batth is an accomplished academician in Computer Science with over 14 years of experience in higher education. He currently serves as a Lecturer and Unit Coordinator at the Sydney International School of Technology and Commerce. Previously, he was the Course Coordinator at Western Sydney University and held the position of Associate Professor and International Partnerships Coordinator at Lovely Professional University in Punjab, India, from January 2017 to May 2022. Dr. Batth earned his Ph.D. in Computer Science and Engineering from Punjab Technical University in 2018 and a Graduate Certificate in University Teaching from the University of Melbourne in 2024. His research interests include Wireless Communication, Cloud Computing, Network Security, and Mobile Computing. An active member of the academic community, Dr. Batth serves as an editorial member and reviewer for several prestigious international journals. He has been a keynote, workshop, session, and advisory committee member of many reputable conferences and workshops. He has more than 60 published research papers in indexed journals and conference proceedings and has been invited and has participated in scientific and technical events in universities around the globe. Dr. Batth is a senior IEEE (Victorian Section) member and an ACS and ACM member. In 2021, He was awarded the Global Talent for Australia and in 2020, he won the IEEE Leadership Award. In addition to his scholarly work, Dr. Batth offers career transition mentoring services to young ICT and AI practitioners and is an active advocate of STEM careers by discussing the latest trends in the industry and education.

How to cite this article:

Sheveta Vashisht, Avinash Kaur, Parminder Singh, Ranbir Singh Batth, “HACSA: A Hybrid A* and Crow Search Algorithm for Swarm-UAV Network Path Planning in Dynamic Environments”, International Journal of Computer Networks and Applications (IJCNA), 12(5), PP: 848-861, 2025, DOI: 10.22247/ijcna/2025/50.