International Journal of Industrial Engineering, 31(3), 429-438, 2024

UNMANNED AERIAL VEHICLE PATH PLANNING USING WATER STRIDER ALGORITHM

Madhusudhanan Sampath[1], Abitha Kumari Duraisamy[2], Amalorpava Mary Rajee Samuel[3, *], and Yamuna Devi Manickam Malu[4]

[1]Department of Artificial Intelligence and Data Science
Prathyusha Engineering College
Chennai, India

[2]Department of Computer Science and Engineering
RMK Engineering College
Chennai, India

[3]Department of Electrical and Information Engineering
University of Witwatersrand Johannesburg
South Africa
[*]Corresponding author's e-mail: maryrajee.samuel@wits.ac.za

[4]Department of Computer Science and Engineering
Koneru Lakshmaiah Education Foundation
Vijayawada, India

This paper presents a multi-agent optimal path planning and obstacle avoidance using a water strider algorithm (WSA) based on Sequential Convex Programming (SCP). The outcome is to find optimal collision-free trajectories. The best collision-free trajectories with minimum control effect is needed in the multi-agent route planning technique, which makes use of a centralized WSA algorithm that can guide drones over congested environments while avoiding both static and moving objects. By applying convex constraints on the drones' such as acceleration, velocity input and jerk, the feasibility of the trajectory is ensured. The optimal trajectory path is iteratively created using SCP and followed by WSA. The outcome guarantees the correctness of the linearization. Since the optimization is centralized, it is possible to find a feasible collision-free path, and the results are validated pre-determined formation. It is shown that the WSA algorithm scales with $O^3(n)$, where n is the number of drones.

Keywords: Drone; Sequential Convex Programming; Water Strider Algorithm; Optimization.

(Received on December 6, 2023; Accepted on March 27, 2024)

1. INTRODUCTION

A drone is common in industries like planting agriculture, moving materials, military reconnaissance and surveillance (Yahia *et al.*, 2022; Fan *et al.*, 2021; Tang *et al.*, 2021). To prevent confrontations with manned aircraft, a lot of nations have imposed altitude restrictions on drone operations. These restrictions may change, but normally, drones are only allowed to fly up to a certain height, such as 120 meters (400 feet) above the earth. The main challenge in Path planning, however, is a significant static and dynamic obstacle for drone technology (Tan *et al.*, 2023). It should be done to make sure the lowest energy or time limitations are met, locate the best flight path to the board, and avoid obstacles in challenging conditions (Tan *et al.*, 2022; Jin *et al.*, 2022; Manh Duong *et al.*, 2021) The vehicle is able to make the decision on path coverage without the involvement of a remote operator; the level of autonomy might vary with onboard sensors. (Li *et al.*, 2016), Even though there are several ways to sense the surroundings, major sensing and positioning technologies like GPS and radar cannot be used to precisely position or sense Unmanned Aerial Vehicles (UAVs) in confined places like caverns or interiors. Because there are strong real-time requirements for the target recognition task in UAV environment sensing. A variety of onboard sensors are featured in drones for the purpose of making autonomous decision-making in situational awareness during operation. (Tang *et al.*, 2020).

DOI: 10.23055/ijietap.2024.31.3.9861 ISSN 1943-670X

1.1. Literature Review

Intelligent autonomous collision avoidance techniques can dramatically increase the safety of flights and prevent fatalities. The computational complexity and sparsity is quite less in handling traffic. (Poudel *et al.*, 2021, Wei Wang *et al.*, 2021). To give a quicker and smoother path, this method has been enhanced with an extra control force (Guo *et al.*, 2021; Meng, *et al.*, 2023). It is also used with the Hamiltonian function (Bai X *et al.*, 2021) or the ebbing horizon optimization (Chen *et al.*, 2015) to generate paths without going against the dynamic collision in connectivity requirements. The potential field method does not become an effective solution. When dealing with in-field local minima, its limits are also acknowledged (Kim *et al.*, 2013)

To create a solid and economical computational issue that could be observed using in-built computer, the optimal control problem was directly approached. A version of the Lagrange multiplier method that allows the incorporation of inequality constraints based on Karush-Kuhn-Tucker (KKT) conditions Mixed integer quadratic programming, RRT and A* algorithms. (Yang *et al.*, 2020). A simple non-linear programming problem (NLP) allows us to accelerate computation and implement it in an onboard computer. It is assumed that the drones are able to track the trajectory by setting feasibility constraints on the abilities of the drones. (Zhou *et al.*, 2021; Luo *et al.*, 2019; Wei *et al.*, 2021; Ajeil *et al.*, 2021) Using these collision-free optimal trajectories, groups of drones can move together in formation to a predetermined final location, avoiding both other drones and any obstacle encountered. (Rajee *et al.*, 2023) It must be noted that if there is no perfect a priori knowledge of the environment, this SCP algorithm can be easily formulated as a Model Predictive Controller (MPC) program, solving for n-steps while updating the knowledge of the world. It works well when it is combined with the water strider algorithm to handle the non-convex constraints. A route planning method based on the dual decomposition communication chain is proposed in (Shiri.*et al.*, 2020; Meng *et al.*, 2022) to improve the path selection, pheromone updating, and rollback approach of the basic WSA algorithm. The following are the major contributions.

(1) Representing the navigation of single-agent around static obstacles in non-convex domains
(2) Demonstrating the optimal path planning constraints with SCP for sequential path planning for multi-agent drone systems
(3) Demonstrating WSA optimization for a final time scaling method that avoids having multiple times with different discretization parameter settings and
(4) Validating WSA amended with SCP is done with a real-time implementable environment with a MATLAB simulation demonstration on a multi-agent drones.

A brand-new technique for optimizing drone operation in outdoor building environment is proposed.

2. PROBLEM FORMULATION

2.1. Dynamics of Trajectory

A liberalized dynamics model *et al.*, 2022 is used to simulate the drone's trajectory dynamics with $P_i[k] \in \mathbb{R}^3$ representing the drone's position at time k and h the discretization time step. The drone's velocity and acceleration are $V_i[k]$ and $a_i[k]$, respectively.

$$V_i[k+1] = V_i[k] + ha_i[k] \tag{1}$$

$$P_i[k+1] = P_i[k] + hV_i[k] + \frac{h^2}{2} a_i[k] \tag{2}$$

The definition of jerk is the pace at which acceleration changes. One can guarantee consistency in the drone's attitude by using a bounding jerk.

2.2. Objective Function

The main objective is to find the minimized path of a system consisting of N number of drones in different dimensions and coordinates (x,y,z)

$$arg\ min \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{K} w_{ij} a_i^T\,[k] a_j^T[k] a_i[k] \tag{3}$$

$$f_o = \sum_{i=1}^{N}\sum_{k=1}^{K}\|a_i[k]\|_2^2 \tag{4}$$

The objective function f_0 is improved to comprise gravitational effects, which involves the thrust vector of the drone to account for gravitational effects. The weights w_{ij} are modified with the water strider algorithm, and it is possible to achieve minimization over any combination of linear functions for acceleration. Goals for minimization could be control effort or path length. In this study, the control effort is kept to a minimum, which is the same as reducing thrust over the trajectory. we fix $w_{ij} = 1$ when i = j and $w_{ij} = 0$ else, which have a tendency to persuade flat trajectories with low curves.

It is obtained by first computing the 2D trajectory in steps and then adding the third dimension that is needed, as was previously explained. After obtaining the 3D trajectory, we assess it using each objective criterion independently. It provides us with a point in an 3D objective space with $\mathbb{R}^3$.

2.3. Path Planning Constraints

In order to confine the drones during simulation and prevent them from getting stuck in a corner, corner, wall, and floor constraints are used. The collision avoidance requirement is analogous to the corner avoidance restrictions, which are enforced using an exponential loss function (Equation 5) linearized around the preceding WSA solution.

$$e^{(C(P_i^x[k]- x_{cor}))} + e^{(C(P_i^y[k]- y_{cor}))} \geq 2\ \forall i,k \tag{5}$$

The position of the drone is

$$P_{min,l} \leq P_{i,l}[k] \ \leq P_{max,l} \quad l \in \{x,y,z\}\ \forall i,k \tag{6}$$

The Maximum acceleration constraints are

$$a_{min,l} \leq a_{i,l}[k] \ \leq a_{max,l} \quad l \in \{x,y,z\}\ \forall i,k \tag{7}$$

The actuator saturation limits are enforced to ensure continuous drone trajectories using equation (7).
A maximum jerk magnitude,

$$j_{min,l} \leq j_{i,l}[k] \ \leq j_{max,l} \quad l \in \{x,y,z\}\ \forall i,k \tag{8}$$

The jerk magnitude is enforced to ensure continuous drone trajectories.

2.4. Collision Avoidance

Dynamic and static obstacle avoidance constraints are applied to guarantee that the trajectory is not impeded at any point during the simulation, both the drone must always maintain an offset distance between them and the static obstacles. Equation 9's non-convex constraint describes this condition.

$$\left\|P_i[k] - P_j[k]\right\|_2 \ \geq d_{ij}\ \forall_{i,j}\ \mathrm{i} \neq \mathrm{j}\ \forall i,j \tag{9}$$

$$\left\|P_i^q[k] - P_j^q[k]\right\|_2 + \ \eta^T[\left(P_i[k] - P_j[k]\right) - \ P_i^q[k] - P_j^q[k]] \ \geq d_{ij}\ \forall_{i,j}\ \mathrm{i} \neq \mathrm{j}\ \forall i,j \tag{10}$$

where

$$\eta = \frac{P_i^q[k] - P_j^q[k]}{\left\|P_i^q[k] - P_j^q[k]\right\|_2} \tag{11}$$

The WSA algorithm finds an initial solution without the constraint in order to linearize it and then enforces the linearized constraint for all subsequent iterations. Given a small time increment and a small trust region, this linear approximation of the non-convex constraint is a reliable approximation.

2.5. Energy consumption

The energy consumption is important in ensuring with minimal energy. The main factors contributing this consumption are path length and turning angle. The energy consumption is represented as

$$E(x) = \frac{\sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{o=1}^{O} x_{ij} d_{io} \theta_{ij}}{\sum_{i=1}^{I} \sum_{j=1}^{J} d_{ij}} \tag{12}$$

I and J are path points, whereas x_{ij} is the drone variable from points i and j. The distance from x_i to obstacle x_o is denoted as d_{io}. The angle variable notation is θ_{ij} from points i and j.

3. WSA OPTIMIZATION METHODOLOGY

Water strider optimization is adapted to minimize the environmental impact of UAV systems. The optimization strategy is based upon the behavior of the water strider and its fitness function to reduce the cost of the system. The WSA algorithm draws its inspiration from how a water strider walks on the surface of water (Rajee *et al.*, 2023; Shiri *et al.*, 2022). In the Initialization, water striders generate population when looking for food, mating and establishing their territory. The fitness function is evaluated, and it is based on the drone's obstacle avoidance and other objective parameters. A given solution's ability to satisfy non-convex requirements, such as minimizing the distance to obstacles or maximizing clearance margins, should be measured by the objective function. Update the locations of the potential solutions in the search space to mimic the actions of water striders. Usually, this stage consists of two parts:

Exploitation: Based on the fitness values of the solutions, move them to promising areas of the search space. As we move closer to ideal solutions, this step promotes investigation of the problem space.

Exploration: To discover new regions of the search space, apply random perturbations to the drone's optimal solutions. This action promotes global exploration and prevents becoming locked in local optima. The optimization process should be directed towards workable solutions that satisfy the non-convex constraints using the constraint management method. Repeat steps 2-4 repeatedly until a termination requirement is satisfied. The Water Strider approach can successfully explore the search space to identify solutions that optimize the non-convex constraints in drone obstacle avoidance by iteratively updating the candidate solutions based on their fitness and utilizing constraint handling mechanisms. As a result, the WSA displays good positive feedback and strong robustness when planning a path.

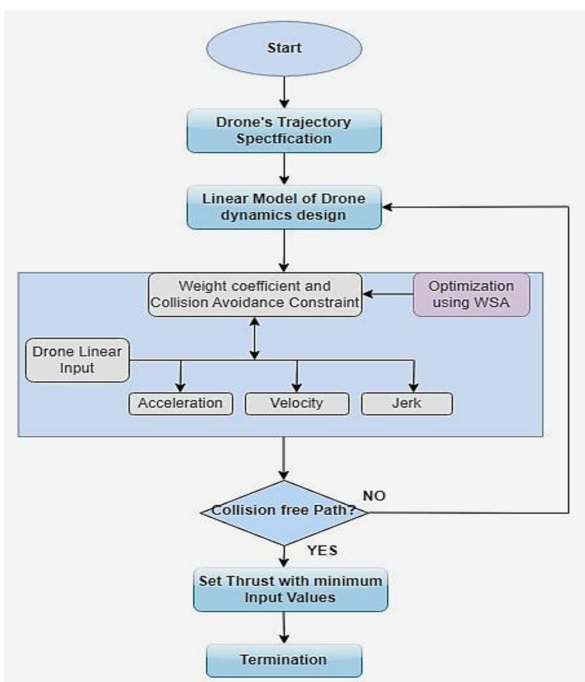


Figure 1. Flow chart of proposed WSA optimization

The following are the stages the WSA algorithm takes to design the drone's path: A decision tree-based implementation flow chart is shown in Figure 1 as the scheme.

Table 1. Stages in the WSA Algorithm

Water Strider Algorithm		
Initialize	:	Set of drones $i = \{1,2,3 \dots . N\}$
		Set of obstacles $S = \{1,2,3 \dots \dots . O\}$
		Set the weight coefficient w_{ij}^N
		Set the power values
Ensure	:	The initial position k, maximum velocity, maximum jerk magnitude
Compute	:	Maintain Near-Far drone distance
		Sort transmission powers in ascending order.
Check if	:	Collision free trajectory path
		$\left\|P_i[k] - P_j[k]\right\|_2 \geq d_{ij}$
Choose	:	weight coefficient w_{ij}^N $\forall Ui$
Set		Minimum thrust
		End if

Step 1: Set the settings, build the environment model, and then decide where to start and where to target.
Step 2 Assign fitness values to each trajectory based on an objective function that captures the optimization goals, such as minimizing distance, maximizing safety, and minimizing energy consumption.
Step 3: Determine the separation between drones in the matrix of path nodes.
Step 4: Determine the promising regions based on the fitness values and repeated iterations. The best water strider should be chosen as the next set of solutions after calculating the fitness values of all the striders in step 4.
Step 5: Track and store the best trajectory path so far as the optimal solution in accordance with Equations (13) and (14)
Step 6: Check to see if the termination condition has been met. Output the results if satisfied or move on to step 2 instead.

First, the box's side length (b), building pillar count (npillars), and building pillar radius (rpillar) must be determined. The user also specifies the simulation time (T), the avoidance radius for static and dynamic obstacles, and the number of obstacle sites (O). T and O establish the simulation's time step. While large T results in a highly high-dimensional optimization issue that causes lengthy calculations, small values of T are likely to result in a breach of the physical restrictions. The building pillar locations within the box can be produced randomly or using a deterministic grid. The initial and end states of the drones are selected randomly when the environment is generated. However, the user can predefine them to allow drones to fly in formation. Before the simulation begins, the initial conditions are checked to see if they satisfy all restrictions. All of the drones' initial trajectories are straight lines between the drone's start and finish points, with the thrust set to half the maximum input value. The main advantage of WSA is that it is relatively easy to implement and does not require complex mathematical formulations or extensive parameter tuning. This simplicity makes it accessible to users with varying levels of expertise in optimization and computational techniques.

Before following the intended trajectory, the drone first gathers samples from the area to check for obstructions. If any are found, it makes a note of its actual position for a predetermined period of time. If any obstacle is discovered, a polynomial regression is performed to fit the sensor readings to predict the upcoming position. In addition, a security distance is established based on the object size. The optimization algorithm is given the location of this new obstacle, which causes it to recalculate the new trajectory profile taking into account its location and utilizing the scheduled trajectory as a starting point for the iterative process. The operation is continued until the drone reaches the last waypoint on the trajectory. The initial conditions are verified to satisfy all constraints before the simulation is started. All initial paths of the drone are straight lines between the starting and end points of the drone, with the thrust set to half the maximum input value.

In the case study, the drone follows a path that is indicated by waypoints. The drone is equipped with sensors that enable the detection of things that were not included in the initial 3D representation of the space and illustration of an Optimal Route.

4. RESULTS AND DISCUSSION

In 14 various scenarios, the WSA algorithm's viability is examined, and the simulation results of the suggested model are presented and contrasted with those from earlier research.

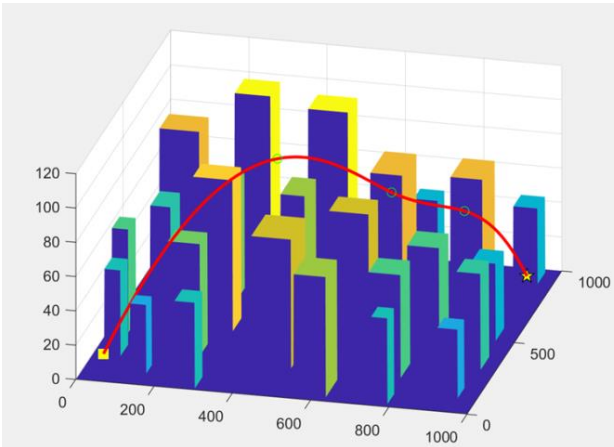


Figure 2. Urban Topography in 3D with optimal path and coverage plan

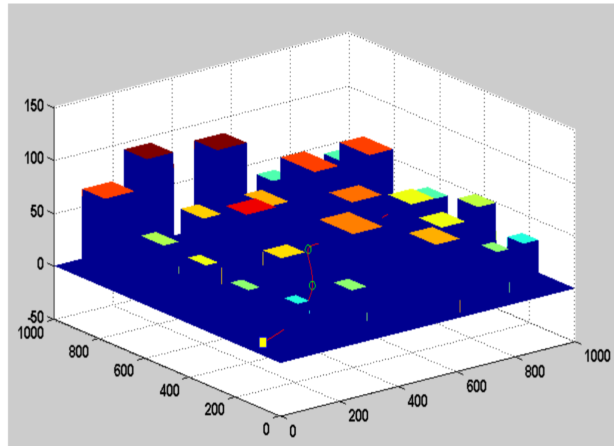


Figure 3. 3D urban topographical environment

The topographic map of an urban area is simulated, and it is shown in Figures 2 and 3.

The top of the buildings is represented by the orange shade portion, while the basins are represented by the blue depression portion. Starting and ending coordinates can range from 1 kilometer to 22 km.

1) Computational Effort: In Figure 8, the average run time of the WSA (averaged over 10 runs) is displayed against the number of drones. The WSA algorithms scale with n the number of drones in $O(n^3)$ units.
2) Random Cluttered Environment: The algorithm can choose the best paths for several drones in a crowded, random environment that is confined to a single room, avoiding obstacles like corners and other drones. In an isometric view, Figures 2 and 3 depict five drones dodging 24 pillars. The reduced control effort, equal to 46.988 m/s^2, is significantly less than the control effort required to maneuver 15 drones without pillars.
3) Deterministic Environment: In addition to a chaotic environment that is congested, the algorithm is also capable of locating the best paths in deterministic environments, such as grids of pillars. The optimal path was planned using the different known algorithm methods using parameter optimization. The response variable results of mean fitness, CPU time and best solution are shown in Table 2.

Table 2. Comparative result Analysis

Algorithm	Mean Fitness	Mean CPU Time	Best Solution
A-Star	1.702	838 ms	1/14
GA	0.9611	35634 ms	13/14
ACO	1.72	1400 ms	13/14
Proposed WSA	1.6	580 ms	14/14

The path length of the on-board flying in the urban environment was set to 100 km after optimizing the essential parameters of the enhanced WSA. Running time was 3.56 s, whereas others running time varied from 5.54 s to 7 s.

The minimum distance between the lined path and obstacle (d) is set as 0.2 meters and is being used by the drones to avoid one another. Any environment in real life may be discretized into a collection of pillars, and WSA can typically identify practicable, ideal collision-free trajectories.

It should be noted that the WSA algorithm found it difficult to discover a workable solution as the forms got more complicated. When the starting locations were set up in a half circle, and the drones had to discover a route out of the circle, it was unable to come up with any workable solutions. It is trivial to extend the method to include the formation flight of drones. With formation flight is meant since the drones are aware of each other locations, they can collision-free transition to any desirable shape.

The likelihood of choosing a task assignment is updated in the current WSA-based algorithms following the construction of each water strider assignment. Then, using the feedback, each task's pheromone value is updated. It is shown in Figure 4

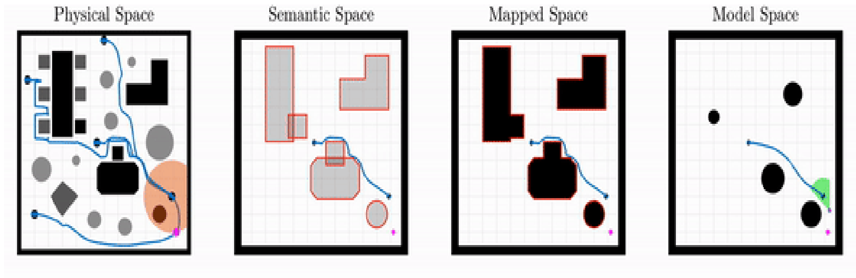
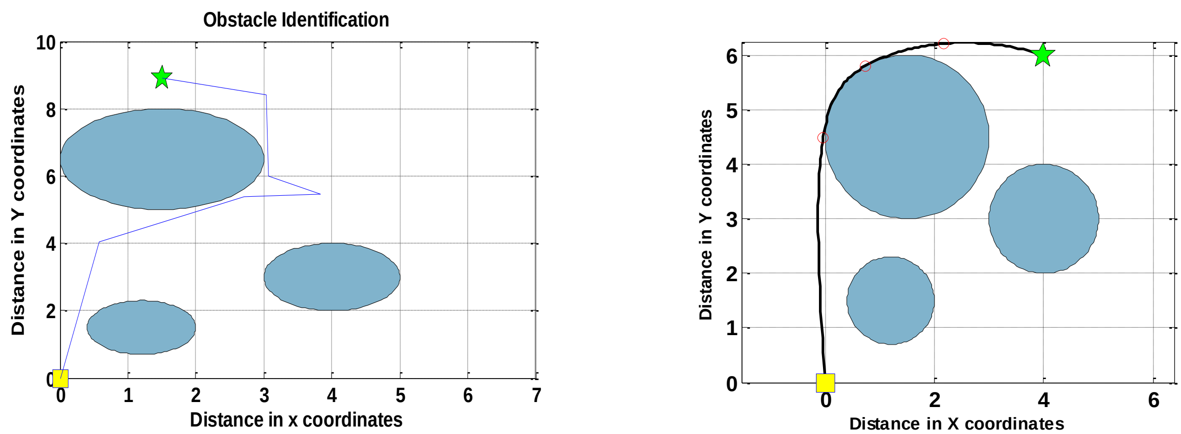


Figure 4. Drone Trajectory path over different domain



(a) (b)

Figure 5. Drone Trajectory path over obstacles

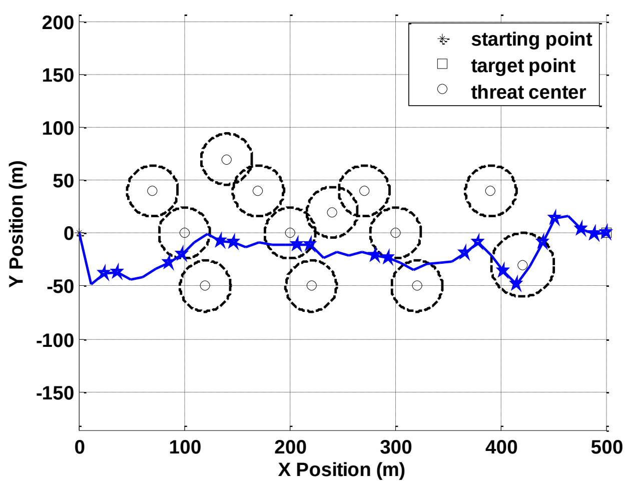


Figure 6. Drone Trajectory path over circular space

In Figures 5 and 6, a 3D urban topographical environment is created with 24 building pillars where drones must move between randomly generated initial and final conditions while keeping a minimum distance of 0.5 meters between each drone.

It is shown in Figure 5. Each and every drone must begin and terminate with a zero initial velocity. The trajectory reduces the energy consumption by reducing the acceleration and velocity whenever the obstacles are about to be identified.

In Figure 7, eight drones navigate a grid of 25 pillars without colliding, using a total control effort of 66.7 m/s^2.

The convergence of the WSA algorithm is shown in Figure 8 and it plots the average WSA run time vs. the quantity of static barriers (averaged over 10 runs). It is obvious that there are a lot less static barriers. It is evident that static obstacles require significantly less computation than moving drone obstacles.

Computational Work: The computational analysis report is given in Figures 9 and 10, where the WSA with SCP algorithm outperforms better than the previous algorithm. The number of static obstacles is plotted on the x-axis, and the algorithm scales with $O^3(n)$. WSA exhibits slower convergence rates compared to some traditional optimization algorithms. This can be problematic when dealing with large-scale optimization tasks where efficiency is critical.

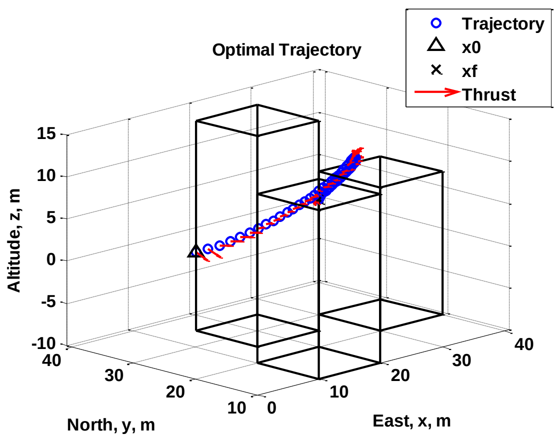


Figure 7. Optimal Trajectory path over pillar space

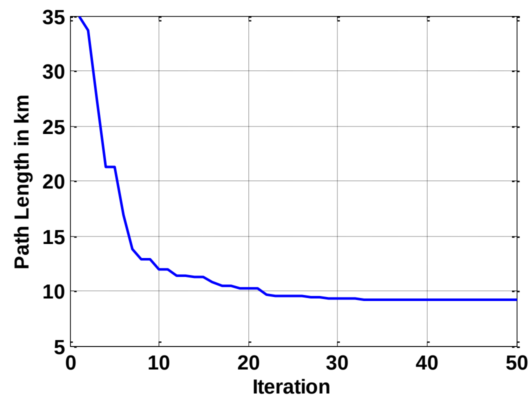


Figure 8. Convergence after optimization

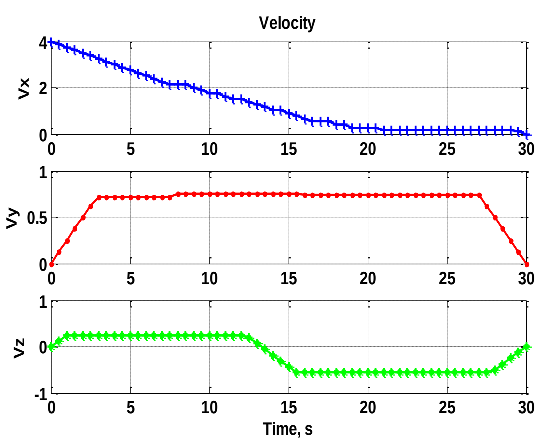


Figure 9. Computation Run time of drone's velocity

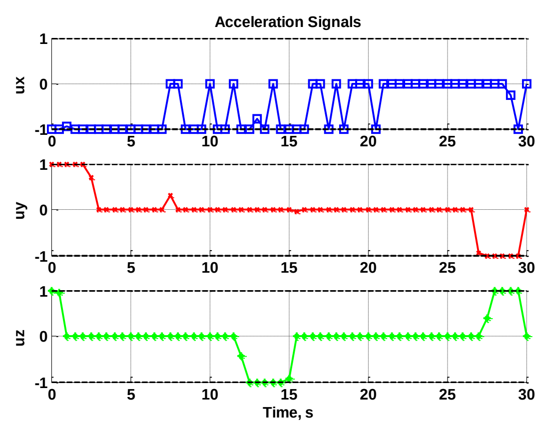


Figure 10. Computation Run time of drone's acceleration signals

This work can be expanded to increase the resilience, efficiency, and convergence rate of the method. For the exploitation and exploration phases, this can entail creating novel strategies, improving the mechanisms for addressing constraints, and introducing adaptive to dynamically modified algorithm parameters. Overall, the Water Strider algorithm's future potential resides in developing its theoretical underpinnings, enhancing its algorithmic elements, and investigating its applicability to numerous optimization issues across many fields. We have tested the system in various simulation environments and scenarios to ensure its reliability and effectiveness. This could involve the integration of both planning and avoidance and simulation-based testing with specific applications or platforms where it can have practical significance. For example, in autonomous vehicles, warehouse robots, or search and rescue drones.

5. CONCLUSION

The WSA with Sequential convex programming is designed and tested through a vigorous simulation environment in this paper to address the drawbacks of traditional SCP. The energy consumption was included in the pheromone update technique, such as autonomous multi-vehicle navigation through congested urban areas without colliding with other vehicles. The method presented in this paper generates collision-free trajectories for multiple agents and multiple obstacles, which was achieved by linearizing the non-convex constraints. By linearizing the non-convex requirements, the method creates collision-free trajectories for numerous agents and multiple barriers. By adding bounding restrictions to the vehicle, the linear approximation of the drone trajectory dynamics was guaranteed to be accurate. As demonstrated by the trajectory re-planning challenge. Although the overall command and control infrastructure still needs refinement, the same capability appears doable for many flights, adding still another level of mission flexibility. Continued research and development efforts can help unlock its potential and contribute to its wider adoption in practical applications.

REFERENCES

Ajeil, F.H., Ibraheem I.K., Azar A T. and Humaidi, A.J. (2021). Grid-Based Mobile Robot Path Planning Using Aging-Based Ant Colony Optimization Algorithm in Static and Dynamic Environments. Sensors 20(7), 1880.

Bai, X., Jiang, H K., Cui, J.J., Lu, K., Chen, P.Y. and Zhang, M.(2021). UAV Path Planning Based on Improved A* and DWA Algorithms. Int. J. Aerosp. Eng. 1(12), 585.

Chen, Y., Cutler, M. and How, J P. (2015). Decoupled Multiagent Path Planning Via Incremental Sequential Convex Programming," in 2015 IEEE International Conference on Robotics and Automation (ICRA), May 2015 5954–5961.

Fan, M. and Akhter, Y. (2021). A Time-Varying Adaptive Inertia Weight Based Modified PSO Algorithm for UAV Path Planning. ICREST 2021 - 2nd International Conference on Robotics, Electrical and Signal Processing Techniques. 573–576. DOI: https://doi.org/10.1109/icrest51555.2021.9331101.

Guo, J., Liang, C., Wang, K., Sang, B., and Wu, Y. (2021). Three-Dimensional Autonomous Obstacle Avoidance Algorithm for UAV Based on Circular Arc Trajectory. Int. J. Aerosp. Eng. 1(13): 204-211

Kim, K.C., Sun J.U. and Lee, G.M. (2013). A Hierarchical Approaches To Vehicle Routing and Scheduling With Sequential Services Using The Genetic Algorithm. International Journal of Industrial Engineering: Theory, Applications, and Practice, 20(1):245-257.

Li, P., and Zhu, H. (2016). Parameter Selection for Ant Colony Algorithm Based on Bacterial Foraging Algorithm. Math. Probl. Eng. 1(12): 1-14.

Luo, Y., Guo P. and Zhang, M. (2019). A Framework of Ant Colony P System. IEEE Access 7, 157655–157666

Manh, Duong., Phung, Quang., Phuc., and Ha. (2021). Safety-Enhanced UAV Path Planning With Spherical Vector-Based Particle Swarm Optimization, Applied Soft Computing, 10(7):107376. DOI: https://doi.org/10.1016/j.asoc.2021.107376.

Meng, X., Zhu, X. and Zhao, J. (2023). Obstacle Avoidance Path Planning Using The Elite Ant Colony Algorithm for Parameter Optimization of Unmanned Aerial Vehicles. Arab J Sci Eng 48, 2261–2275 2023. DOI: 10.1007/S13369-022-07204-7

Poudel, S., and Moh, S. (2021). Hybrid Path Planning for Efficient Data Collection in UAV-Aided Wsns for Emergency Applications. Sensors (Basel). 21(8): 2839. DOI: https://doi.org/10.3390/s21082839.

Qibing, Jin., Chuning, Tang., and Wu, Cai. (2022). Research on Dynamic Path Planning Based on The Fusion Algorithm of Improved Ant Colony Optimization and Rolling Window Method IEEE Access. 10(1): 28322-28332. DOI: https://doi.org/10.1109/access.2021.3064831.

Rajee, S.A.M., Merline A. and Devi M M Y. (2023). Game Theoretic Model for Power Optimization in Next-Generation Heterogeneous Network. Sivip 17, 3721–3729. DOI: https://doi.org/10.1007/s11760-023-02599-8.

Shiri, H., Park J. and Bennis M. (2020). Remote UAV Online Path Planning Via Neural Network-Based Opportunistic Control. IEEE Commun. Lett. 9(6), 861–865

Tan, Y., Ouyang. J., Zhang. Z., Lao, Y. and Wen, P. (2023). Path Planning for Spot Welding Robots Based on Improved Ant Colony Algorithm Robotica. 41(3): 926-938. DOI: https://doi.org/10.1017/s026357472200114x.

Tang, A.D., Han T., Zhou H. and Xie, L.(2021). An Improved Equilibrium Optimizer With Application in Unmanned Aerial Vehicle Path Planning. Sensors. 21(5), 1–21. DOI: https://doi.org/10.3390/s21051814.

Tang, B., Kui, X., Pang, M. and Zhu, Z. (2020). Multi-Robot Path Planning Using An Improved Self-Adaptive Particle Swarm Optimization. Int. J. Adv. Robot. Syst. 17(5): 1–19.

Wang, X., Gursoy, M C., Erpek, T. and Sagduyu, Y E. (2022). Learning-Based UAV Path Planning for Data Collection With Integrated Collision Avoidance IEEE Internet of Things Journal. 9(17): 16663-16676. DOI: https://doi.org/10.1109/jiot.2022.3153585.

Wei, Wang., Chao, Fang., and Ting, Liu. (2021). Quality-Oriented Hybrid Path Planning Based on A* and Q-Learning for Unmanned Aerial Vehicle IEEE Access *Pub* 10 7664-7674. DOI: https://doi.org/10.1109/access.2021.3139534.

Wei, X. and Xu J. (2021). Distributed Path Planning of Unmanned Aerial Vehicle Communication Chain Based on Dual Decomposition. Wirel. Commun. Mob. Comput. 1–12

Yahia, H.S. and Mohammed, A.S. (2022). Path Planning Optimization in Unmanned Aerial Vehicles Using Meta-Heuristic Algorithms: A Systematic Review. Environ Monit Assess. 195(1): 30-38 DOI: https://doi.org/10.1007/s10661-022-10590-y.

Yang, Liuqing., Guo, J. and Liu, Y. (2020). Three-Dimensional UAV Cooperative Path Planning Based on The MP-CGWO Algorithm. International Journal of Innovative Computing, Information and Control, 16(3), 991–1006. DOI: https://doi.org/10.24507/ijicic.16.03.991.

Zhou, X., Gao, F., Fang, X. and Lan, Z. (2021). Improved Bat Algorithm for UAV Path Planning in Three-Dimensional Space. IEEE Access, 9, 20100–20116. DOI: https://doi.org/10.1109/access.2021.3054179.