Developing a model for unmanned aerial vehicle with fixed-wing using 3D-map exploring rapidly random tree technique

Omar I. Dallal Bashi[1], Husamuldeen K. Hameed[2], Yasir Mahmood Al Kubaisi[3], Ahmad H. Sabry[4]
[1]Mosul Medical Technical Institute, Northern Technical University, Mosul, Iraq
[2]Department of Telecommunications and Information, High Institute of Telecommunications and Post, Baghdad, Iraq
[3]Department of Sustainability Management, Dubai Academic Health Corporation, Oud Metha, Dubai, United Arab Emirates
[4]Department of Computer Engineering, University of Al-Nahrain, Baghdad, Iraq

Article Info

Article history:

Received Nov 20, 2022
Revised Apr 17, 2023
Accepted May 3, 2023

Keywords:

Fixed-wing
Motion planning
Obstacle avoidance
Rapidly random tree
Unmanned aerial vehicle

ABSTRACT

While the motion planning algorithms consider the obstacles that were known in the map, it is possible to use obstacle avoidance algorithms to take over and send commands to theunmanned aerial vehicle (UAV), when there is an unknown obstacle on the way. The rapidly random tree (RRT) algorithm is used to plan paths for a quad-copter or a fixed-wing UAV. This work develops a model for UAV with fixed-wing using a 3D map exploring the RRT technique. The first step is to obtain a 3D occupancy map from the map data stored in the UAV city to provide a map with some pre-generated obstacles. The contribution of this work is to use RRT planning for 3D state space, where the motion segment or motion primitive connecting the two consecutive states should be defined in a 3D space while satisfying the motion constraints of a UAV. The simulation includes setting up a 3D map, providing the starting and destination pose, planning a way using RRT and 3D Dubins moving primitives, smoothing the acquired trajectory, and simulating the UAV flight. The results obtained demonstrate that the smoothed-generated waypoints significantly improved tracking in general with shorter paths.

Corresponding Author:

Ahmad H. Sabry
Department of Computer Engineering, University of Al-Nahrain
Baghdad, 64074, Iraq
Email: ahs4771384@gmail.com

1. INTRODUCTION

Unmanned aerial vehicles are now a common tool in the business, industry, and science sectors due to their lower operational costs and enhanced robustness. They are appealing for a variety of applications due to their speedy mapping and surveillance of broad areas. In emergency situations like avalanches [1], earthquakes [2], floods [3], creating a map in real-time are crucial for first responders. It may also enable other aerial vehicles (UAVs) to localize themselves without the need of global positioning systems [1], [4].

The work flow starts by loading the map of a 3D environment then selects the desired start and goal locations on this map. Then the path planning algorithm works to find an obstacle free path between the start and the goal location. While doing that, visualizing the path helps us to see what optimizations maybe required, this brings us to the last step that is optimizing or making the path smoother. The workflow of the 3D unmanned UAVmotion planning and obstacle avoidance can be demonstrated as in Figure 1.

Several questions that may arise here such as; where does the map come from? [5] or how will the UAV follow the planned path and avoid unknown obstacles? The planning and control algorithms of UAV include generating 3D occupancy maps, where it is possible to test the planning algorithms [6]. The UAV control algorithm can also create a 3D map by scanning a simulated world using a LiDAR sensor model [7]. Once obtaining the waypoints from the motion planning algorithms, it is possible to model the aerodynamics constraints of a UAV and tune waypoint following algorithms to make the UAV follow the planned path. While the motion planning algorithms consider the obstacles that were known in the map, it is possible to use obstacle avoid algorithms to take over and send commands to the UAV, when there is an unknown obstacle on the way.

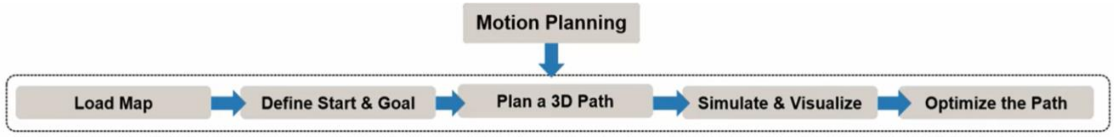

Figure 1. Workflow of the 3D UAV motion planning and obstacle avoidance

Therapidly random tree (RRT) algorithm is used to plan paths for a quad-copter or a fixed-wing UAV [8]. This work is focused on path planning for fixed-wing UAVs. A UAV with fixed-wing is non-holonomic in character that has to obey aerodynamic restrictions such as airspeed, flight path angle, and maximum roll angle as move among waypoints [9].

Small UAVs have demonstrated considerable potential as tools for swiftly and affordably gathering aerial photography. A team of tiny UAVs rather than a single giant UAV has also showed promise as a more reliable, quicker, and less expensive way to capture visual data over a vast region. Unfortunately, small unmanned aerial vehicles still lack the autonomy necessary to be trusted to gather high-quality aerial imagery without guidance or supervision from humans [10]. Small fixed-wing UAVs are getting better and better at flying in confined spaces and at low altitudes [11]. Physically, a class of contemporary small fixed-wing unmanned aerial vehicles is capable of doing extraordinary aerobatic feats [12]. Path planning is essential forUAVs to carry out autonomous military missions in difficult situations [13], [14]. One of the important tasks in completing the mission ofUAVs is path planning to prevent collisions with moving objects or other UAVs in motion [15].

Guo *et al*. [13] presented a new method named as a star flight-cost exploring rapidly random tree (FC-RRT) expanding the traditional RRT star by dealing with the flight constraints and safety requirements for UAVs within a compound 3D map. Although this algorithm solved some UAV path planning issues by the flight constraints, the flight cost function, and front-end by inspiring new nodes expanding, it was complicated and requires large memory and processing speed. Levin *et al*. [11] addressed these problems by automating a fixed-wing flight of UAV. The experiments conducted used only on-board sensing and computing but with an extremely inhibited environment. Ramana *et al*. [10] also proposed a UAV fixed-wing motion planning by employing RRTs with known destination and starting locations over a map with statically obstacles. This work generated an optimal path close to obstacle loaded map in a extremely short time gap, which was concentrated on the RRT algorithm on a 2D environment. Although this work combined a method to pursuit line of sight guidance law and guidance law to track 3D amp-based UAVs, its finding limited to demonstrate the guidance law performance only.

Jackson [16] proposed increasing the autonomy level UAVs in independent and reliable gather worth aerial images. The study presented an algorithm to control small fixed-wing UAVs to optimize image quality by on board cameras. The study discussed the fixed-wing UAVs in RRT model. However, the considered iterative sampling Gaussian strategy of the RRT didn't show how to design a 3D random occupancy map. Lee and Shim [17] proposed a path planning approach according to RRTs for UAVs with fixed-wing. The technique used a pre-defined movement set to expand RRT and to reproduce the dynamic ability of the path towards the destination point with collision avoidance. However, this work didn't determine the exact target positions with respect to the arrival time. Levin *et al*. [12] offered a methodology to include the knife-edge and maneuvers of RRT-based motion planning. Although a demonstration for the interaction of knife-edge maneuvers with the motion planning was discussed, the motion planner was limited to a modified version of RRT method. Ma *et al*. [18] presented a fixed-wing UAV real-time obstacle avoidance algorithm in also complex surrounding map. The RRT technique is employed here to adapt the RRT with fixed-wing environment with some extensions. However, this method only provided worthy results in the autonomous stage of UAV elevation.

Huang and Savkin [19] focused on applying a UAV powered by solar within mountain sites environment for the purpose of rescue and safety. This algorithm initially found a feasible path to satisfy the required energy residual and next shortened the waypoints when there are some abundant energy residual on the ending. However, this method successful in restricted environment. Saravanakumar *et al*. [15] discussed the growth of UAV path planning techniques due to sampling. The RRT was chosen with a small number of development were made in RRT by shortening the node connection techniques to create feasible waypoints satisfying the working surroundings constraints. However, the proposed 3D-map exploring RRT was not effectively used to optimize the path towards the destination target. Huang *et al*. [20] mainly discussed the rapid reconstruction and obstacle avoidance of UAV formation. This work used a hybrid path planning approach according to a dynamic model of potential field fluid RRT to recover the capability of UAV formations in dynamic complex environments [21].

The aim of the study is to develop a model for UAV with fixed-wing using 3D-map exploring theRRTtechnique, which will make it possible to generate smoothed waypoints as an improved tracking with the shorter path as compared with traditional RRT technique. To achieve this aim, the following objectives are accomplished:

- To generate 3D random occupancy map using an automatic random positionings with different dimensions and number of obstacles.
- To simulate of a UAV follows path planning with Dubinspath.
- To simulate UAV trajectory and smooth Dubinspath.

2. METHOD

This work presents a demonstration of motion planning for an UAV with fixed-wing based on exploring RRT technique [22] for a 3D map known destination and start pose. A UAV with fixed-wing is a naturally nonholonomic and should follow aerodynamic restrictions such as the airspeed, flight path angle, and maximum roll angle when there is a movement between two points on the map. The approach includes the following:i) setting up a 3D map, ii) providing the starting and destination pose, iii) planning a way using RRT and 3D Dubins moving primitives, iv) smoothing the acquired trajectory, and v) simulating the UAV flight.

2.1. Setting the random number generator seed for repeatable result

By looking at this map, we can select a start and goal location in the free space. Since it's a fixed wing UAV, we need to provide $x\ y\ z$ and a heading angle for both the start as well as the goal state. The first step is to obtain a 3D occupancy map from the map data stored in the UAV city. This provides a map with some pre-generated obstacles. We generate 3D random occupancy map through an automatic or random positions and selection for the varying dimensions number of obstacles over the map according the flowchart and the corresponding algorithm shown in Figure 2.

2.2. 3D Dubins motion primitives to plan a path with rapidly random tree

Using 3D Dubins motion primitives in conjunction with rapidly exploring random trees (RRT) is a powerful approach for path planning in three-dimensional space, particularly for autonomous drones, robots, or vehicles. The Dubins motion primitives allow you to generate feasible paths for vehicles with minimum turning radii. The motion planners with RRT designs a scan tree increasingly due to arbitrary models for a known state-space. The tree ultimately extents the explore gap and joins the starting and the destination states.

2.3. Defining the state space object

We define a custom state space to incorporate the UAV aerodynamic constraints. We use the workspace goal region property from the state space class to specify the target pose and the target goal area about it. This helps to define how big the goal workspace gold area about the target pose is. That is employed to bias the gold area sampling. Now we have our state space defined, next we define the state validator.

The new in this work is that here we use RRT planning for 3D state space, that means the motion segment or motion primitive connecting the two consecutive states should be defined in a 3D space while satisfying motion constraints of a UAV. Now the UAV Dubian's connection object needs to be defined within a state space before we dive further into the implementation, let's quickly refresh on how we define the state space and the planner. The four-step motion planning workflow we used in this UAV work is shown in [23]. We use a MATLAB helper function UAV state space, which helps to define the state space and provide the UAV constraints. The state space is defined with the position and the heading angle for the UAV. We know that the flight path angle limit, airspeed and highest roll angle will be used to define the UAVS Duban's path segments. We will also set the bounds to identify the orientation and position restrictions of UAVs. This work offers a pre-identified path planning state-space. The state spaces can be identified by: $[x\ y\ z\ heading\_Angle]$, where $[x\ y\ z]$ indicates the UAV location, the $heading\_Angle$ in radians. We use a Dubins-based object to represent the kinematic UAV model, which is restricted by an optimal flight path angle, airspeed, and roll angle. Creating the object state-space through indicating the optimal flight path angle, airspeed, and roll angle, this bounds the UAV properties by (name-value) pairs. Using (name-value) pairs argument to indicate the orientation and position UAV boundaries by $4-x-2$ matrix within the 3D map occupancy where the final row refers to the heading angle in the range $[-\pi,\pi]$ radians. Setting the bounds of the threshold for the workspace according to the target destination pose. The workspace goal region approach uses bias sampling to determine how large the target workspace goal area should be around the goal posture.

2.4. Defining the state validator object

The purpose of state validator is to validate the connectivity between the sample states. It determines that a state is valid or invalid based on the information if the state is occupied or free on the plan. A movement between each state is satisfied just if all the in-between conditions are satisfied, which leads that the UAVs do not go through some occupied cell on the environment. Here we use out-of-the-box state validator for 3D state spaces from MATLAB navigation toolbox [24]. So we create a validator occupancy map 3D object by identifying the inflated map and the object state space and. After that we set the confirmation space with meters to interpolate among the states. The object of validator occupancy 3D map indicates invalid state when thexyz-position is engaged within the map. A moving between couple states is applicable just when all in-between states are applicable indicating that the UAV does not go by any occupied map position.

2.5. Executing path planning RRT

Performing RRT-based path planning in 3D map, where the planners search for a collision-free path appropriate to a flight with fixed-wing. Now we have the required inputs ready for the path planner to work. We create the planner object and specify the state validator and state space as an input. We adjust the properties for the planners and the properties we specify here are maximum connection distance, which is the maximum length of a motion allowed in the tree another property is the goal bias, which is the probability of choosing the goal state during state sampling. Another in the other one is max iterations which is maximum number of iterations. We also identify a tradition target function, which decides to achieve the target pathway if the Euclidean space to the goal has a threshold less than 4 m.

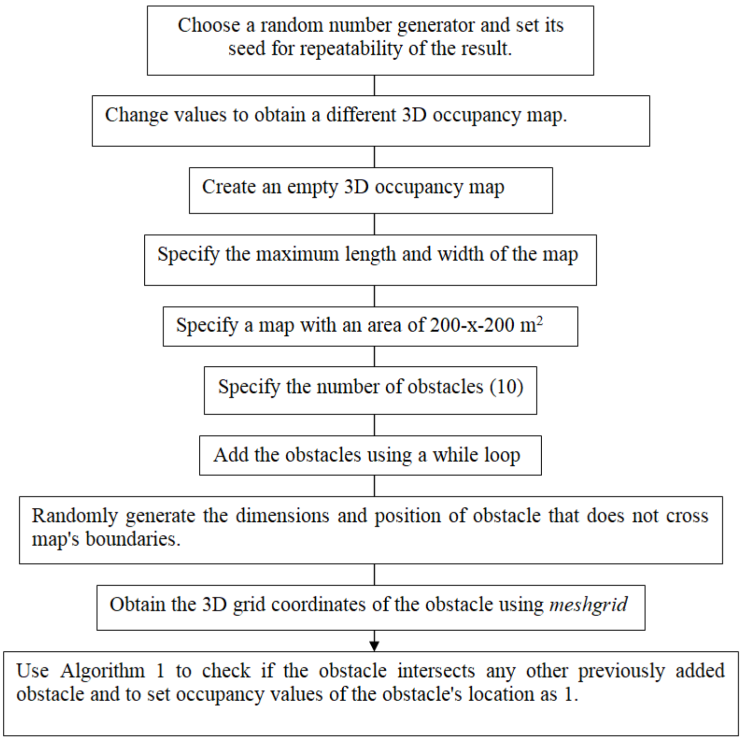

Figure 2. Flowchart for generating a 3D random occupancy map

Now based on all the specified properties for the planner the plan function executes the path planning to conduct the path planning with RRT into 3D space. It outputs path object where we can see all the states that the path includes.

```
Algorithm generate 3D random occupancy map
Obstacle Number = 1;
while obstacle Number <= number Of Obstacles
    width = randi ([ 1 50 ],1 );  % The biggest integer in sample intervals to get height, length, and width
    length = randi ([ 1 50 ],1 ); % can be altered as required to generate unlike occupancy maps.
        height = randi ([ 1 150 ],1 );
        x_Position = randi ([ 0 map Width-width ],1 );
        y_Position = randi ([ 0 map Length-length ],1 );
            [x_Obstacle, y_Obstacle, z_Obstacle] = meshgrid ( x Position: x Position+width, y Position: y
    Position +length, 0:height );
        Xyz_Obstacles = [x_Obstacle (:) y_Obstacle (:) z_Obstacle (:)];
            Check Intersection = false;
                for i = 1:size(xyz_Obstacles,1)
                    if check Occupancy(omap 3D,xyz_Obstacles(i,:)) == 1
                            check Intersection = true;
                            break
                     end
            end
            if check Intersection
                    continue
                end
                Set Occupancy(omap 3D,xyz Obstacles,1)
                obstacle Number = obstacle Number + 1;
End
[x_Ground, y_Ground, z_Ground] = meshgrid (0:map_Width,0:map_Length,0);
Xyz_Ground = [ x_Ground (:) y_Ground (:) z_Ground (:) ];
Set_Occupancy (omap3D, xyz_Ground,1)
Display the final occupancy map. Figure ("Name","3D Occupancy Map"), show( omap3D )
```

2.6. Simulating of a UAV follows path planning

Now to visualize the plant path we interpolate it according to based on the UAV dubin's relations and design these interpolated conditions as a red line and the simulated states is plotted as a green line.

This work provides a helper function to simulate the UAV and can visualize the UAV following this planned path. It requires these inputs such as the set of path, time to reach the goal and airspeed. This helper command uses fixed wing supervision form to simulate the UAV actions according to input control created by the path points. Both of these are part of the UAV toolbox. It also uses the waypoint follower to simulate the UAV following these waypoints:

- Visualizing that path planning.
- Interpolating the produced path planning according to the UAV Dubins joints.
- Plotting the states of interpolation process with redlines.
- Simulating the UAV flight employing a MATLAB commands "SimulateUAV" that involves using the airspeed, time to reach the goal, and waypoints (according to path length and airspeed). This command employs a command "fixedwing" guide modeling for simulating the UAV performance according to the way-path points of the generated control inputs.
- Plotting the states that are simulated with a blue line.
- Visualizing the 3D map.
- Plotting the UAV Dubins connections with interpolated path planning.
- Plotting fixed-wing guidance model of the simulated UAV trajectory.
- Adding a buffer and computing the whole flight time.

To make this path smoother, we simplify the 3D durability pad by employing the patch smoothing technique proposed in this work.

2.7. Simulating UAV trajectory and smooth Dubins path

There is another helper function that is provided to implement the path smoothing algorithm this purpose eliminates intermediate 3D dubin's poses according to an iterative approach. The created planning path includes some avoidable rolls throughout its navigation towards the destination. The proposed method simplifies the 3D Dubins' waypoints using the smoothing path technique [25]. This algorithm removes in-between 3D Dubins facades according to an iterative approach. Using the smoothing path technique joins non-sequential 3D Dubins facades every other as long as there are no collisions. The smoothed waypoints produced due to this function improves tracking characteristic of fixed-wings simulation models. The simulating of the fixed-wings UAV model by the new smoothed path consists of: i) plotting 3D map; ii) plotting smoothed path according to UAV Dubins joints; and iii) plotting the fixed-wing guide model of the simulated flight path.

3. RESULTS AND DISCUSSION

3.1. Generated 3D random occupancy map

Consider the ground as an obstacle since a UAV must avoid hitting the ground while in flight. As a result, mark the ground plane's (x-y plane) occupancy as 1, indicating that it is an obstruction. The generated 3D random occupancy map is shown in Figure 3.

3.2. Selection of an unoccupied starting and destination poses

This section includes loading the 3D map occupancy and pre-generated obstacles. We assume unknown spaces to be unoccupied and by means of a reference points on the map, we select an unoccupied starting and destination poses as shown in Figure 4. We see it created an object O map for occupancy map 3D class in the workspace. The grid resolution is said to be one cell per meter by default for the map. We have set the threshold for free and occupied cells to be the same so any probability values between 0.65 will be considered as an obstacle free and above that will be considered as an occupied cell. Now we plot this and visualize the start and goal on the occupancy map. The next step is to plan a path between the start and the goal location. So, the state space is a representation of the map in a way that the UAV understands. That means it needs to account for the aerodynamics constraints of the UAV the tree finally crosses the search in the state space to connect the goal states by the start state.

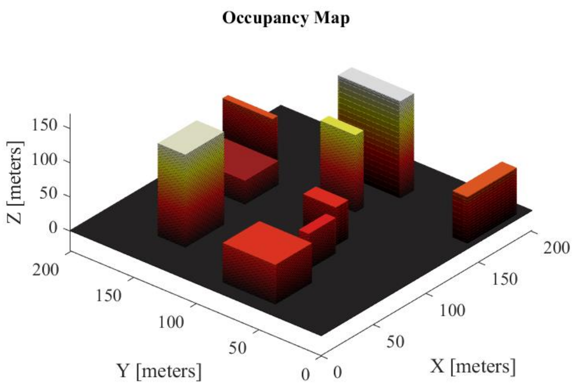

Figure 3. The generated 3D random occupancy map

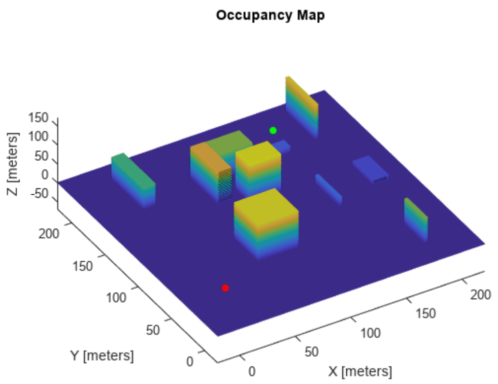

Figure 4. Selection of an unoccupied starting and destination poses

3.3. The produced path planning according to the UAV Dubins joints

The produced path planning according to the UAV Dubins joints is shown in Figure 5.It is observed that the UAV simulation flight was slightly deviated with respect to the produced path planning due to small tracking control error. In addition, the Dubins 3D path algorithm supposes that the changes of the UAV roll angle are instantaneous. However, the real dynamics responses with a slower rate to rolling instructions. Another method to compensate this delay is by planning paths of additional conventional aerodynamic restrictions.

3.4. Smoothing strategy of path planning according to the UAV Dubins joints

The smoothing strategy of path planning according to the UAV Dubins joints is shown in Figure 6. In these simulated results, the flight of a UAV deviates a little from the path planning due to small errors of control tracking and also the 3D Duben's path supposes immediate varies over roll angle of the UAV, other than the real dynamics has a slow reaction to these role instructions. Another method to recompense of these lags is to map the path with more traditional aerodynamic restrictions. Now we have our visualization and simulation working next we would like to smoothen the path. The creative plant pathway builds some superfluous rotates while find the way towards the target. The smoothing strategy the smooth path generated by this process is much shorter and improves tracking properties of the fixed-wing model simulation. The results obtained demonstrates that the smoothed generated waypoints is significantly improved tracking in general with shorter path.

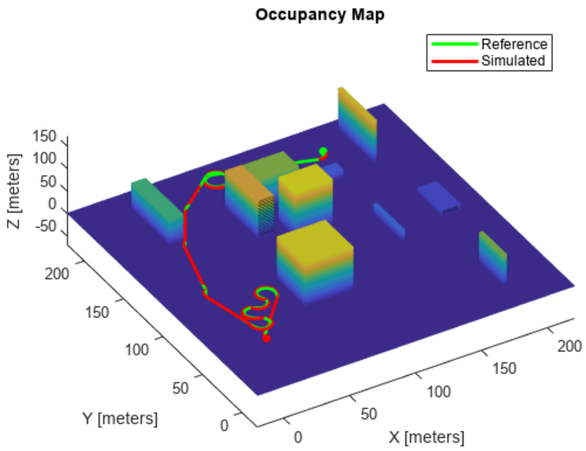

Figure 5. Interpolating the produced path planning according to the UAV Dubins joints

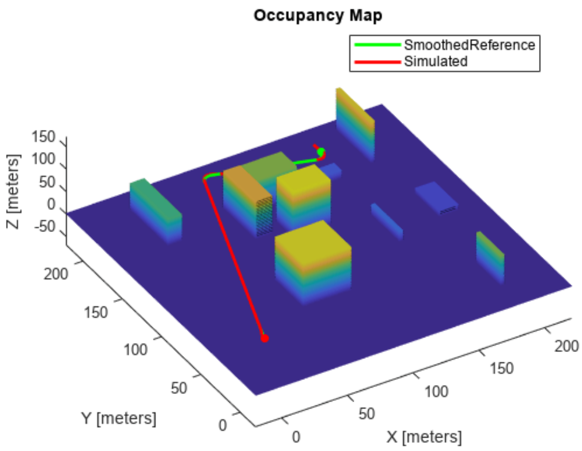

Figure 6. Smoothing strategy of path planning according to the UAV Dubins joints

4. CONCLUSION

This work presented a demonstration of motion planning for an UAV with fixed-wing based on exploring RRT technique for a 3D map known destination and start pose. A UAV with fixed-wing is a naturally nonholonomic and should follow aerodynamic restrictions such as the airspeed, flight path angle, and maximum roll angle when there is a movement between two points on the map. This work concluded as: i) the 3D random occupancy map has been generated using an automatic random positioning with any different dimensions and number of obstacles; ii) the work successfully simulated the fixed-wing UAV model that follows path planning using Dubinspath algorithm. However, it was slightly deviated with respect to the produced path planning due to small tracking control error; and iii) the work simulated UAV trajectory path with smooth Dubinspath to improve tracking characteristics and shorten the path between the start and goal points.

ACKNOWLEDGEMENTS

Author thanks the Department of Robotics and Automation Engineering, Northern Technical University, Mosul, Iraq for their support.

REFERENCES

[1] T. Hinzmann, J. L. Schönberger, M. Pollefeys, and R. Siegwart, "Mapping on the Fly: Real-Time 3D Dense Reconstruction, Digital Surface Map and Incremental Orthomosaic Generation for Unmanned Aerial Vehicles," in *Springer Proceedings in Advanced Robotics*, 2018, pp. 383–396. doi: 10.1007/978-3-319-67361-5_25.
[2] W. Zhang, S. Zhang, F. Wu, and Y. Wang, "Path Planning of UAV Based on Improved Adaptive Grey Wolf Optimization Algorithm," *IEEE Access*, vol. 9, pp. 89400–89411, 2021, doi: 10.1109/ACCESS.2021.3090776.
[3] Y. Song, H. Lee, D. Kang, B. Kim, and M. Park, "A Study on the Determination Methods of Monitoring Point for Inundation Damage in Urban Area Using UAV and Hydrological Modeling," *Water*, vol. 14, no. 7, p. 1117, Mar. 2022, doi: 10.3390/w14071117.
[4] K. W. Lee and J. K. Park, "Comparison of UAV Image and UAV LiDAR for Construction of 3D Geospatial Information," *Sensors and Materials*, vol. 31, no. 10, p. 3327, Oct. 2019, doi: 10.18494/SAM.2019.2466.
[5] O. I. D. Bashi, W. Z. W. Hasan, N. Azis, S. Shafie, and H. Wagatsuma, "Unmanned Aerial Vehicle Quadcopter: A Review," *Journal of Computational and Theoretical Nanoscience*, vol. 14, no. 12, pp. 5663–5675, Dec. 2017, doi: 10.1166/jctn.2017.7049.
[6] S. Niijima, R. Umeyama, Y. Sasaki, and H. Mizoguchi, "City-Scale Grid-Topological Hybrid Maps for Autonomous Mobile Robot Navigation in Urban Area," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Oct. 2020, pp. 2065–2071. doi: 10.1109/IROS45743.2020.9340990.

[7] G. Kumar, A. Patil, R. Patil, S. Park, and Y. Chai, "A LiDAR and IMU Integrated Indoor Navigation System for UAVs and Its Application in Real-Time Pipeline Classification," *Sensors*, vol. 17, no. 6, p. 1268, Jun. 2017, doi: 10.3390/s17061268.

[8] O. I. D. Bashi, W. Z. W. Hasan, N. Azis, S. Shafie, and H. Wagatsuma, "Quadcopter sensing system for risky area," in *2017 IEEE Regional Symposium on Micro and Nanoelectronics (RSM)*, IEEE, Aug. 2017, pp. 216–219. doi: 10.1109/RSM.2017.8069152.

[9] O. I. D. Bashi, W. Z. W. Hasan, N. Azis, S. Shafie, and H. Wagatsuma, "Autonomous quadcopter altitude for measuring risky gases in hazard area," *Journal of Telecommunication, Electronic and Computer Engineering*, vol. 10, no. 2–5, pp. 31–34, 2018, [Online]. Available: https://jtec.utem.edu.my/jtec/article/view/4345

[10] M. V Ramana, S. A. Varma, and M. Kothari, "Motion Planning for a Fixed-Wing UAV in Urban Environments," *IFAC-PapersOnLine*, vol. 49, no. 1, pp. 419–424, 2016, doi: 10.1016/j.ifacol.2016.03.090.

[11] J. M. Levin, M. Nahon, and A. A. Paranjape, "Real-time motion planning with a fixed-wing UAV using an agile maneuver space," *Autonomous Robots*, vol. 43, no. 8, pp. 2111–2130, Dec. 2019, doi: 10.1007/s10514-019-09863-2.

[12] J. M. Levin, A. Paranjape, and M. Nahon, "Agile fixed-wing UAV motion planning with knife-edge maneuvers," in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, Jun. 2017, pp. 114–123. doi: 10.1109/ICUAS.2017.7991475.

[13] Y. Guo, X. Liu, X. Liu, Y. Yang, and W. Zhang, "FC-RRT*: An Improved Path Planning Algorithm for UAV in 3D Complex Environment," *ISPRS International Journal of Geo-Information*, vol. 11, no. 2, p. 112, Feb. 2022, doi: 10.3390/ijgi11020112.

[14] M. R. Al-Obaidi *et al.*, "Efficient Charging Pad for Unmanned Aerial Vehicle Based on Direct Contact," in *2018 IEEE 5th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA)*, IEEE, Nov. 2018, pp. 1–5. doi: 10.1109/ICSIMA.2018.8688767.

[15] A. Saravanakumar, A. Kaviyarasu, and R. Ashly Jasmine, "Sampling based path planning algorithm for UAV collision avoidance," *Sādhanā*, vol. 46, no. 3, p. 112, Sep. 2021, doi: 10.1007/s12046-021-01642-z.

[16] S. P. Jackson, "Controlling Small Fixed Wing UAVs to Optimize Image Quality from On-Board Cameras," 2011. [Online]. Available: https://escholarship.org/content/qt7xj6v7q9/qt7xj6v7q9_noSplash_81c3a9f64142c569ac602486fa90d891.pdf

[17] D. Lee and D. H. Shim, "RRT-based path planning for fixed-wing UAVs with arrival time and approach direction constraints," in *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, May 2014, pp. 317–328. doi: 10.1109/ICUAS.2014.6842270.

[18] R. Ma, W. Ma, X. Chen, and J. Li, "Real-time obstacle avoidance for fixed-wing vehicles in complex environment," in *2016 IEEE Chinese Guidance, Navigation and Control Conference (CGNCC)*, IEEE, Aug. 2016, pp. 498–502. doi: 10.1109/CGNCC.2016.7828835.

[19] H. Huang and A. V Savkin, "Path planning for a solar-powered UAV inspecting mountain sites for safety and rescue," *Energies*, 2021, doi: 10.3390/en14071968.

[20] J. Huang, W. Sun, and Y. Gao, "A Method of Trajectory Planning for Unmanned Aerial Vehicle Formation Based on Fluid Dynamic Model," *IEEE Access*, vol. 8, pp. 2824–2834, 2020, doi: 10.1109/ACCESS.2019.2961632.

[21] M. R. AL-Obaidi, M. A. Mustafa, W. Z. W. Hassan, N. Azis, A. H. Sabry, and M. Z. A. Ab-Kadir, "Improvement in energy conversion for unmanned aerial vehicle charging pad," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 17, no. 2, p. 767, Feb. 2020, doi: 10.11591/ijeecs.v17.i2.pp767-773.

[22] B. Li and B. Chen, "An Adaptive Rapidly-Exploring Random Tree," *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 2, pp. 283–294, Feb. 2022, doi: 10.1109/JAS.2021.1004252.

[23] "Motion Planning with MATLAB - MATLAB & Simulink." Accessed: Sep. 27, 2022. [Online]. Available: https://www.mathworks.com/campaigns/offers/next/getting-started-with-motion-planning-in-matlab-ebook.html#3

[24] "Navigation Toolbox." Accessed: Apr. 24, 2023. [Online]. Available: https://www.mathworks.com/products/navigation.html

[25] R. W. Beard and T. W. McLain, *Small Unmanned Aircraft: Theory and Practice*. 2012. [Online]. Available: https://press.princeton.edu/books/hardcover/9780691149219/small-unmanned-aircraft

BIOGRAPHIES OF AUTHORS

Omar I. Dallal Bashi received the Technical B.Eng. and Technical Master Degrees in Computer Engineering Technology from Technical Engineering Collage of Mosul, Northern Technical University, Iraq in 2009 and 2012 respectively. He received a PhD degree in autonomous flight algorithm of a quadcopter sensing system for methan gas concentration measurements at landfill site from the Department of Computer and Embedded Systems Engineering, Robotics and Automation Engineering from University Putra Malaysia (UPM) in 2018. He is currently a lecturer and deputy dean for scientific and student affairs, Agricultural Technical College/Northern Technical University, Iraq. His research interests include drones, IoT, robotics and automation. He can be contacted at email: omardallalbashi@ntu.edu.iq.

Yasir Mahmood Al Kubaisi was born in Baghdad, Iraq. He received the B.Sc. and M.Sc. degrees in electrical and computer, control and automation, and engineering from the University of Technology-Baghdad, Iraq, in 1994 and 2005, respectively, and a PhD degree in DC-based PV-powered home energy system from the Department of Electrical and Electronic Engineering, Control and Automation, UPM, Malaysia, in 2020. He is currently an Employ at Health, Safety and Sustainable Department, Dubai Academic Health Corporation, UAE. He is the author of more than 8 articles, more than one invention, and holds one patent. His research interests include integrated solar-powered smart home systems, dc distribution, and robotic systems. He can be contacted at email: yaser.19711@gmail.com.

Husamuldeen K. Hameed received the Bachelor and Master Degrees in Electrical and Electronic Engineering from the University of Technology/Iraq in 1994 and 2002 respectively. He received a PhD degree in signal processing from the University Putra Malaysia (UPM) in 2020. He is currently a lecturer in the High Institute of Telecommunications and Post/Iraq. His research interests include digital signal processing, biomedical engineering, telecommunications, robotics and automation. He can be contacted at email: husamuldeen72@gmail.com.

Ahmad H. Sabry was born in Baghdad, Iraq. He received the B.Sc. and M.Sc. degrees in electrical and electronics, control and automation, engineering from the University of Technology-Baghdad, Iraq, in 1994 and 2001, respectively. He received the Ph.D. degree in DC-based PV-powered home energy system from Department of Electrical and Electronic Engineering, Control and Automation at UPM, Malaysia in 2017. He is the author of more than 45 articles, and more than 5 inventions, and holds one patent. His research interests include integrated solar powered smart home system based on voltage matching, DC distribution, industrial robotic systems, wireless energy management systems. He is a reviewer in more than seven ISI journals. He can be contacted at email: ahs4771384@gmail.com.