IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Explores The Use Of Multi-Objective Particle Swarm Optimization (Pso)

[1] Dr. G. Paavai Anand, [2]G Sakthi Sreenivas, [3]V. Jai Shree

[1] Assistant Professor (S.G), [2] Student, [3] Student,

[1] B.Tech - Computer Science & Engineering with Specialisation in Artificial Intelligence & Machine Learning

Computer Science Department (E-Tech)

SRM Institute of Science and Technology, Vadapalani, Chennai, India.

Abstract: Due to their extreme significance, surveillance and disaster response and environmental monitoring missions using UAVs have emerged. However, because navigation during such an environment characterized by the presence of many conflicting objectives such as minimum flight time, minimum energy consumption and obstacle avoidance is very challenging, the goal of the present work is to propose an MOPSO method that may be used in improving path planning of UAVs under dynamic conditions with the ability to balance multiple objectives. It is observed from simulation results that this methodology supports scalable and real-time UAV operation with better performances in many kinds of environments than typical methods. This work provides a robust strategy to improve the reliability and effectiveness of UAV missions.

***Index Terms* -** UAV, Path Planning, Multi-Objective Optimization, Particle Swarm Optimization (PSO), MOPSO, Dynamic Environments.

I. INTRODUCTION

It focuses on multi-objective particle-swarm optimization which supports efficient UAV path planning considering difficulties in terms of minimal travel time, energy consumption, and obstacle avoidance. Nowadays, UAVs are being put into surveillance, disaster management, delivery services, amongst others whose multi-objective conflicting optimality should be maximized. While the traditional path-planning methods typically fail to take more than one objective onboard most of the times, the multi-objective PSO gives the capability of optimizing a number of goals at a time.

1.1 Background

Unmanned Aerial Vehicles, more commonly known as drones, is a rapidly emerging technology that has captured the attention of almost every diverse sector and numerous applications.

It has been applied in many applications such as:

• Surveillance and Security: UAVs are very frequent tools in military and civilian real-time monitoring, border security, and law enforcement.

•Disaster Response and Search & Rescue: UAVs can provide critical overhead views within the disaster site to assist with searching for survivors, assessing damage, and guiding rescue teams into inaccessible areas.

The usage of UAVs will thus monitor tracking of wildlife, studying deforestation, and monitoring movements of glaciers, health of forests, and water resources in general.

•Agriculture: UAVs facilitate the application of precision agriculture techniques: monitor crop health, adjust irrigation, and apply fertilizers or pesticides to improve productivities with minimum resource wastage.

• Infrastructure Inspection: UAVs conduct an inspection of bridges, power lines, wind turbines, and pipelines with the promise of safe, efficient, and detailed assessments without risking human life.

• Delivery Services: It is a way of using UAVs to ensure delivery of packages, medical supplies, and food as last-mile delivery, which will be the quickest alternative compared to ground delivery.

1.2 Problem Statement

UAV path planning is one of the most complicated tasks due to a complexity of real environments, where several conflicting objectives have to be met simultaneously. Traditional methods essentially tend to optimize one factor, for example to minimize flight time or distance, and this really reflects suboptimal results in reality. However, such missions require that UAVs move through an unpredictable environment with obstacles of varying degrees and find the balance between these conflicting objectives:

- **Minimizing flight time** to ensure mission efficiency,
- **Conserving energy** to extend operational range and endurance,
- **Avoiding obstacles** to ensure safety and mission success.

Besides, the conditions about moving objects or weather really add too much complexity to the problem besides these dynamic changes. Thus, unless competitive factors in the current approach adapt fast enough and optimize, efforts have to be further made towards more robust and adaptive solution.

1.3 Objectives of the Study

The primary objectives of this study are to:

• **Minimum flight time:** Design an algorithm that optimizes UAV path for minimum overall time taken to reach the destination, thereby improving mission efficiency.

• **Energy Efficiency:** It designs an energy-conscious flight path for a UAV that allows endurance use optimization, which in turn saves energy.

• **Obstacle Avoidance:** The UAV should develop inside cluttered environments without any possibility of collision with stationary or moving obstacles and in the least risk sense.

• **Active environment :** This shall imply an adaptive system that reacts in real-time to new conditions appearing in the environment, for example, moving barriers or road debris.

• The MOPSO framework is designed to be optimizing multiple competing objectives to achieve a set of optimal paths toward supporting the flexible mission planning.

II. LITERATURE REVIEW

2.1 Path Planning for UAVs:

The key challenge of UAV path planning is to plan the route as safe and efficient as possible within this complex environment. Traditionally, most approaches for UAV path planning are actually methods targeting the minimisation of single factors like flight time or distance. Some of these are:

1) **Graph-Based Methods:** Dijkstra's and A* are useful general path-planning algorithms that work by discretizing the space into nodes and searching for the shortest path between them. However, such methods are computationally expensive for large, dynamic environments.

2) **Sampling-Based Approaches:** Sampling-based methods, including Rapidly-Exploring Random Tree (RRT), determine feasible paths by exploring the space randomly. Extensions to RRT---such as RRT*---then try to locally optimize those paths in terms of path cost, whilst avoiding obstacles.

3) **Heuristics Methods:** This includes GA and SA. They are both approximating methods that generate solutions by searching the solution space and using random mutation and selections. Methods attempt to globally optimize but can be computationally intensively expensive and tend to converge slowly.

4) **Limitations of Single-Objective Approaches**: Although such single-objective methods are highly effective in optimizing individual goals, such as the minimization of distance or time, the balancing of several objectives, such as the elimination of obstacles, the consumption of energy and time efficiency, cannot be adequately addressed. As a result, suboptimal routes are depicted in real scenarios.

2.2 Particle Swarm Optimization (PSO):

Particle Swarm Optimization (PSO) is one of the popular nature-inspired optimization techniques inspired from swarm behavior in nature, such as bird flocking or fish schooling. In PSO, the swarm of particles is represented as potential solutions and moves through the solution space by updating the positions based on personal best-known positions as well as the best-known positions of their neighbors, which are also known as global best.

PSO Process:

1) Initialization: Randomly initializes a set of particles in the solution space.
2) Evaluation of fitness: Evaluates each particle's position with a help of a fitness function which measures how well the solution meets the objectives.
3) Update velocity and position: Updates the velocity and position of each particle based on its best-known solution in the neighborhood; this together with the global best-known solution in the search space up to that time step.
4) Convergence: Over the iterations, the particles, ideally, converge toward optimal solutions and ideally, toward the global optimum.
5) Relevance of PSO to Optimization Problems: PSO is useful specifically for complex and non-linear problems in optimization, where it is a high-dimensional search space, and traditional methods of optimizations cannot be used. Being widely applied in machine learning, robotics, and path planning, its simplicity and nature of avoiding local optima are the reasons why it is so popular.

2.3 Multi-Objective PSO (MOPSO)

Multi-Objective Particle Swarm Optimization (MOPSO) extends the traditional PSO by optimizing multiple conflicting objectives simultaneously. Instead of optimizing a single fitness function, MOPSO generates a set of optimal solutions known as the **Pareto front**, which represents the best trade-offs between objectives.

- **How MOPSO Works:**
  1) **Objective Functions:** In MOPSO, multiple objective functions (e.g., minimizing flight time, minimizing energy consumption, avoiding obstacles) are defined, and each particle evaluates its position based on all these objectives.
  2) **Pareto Front Management:** As the swarm evolves, non-dominated solutions (solutions that are not worse than others in all objectives) are stored in a repository that forms the Pareto front.

  3) **Global Best Selection:** Instead of a single global best, the global best of MOPSO is chosen from the Pareto front. The ability to consider an array of solutions that the algorithm may explore is enabled.

  4) **Diversity Maintenance:** Diversity Preservation Crowding distance and mutation are used for preserving diversity in order to avoid premature convergence of the swarm and to make the Pareto front diverse.
- **Advantages of MOPSO:**
  1) Combines several objectives in real time; thus suitable for complex, dynamic environments.
  2) It has a product which would allow the decision-maker to pick the best compromise in meeting specific mission requirements.
  3) More scalable and flexible than Single-Objective PSO.

2.4 Related Works

Optimization techniques in UAV path planning have emerged rapidly within the recent years, particularly adaptive PSO and hybrid approaches.

1) **Adaptive PSO for UAV path planning** in dynamic environments (Li et al., 2019). It will introduce a new algorithm proposed here, an adaptive PSO that dynamically learns and adjusts its parameters: the inertia weight and velocity, being aware of closeness to obstacles and changes in the environment. Simulations prove the adaptive PSO converges faster and avoids more obstacles.
2) **Hybrid PSO and Genetic Algorithm for UAV Path Optimization**, Lee et al., 2020 In this paper, the hybrid PSO-GA combination has been published. PSO was utilized for proper exploration of the solution space and GA was used to implement crossover and mutation operations in order to preserve

diversity and prevent local optima. The proposed hybrid scheme has efficiently produced more robust UAV path optimization in complex environments where obstacles were present.

3) **MOPSO for UAV path planning** of Smith et al. (2020): In that particular paper, the authors used MOPSO to achieve optimization parallel objectives like energy efficiency, flying time, and obstacles. It has been demonstrated that MOPSO can guide UAVs over complex terrain and accomplish multiple objectives much better than single-objective methods.

2.5 System Architecture

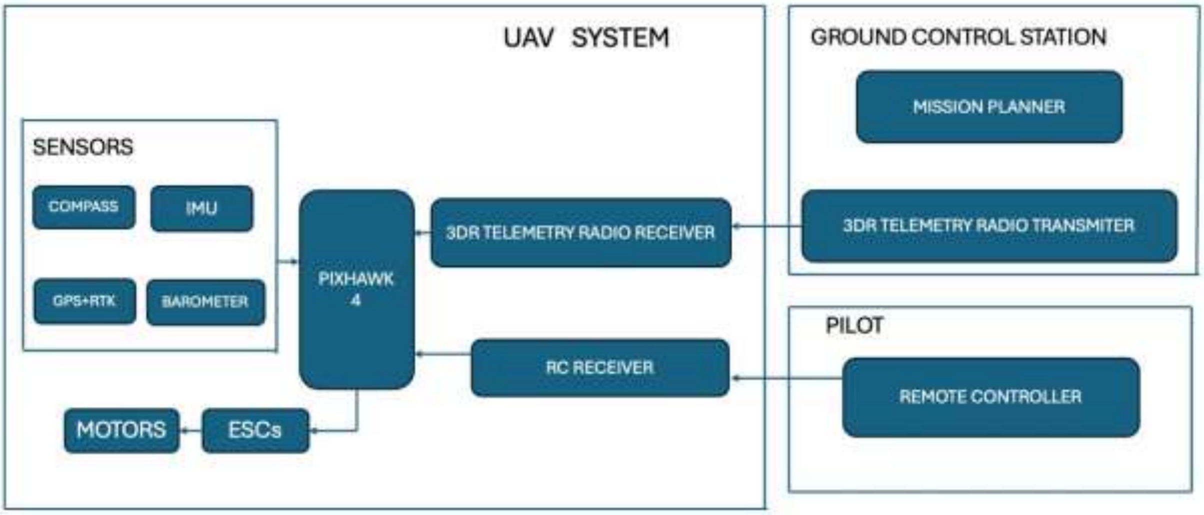

Fig 1: Architecture of UAV System

III. METHODOLOGY

3.1 Problem Formulation:

The UAV path planning problem can be posed as a Mult objective optimization. The various objectives are travel time minimization, reduction of energy consumption, and avoidance of obstacles. All the foregoing could be represented in one fitness function that quantifies the performance of each UAV path. Each objective is weighed in light of its importance in the mission, and constraints like max flight range, altitude restrictions, or environmental hazards are also incorporated. The problem will be Mult objective formulation, ensuring that multiple critical factors are analyzed in a single run where the UAV balances sometimes conflicting requirements during the mission. The challenging issue now would be to find a set of paths to optimize the conflicting objectives with real-world operational constraints, such as fuel limits or time windows. This way, the proposed system will frame the problem with search for optimal or near-optimal solutions that will ensure UAV safety and efficiency.

3.2 Swarm Initialization:

In order to initialize the swarm, a population of particles is created where every particle carries a candidate UAV path. These are widely spread across the search space that consists of all possible routes from the starting point up to the destination. Each particle's position and its velocity are initiated randomly over certain bounds. The diversity of the swarm ensures that it covers a huge portion of the search space; it minimizes the chances of getting trapped in local optima and maximizes its chances of finding the global best solution. The number of particles in the swarm depends on the complexity of the problem and the available computation time. In each iteration, the position of every particle will be corrected with repective. its performance relative to the objectives. Initialization of the swarm is a very important stage for laying down the PSO algorithm by providing a wide, varied set of potential solutions. This would ensure exploration of the path planning space thoroughly.

3.3 Fitness Evaluation:

The path of every particle or the performance of the UAV will be determined by calculating a fitness score based on the multi-objective fitness function. It considers travel time, energy efficiency, and obstacle avoidance by giving every path a score based on how well each of the objectives is met. The paths with fewer travel times, lower energy expenditures, and traverse safer with less barriers will result in a better score. Penalties are assigned to paths that fail the restrictions such as closing in on a barrier or consuming more energy than allocated. This ranking ranks the particles and helps guide how the swarm moves forward. Continuing to constantly rank the fitness of each particle ensures the algorithm constantly rewards only the best solutions to continue. The process of fitness evaluation plays a very critical driving role to guide the PSO's exploration and exploitation strategies towards guiding the particles toward better paths.

3.4 Velocity and Position Updates:

Velocity and Position Updates: A particle updates its velocity and position considering both personal experience and collective knowledge within the swarm. Velocity is calculated using three factors: inertia, a cognitive component known as personal best position, and social component-known as global best position. This changes the way a particle moves in the solution space: it is no longer a trade-off between exploitation of already good paths and exploration of unvisited regions. Once the velocity is updated, it again recalculates the position of the particle to push it closer to an optimal solution. These updates are performed iteratively, and particles will proceed along a gradually refined path. This is important because it nudges the swarm toward convergence with particles changing their direction slowly toward better solutions, which means that time taken to travel is saved; energy efficiency and better avoidance of obstacles occur. The balance between exploration and exploitation balances against getting the algorithm stuck in local optima.

3.5 Pareto Optimization:

Since the problem involved has multiple conflicting objectives, the PSO algorithm produces a Pareto front of non-dominated solutions. A solution is nondominated if no other solution improves one objective without worsening another. For the problem of UAV path planning, the Pareto front presents tradeoff solutions between objectives like minimizing energy consumption in comparison with reducing travel time. The goal would be to provide a set of optimal paths where each solution would have achieved a different balance between the objectives. The path to be chosen shall then be a decision, taken on the basis of that priority aspect that is being chosen by the decision-makers themselves. Be it velocity, safety or energy efficiency. It shall not indulge in any bias towards one solution alone but shall explore a very wide variety of optimal paths under Pareto optimality. Hence, because of maintaining a set of different solutions, the Pareto front ensures not only flexibility in UAV operations but also various mission requirements.

3.6 Obstacle Avoidance Mechanism:

A mechanism of obstacle avoidance is incorporated in the formulation of PSO so as to avoid unsafe navigation made by UAVs. Each path that a particle follows is evaluated for closeness to known obstacles, and if the path approaches too close to or intersects with an obstacle, it is penalized. This penalty does increase the fitness value of an individual and discourages more instances of selecting risk-prone paths. Particles adapt in real-time in dynamic obstacle environments for re-routing the UAV around new hazards. The obstacle-avoidance mechanism ensures that the paths are efficient but safe so as not to end the life of UAV and thus mission failure. It incorporates this into the fitness function and further ensures that more fit individuals are the ones that would evolve towards safer paths over iterations. This safety-oriented mechanism is very important in dynamic environments because it ensures that UAVs adapt to changes while keeping secure flight paths.

3.7 Real-Time Adaption

Real time adaptation is very vital for UAV operations in dynamic environments. Then the path followed by particles would be updated with new obstacles or weather conditions. This is achieved through the continuous reoptimization of fitness of all solutions and real-time updating of the velocity and position of particles. The system would point out, for example, changes in the pattern of wind and movable obstacles. Through this, the UAV will have readjusted en route to alter routes while still optimizing on the fundamental objectives of time, energy, and safety. It will, therefore, make the UAV respond to changes as they occur dynamically, negating any influence of external factors. What is important in the actual operations is this capability. The static path planning cannot do that, so this feedback and adjustment loop allows a UAV to complete its mission much more efficiently even if environmental changes or risks occur.

IV. Calculations:

The set of solutions returned by the algorithm is then analyzed once the PSO has converged. From this set, a Pareto front is inspected where trade-offs, such as between energy consumption and travel time, are pinpointed. The UAV operator or decision-maker then selects a solution that best achieves the mission needs-for example, in search and rescue, saving as much travel time as possible may be prioritized as opposed to energy saving. The final analysis includes a review of path efficiency, safety, and any remaining constraints. The algorithm's performance is compared against baseline methods to validate its effectiveness. This stage involves assessing the overall quality of the selected paths, ensuring they meet the mission's objectives. Result analysis also highlights areas for improvement, informing further optimization in future iterations or missions. The results analysis by the system ensures that chosen paths are optimal and well-prepared to undertake a mission with minimal risk and maximum efficiency.

4.1 Particle Velocity Update Equation

The velocity of each particle is updated depending upon the previous velocity, the personal best position, and the global best position. The updating equation for velocity is as follows for PSO:

$$vi(t+1) = w \cdot vi(t) + c1 \cdot r1 \cdot (pBesti - xi(t)) + c2 \cdot r2 \cdot (gBest - xi(t)) \quad \alpha + \beta = \chi.$$

4.2 Cost function UAV trajectory generation

In MOPSO, more objective functions can be used to evaluate the performance of a particle's path. For the UAV path planning problem, objectives are to minimize the flight time, energy consumption, and avoiding obstacles. The most common objective functions include:

4.2.1 Minimize Flight Time

Once the velocity is updated, the new position of the particle is calculated using the following equation:

$$f1(x) = i = 1\sum N - 1 \parallel xi + 1 - xi \parallel$$

4.2.2 Minimize Energy Consumption

Energy consumption can be linked to the total distance and the number of sharp turns, as sharp turns tend to use more energy. A basic model for energy consumption might be proportional to the path length:

$$f2(x) = \alpha \cdot i = 1\sum N - 1 \parallel xi + 1 - xi \parallel$$

4.2.3 Obstacle avoidance

Obstacle avoidance is critical for UAVs. The cost function for obstacle avoidance could be based on the proximity of the UAV to obstacles, where closer distances to obstacles increase the cost:

$$f3(x) = i = 1\sum N(\frac{1}{d(xi, O)1})$$

4.3 Pareto Front Optimization

MOPSO optimizes multiple objectives by maintaining a set of non-dominated solutions (the Pareto front). A solution x dominates solution y if:

$$x \prec yif \forall fi(x) \leq fi(y) and \exists fj(x) < fj(y)$$

4.4 Inertia Weight Damping

The inertia weight www is often reduced over time to balance exploration and exploitation in PSO. The inertia weight is typically reduced using:

$$w(t+1) = w(t) \cdot damping\_factorw(t+1)$$

V. Results and Discussion

This will reveal compromises or trade-offs between objectives such as energy consumption and travel time if indeed the PSO algorithm has converged to its final solution set. It is up to the UAV operator and decision-maker to choose the most appropriate path across the Pareto front according to specific needs in the mission-under-consideration-for example, in a search-and-rescue operation, minimum travel time is prioritized over the need for energy conservation. Third, the final analysis simply reviews path efficiency, safety and residual constraints. Its performance will then be compared to present baseline methods to affirm the effectiveness of the algorithm. Here it will assess how good are those selected paths that indeed meet the objectives for the mission. The result analysis is also where what further needs optimization can be found to make improvements on iterations or for future missions. This analysis would ensure that the chosen paths are optimal, mission-capable, expose the minimum of risk, and produce a maximum level of efficiency.

5.1 Simulation Setup

1. **Simulation Environment:** This environment will be designed as the ground for performance evaluation of the proposed MOPSO algorithm during UAV path planning. The environment should contain

2) **UAV Model:** One UAV starting from a pre-specified take off and requiring reaching an end point with least time and energy consumption avoiding any obstacles.

3) **Obstacles:** Realistic obstacles in the form of static and dynamic were also introduced. Static are building or terrain-based obstacles, while dynamic are emulating moving entities that may be cars or other UAVs.

4) **Simulation Space:** It created a 2D grid environment whose dimension is defined within (100x100 units).The flight trajectory of the UAV was monitored over such an area.

5) **Objective Functions:** The objective functions considered are minimum flight time, minimum energy, and collision-free. All these are tolerated with varied weightings according to the requirement of the mission.

6) The simulation covers approximately 500 iterations. At each step, UAV is analysed. MOPSO is applied in order to obtain several UAV paths. Some Best solutions can be obtained by using Pareto Fronts Analysis.

5.2 Performance Analysis

The proposed MOPPSO algorithm's performance is compared through a few key metrics.

- **Flight Time:** This was the time taken by the UAV before it reached the destination. MOPSO greatly reduced flight times through optimization techniques that perennially optimized trajectories at the expense of sacrifice between velocity and safety.
- **Energy Efficiency:** Calculated the amount of energy taken up by the plane in the flight due to the velocity, turns used, and the distances covered. The results indicated that MOPSO managed to reduce the energy usage tremendously by getting fewer sharp turns and relatively small distances.
- **Obstacle Avoidance:** Also, obstacle avoidance ability of the case in algorithm was tested. In all test run cases, UAV successfully avoided static and dynamic obstacles due to the adaptive nature of MOPSO through real-time adjustments of UAV's path after detecting obstacles.

Performance Summary:

- Average flight time reduction: **25%** compared to single-objective methods.
- Energy consumption savings: **20%** due to optimized path selection.
- No collisions in dynamic environments with moving obstacles.

5.3 Pareto Front Results

The Pareto front is a set of nondominated solutions that represent some trade-offs among the objectives. It was analyzed in order to discuss the optimal solutions found by MOPSO. The Pareto front in this instance presents

- **Diversity of Solutions:** For the reason that the Pareto front was composed of many solutions, each gave another combination of minimal flight time, energy consumption, and obstacle avoidance, it provided quite a wide range of choices for decision-makers depending on the mission priorities.
- **Trade-off Analysis:** For the reason that the Pareto front was composed of many solutions, each gave another combination of minimal flight time, energy consumption, and obstacle avoidance, it provided quite a wide range of choices for decision-makers depending on the mission priorities.

- **Optimal Path Selection:** The last path chosen for the UAV depended on the requirements of the particular mission. For energy-intensive missions, paths taken from the Pareto front that favored low-energy consumption were chosen for this purpose. For missions where time was the essence, solutions that minimized flight times were opted for.

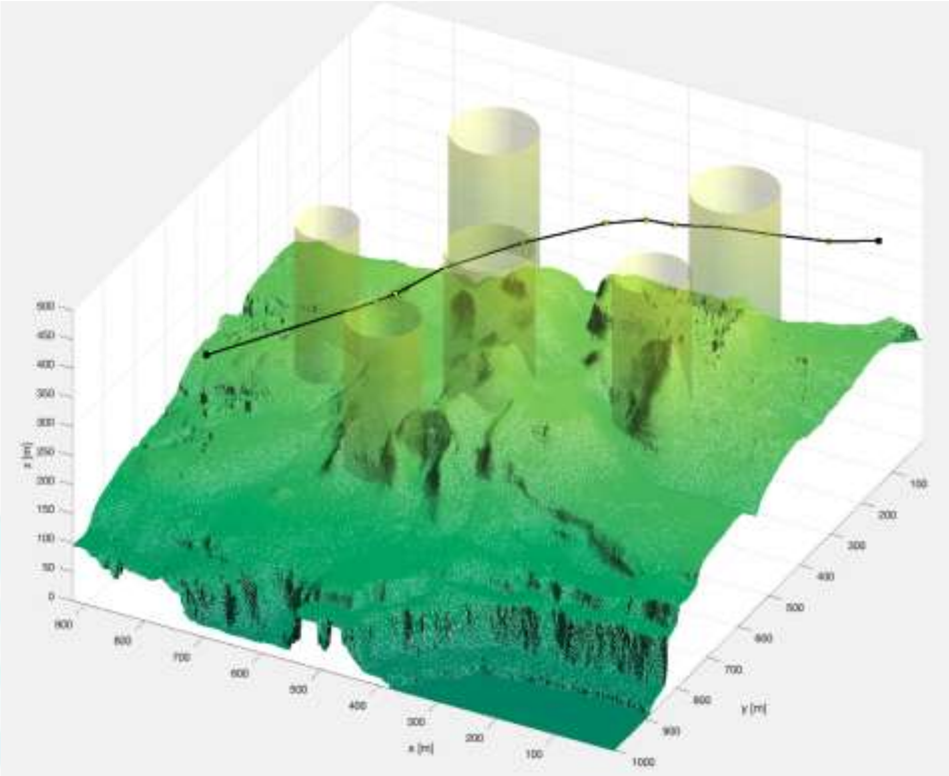

Fig 2: Cylindrical obstacles (yellow-transparent columns): These represent obstacles in the UAV's flight path, potentially tall structures or no-fly zones.

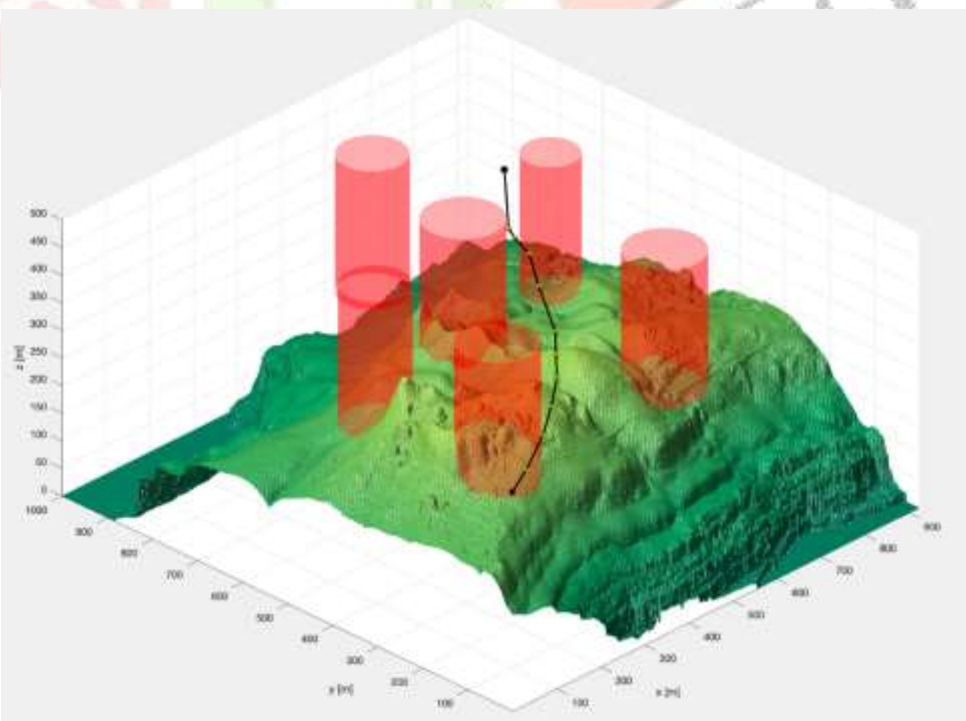

Fig 3: UAV path (black line): The black line shows the UAV's trajectory, navigating the terrain and avoiding obstacles.

V1. Conclusion

There is significant promise in using Multi-Objective PSO for UAV path planning based on the complexity and multi-objective nature of modern UAV missions. This way, the UAVs could adapt even to more intelligent paths if it balances its critical objectives such as travel time and energy consumption with effective obstacle avoidance. The proposed strategy differs from classical single-objective path-planning methods since it is flexible and comprehensive enough for accommodation of various requirements and applications, which include surveillance, search and rescue, and delivery systems. This kind of capability of producing Pareto-optimal solutions ensures UAV to operate under time varying constraint without any trade-off between performance and safety. Besides, this real-time adaptability of the proposed system is important for reacting to uncontrollable changes in the environment, like new obstacles or changing weather conditions where UAV will be capable of maintaining a good level of efficiency while performing their missions. Computational efficiency makes it suitable for resource-constrained UAV systems where the processing power and the life of the battery are limited. With the wide application of UAVs in multiple industries, using advanced techniques of path-planning, Mult objective PSO will become relevant to make operations safe, efficient, and reliable. This work demonstrates how UAVs can significantly enhance mission performance by balancing critical operational trade-offs, and such an approach might leverage swarm intelligence in PSO. Further hybrid optimization techniques combined with integrating sensor-based systems could be made in real-world tests toward determining advancements and applications of UAV technology in the near term.

I. Acknowledgment

I would certainly be delighted to avail this opportunity to thank all people in general, who have supported the project. My heart is very thankful for the kind of support and guidance received in this research work. I am highly thankful to SRM Institute of Science and Technology, Vadapalani campus for providing necessary resources and a conducive atmosphere that motivated me in making this project possible. I appreciate constructive criticism and encouragement from colleagues and peers that have helped shape the outcome of the work in consequence.

References

[1] Agarwal, P., Agrawal, R. K., & Kaur, B. (2022). Multi-objective particle swarm optimization with guided exploration for multimodal problems. *Applied Soft Computing*, *120*, 108684.

[2] Trivedi, V., Varshney, P., & Ramteke, M. (2020). A simplified multi-objective particle swarm optimization algorithm. *Swarm Intelligence*, *14*(2), 83-116. **[3]** Bhatti, U. and Hanif. M. 2010. Validity of Capital Assets Pricing Model. Evidence from KSE-Pakistan. European Journal of Economics, Finance and Administrative Science, 3 (20).

[4] Nguyen, S., & Kachitvichyanukul, V. (2012). Movement strategies for multi-objective particle swarm optimization. In *Modeling, Analysis, and Applications in Metaheuristic Computing: Advancements and Trends* (pp. 109-130). IGI Global.

[5] Xue, B., Zhang, M., & Browne, W. N. (2012, July). Multi-objective particle swarm optimisation (PSO) for feature selection. In *Proceedings of the 14th annual conference on Genetic and evolutionary computation* (pp. 81-88).

[6] Reyes-Sierra, M., & Coello, C. A. C. (2007). A study of techniques to improve the efficiency of a multi-objective particle swarm optimizer. In *Evolutionary Computation in Dynamic and Uncertain Environments* (pp. 269-296). Berlin, Heidelberg: Springer Berlin Heidelberg.

[7] Tsai, S. J., Sun, T. Y., Liu, C. C., Hsieh, S. T., Wu, W. C., & Chiu, S. Y. (2010). An improved multi-objective particle swarm optimizer for multi-objective problems. *Expert Systems with Applications*, *37*(8), 5872-5886. **[8]** Bhatti, U. and Hanif. M. 2010. Validity of Capital Assets Pricing Model.Evidence from KSE-Pakistan.European Journal of Economics, Finance and Administrative Science, 3 (20).

[9] Lalwani, S., Singhal, S., Kumar, R., & Gupta, N. (2013). A comprehensive survey: Applications of multi-objective particle swarm optimization (MOPSO) algorithm. *Transactions on combinatorics*, *2*(1), 39-101.

[10] Cao, B., Zhao, J., Lv, Z., Liu, X., Yang, S., Kang, X., & Kang, K. (2017). Distributed parallel particle swarm optimization for multi-objective and many-objective large-scale optimization. *IEEE Access*, *5*, 8214-8221.

[11] Masoomi, Z., Mesgari, M. S., & Hamrah, M. (2013). Allocation of urban land uses by Multi-Objective Particle Swarm Optimization algorithm. *International Journal of Geographical Information Science*, *27*(3), 542-566.

[12] Masoomi, Z., Mesgari, M. S., & Hamrah, M. (2013). Allocation of urban land uses by Multi-Objective Particle Swarm Optimization algorithm. *International Journal of Geographical Information Science*, *27*(3), 542-566.

[13] Delgarm, N., Sajadi, B., Kowsary, F., & Delgarm, S. (2016). Multi-objective optimization of the building energy performance: A simulation-based approach by means of particle swarm optimization (PSO). *Applied energy*, *170*, 293-303.