Study of Motion Planning of Quadcopter Under Uncertain Environment

Khitam Mohammed
Computer Engineering Department
University of Basrah
Basrah, Iraq
engpg.khitam.mohamed@uobasrah.edu.iq

Alaa Al-Ibadi
Computer Engineering Department
University of Basrah
Basrah, Iraq
alaa.abdulhassan@uobasrah.edu.iq

Ali Aliedani
Computer Engineering Department
University of Basrah
Basrah, Iraq
ali.nabeel@uobasrah.edu.iq

***Abstract*: The consequences of drone movement state (motion planning), such as position, orientation and speed, during the travelling mission to the destination are crucial factors in its safety, resource consumption and task accomplishment. The unmanned aerial vehicle can be obliged in an unknown environment to follow an unplanned trajectory when faced with obstacles. Hence, this paper suggested two motion planning approaches for quadcopter navigation implemented in two scenarios under uncertain environments with incomplete information about the investigated environment. In the first scenario, the quadcopter's motion will be controlled using the PID control algorithm to accomplish multiple goals in an environment artificially created to be full of obstacles. The pure pursuit control algorithm, employed for the same purpose in the same uncertain environment, is included in the second scenario. In both cases, the quadcopter will navigate and avoid obstacles using a vector field histogram plus algorithm and a LiDAR sensor in an obstacle avoidance system. We will compare the two tests regarding the time and distance required to evaluate the suggested scenarios' performance in different environmental conditions and show the advantages of utilising such strategies. MATLAB Simulink was used to conduct the simulation.**

Keywords: Quadcopter; Motion planning; Pure pursuit; VHF+; Uncertainty; Unmanned aerial vehicles (UAVs); MATLAB; LiDAR; PID; Motion planning.

I. INTRODUCTION

Unmanned aerial vehicles (UAVs) have lately been given increasing consideration in the robotics community. The quadcopter, also known as a quadrotor or drone, is a special kind of UAV that has vertical takeoff and landing (VTOL) capability [1]. It has the benefit of maneuverability caused by its inherent dynamic [2]. Additionally, UAVs' small size, mobility, agility, and flexibility make them particularly appropriate for situations such as aerial photography, surveillance, industry, search, rescue, agriculture, inspection, and more. Aerial robots must have the ability to navigate on their own to complete jobs in harsh situations that may be hazardous or unavailable for human operators [3]. This has been accomplished mostly as a result of very active robotics and autonomous technology research and development. In order for a quadcopter to travel effectively and dependably in an area without any human aid, there are still numerous difficulties to overcome. UAVs should be able to collect essential data from the environment and take the appropriate actions to design a workable route for a collision-free movement to achieve its target. Motion planning is one of the most important challenges faced by quadcopter to achieve a successful flight and accomplish a specific mission [4].

II. RELATED WORK

A variety of strategies have been suggested for UAV motion planning implementation. These methods are based on several factors, including the robot's capabilities, type of sensors, environment, and algorithms. They aim to progressively improve performance in terms of speed, distance, safety, cost, smoothness, and complexity [5]. Additionally, sensing, mapping, and re-planning are other UAV planning strategies that have been discussed in the literature for operation under unpredictable environments. Shim et al. [6] introduced an exploration technique for a UAV copter equipped with a laser scanner to create obstacle maps, utilizing MPC-based obstacle avoidance. Sinopoli et al. [7] used stereo vision to enable autonomous UAV navigation in partially understood surroundings. Offline computing leverages a priori knowledge of the environment to generate an initial estimate of the optimal path, employing Dijkstra's algorithm and the wavelet transformation of the map. Bio-inspired algorithms have been studied by H. T. Nguyen and others to offer enhanced path planning with reduced run times. Many researchers have adopted hybrid algorithms to determine the most efficient route, which in turn minimizes costs and convergence times [8]. D. A. Ramadan [9] introduced a technique for path planning that employs a two dimensional path-planning algorithm designed for mobile robot route planning. This method yielded favorable outcomes in terms of arrival time and path length. The binary tree vision algorithm, presented by A. T. Rashid [10], is a technique for mobile robot path planning in scenarios with both local and global knowledge of obstacles. Their discussion encompasses both low- and high-level path planning.

In this study, two motion planning approaches were introduced and evaluated for a quadcopter operating in an environment that has incomplete information about the field. The first approach involved navigating a quadcopter around an environment to perform various tasks over obstacles using the VFH+ algorithm paired with an enhanced PID controller. The second scenario utilized the VFH+ algorithm with a pure pursuit controller .

The objective of this study is to ascertain the most efficient path for a robot in terms of time, length, and smoothness, considering both obstacle fields and robot motion limitations.

III. CONTROL SYSTEM

A. PID controller

PID is a typical control technique for quadcopters that is also utilized in industrial control systems. Researchers controlled the attitude and position of the quadcopter using this controller, showing that it is effective at low speeds with little aerodynamic disturbances, such as indoor missions [11].

Reference [12] used the controller for the quadcopter's position control system. Other researchers use the same controller to fully control their quadcopters [13]. They used PID to execute attitude control and trajectory detection based on isolated pitch and yaw motion. Low speeds allow this controller to work inside. The authors addressed the issue of attitude stabilization [14, 15]. Without linearizing gyroscopic effects, they were successful in establishing asymptotic stability. Altitude and attitude proportional-derivative (PD) controllers are developed using a simpler dynamic model [16].

The controller that is being employed divides the pathways between waypoints into segments of constant acceleration or constant velocity and determines an "error" value by computing the contrast between a fixed point and a wanted set point during the process. The controller attempts to get to the specified point by downloading the error's smallest value.

This control algorithm, whose range of applications includes the following:

$$\upsilon(\tau) = K_p e(t) + K_i \int_0^t e(t)d(v) + K_d \frac{de(t)}{d(t)} \quad (1)$$

Where:

K_p coefficient of proportion, K_i coefficient of integral

and,K_d coefficient of the derivative.

Three different mathematical procedures are used to create the control output, which is then produced by summing. Effects on the system are as follows:

- Proportional Effect (P). determines the present error.
- Integral Effect (I): the integral effect is the total of the system's past mistakes.
- Derivative Effect (D): determines the future error.

The Traditional PID controller is shown in Fig.1.

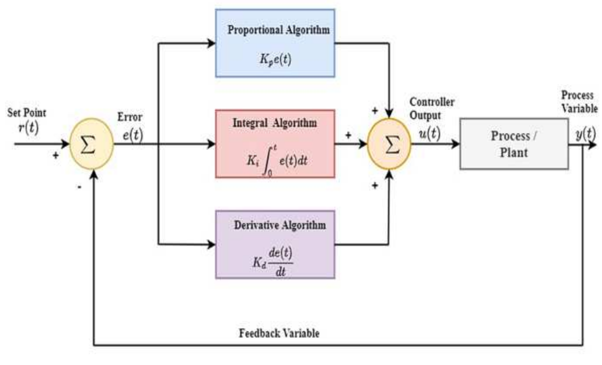

Fig. 1. Traditional PID controller [17].

B. Pure pursuit algorithm

One of the primary tracking methods involves using the pure pursuit algorithm (PPA), which guides the quadcopter throughout various tasks [18]. PPA entails calculating the path's curvature and directing the vehicle toward an area on the path that is a certain distance from the present position. This method is helpful for traveling through situations with complicated pathways, including urban areas or off-road terrain. It is also adaptable to various vehicle and sensor types and reasonably easy to apply. Using sensor data or a pre-made map, the PPA determines the curvature of the route in front of the robot, as shown in Fig. 2[19].

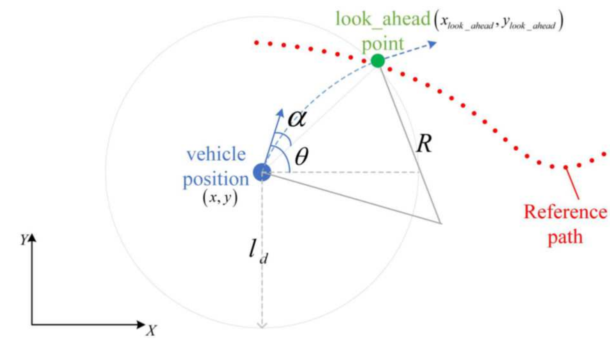

Fig. 2. Diagram illustrating the pure pursuit algorithm [19].

A straight line connects the look-ahead point to the vehicle's (rear wheel) position to determine the steering angle output. The angle created by this line and the drone body is set to this angle termed the look-ahead distance angle is denoted as (2) [20].

$$\alpha = \left|\theta - arctan\left(\frac{y_{look\_ahead} - y}{x_{look\_ahead} - x}\right)\right| \quad (2)$$

Where x_{look_ahead} and y_{look_ahead} pinpoint the position of the look-ahead point corresponding to the vehicle position, x and y define the vehicle's position, and θ represents the vehicle's direction.

Equation (3) illustrates the curvature radius R that the vehicle must follow, derived from geometric considerations.

$$R = \frac{l_d}{2\, sin\alpha} \quad (3)$$

Where l_d represents the look-ahead distance of the quadcopter. The quadcopter's steering angle, as presented in Equation (3), is illustrated in Fig. 3 and is based on the Ackerman geometric model.

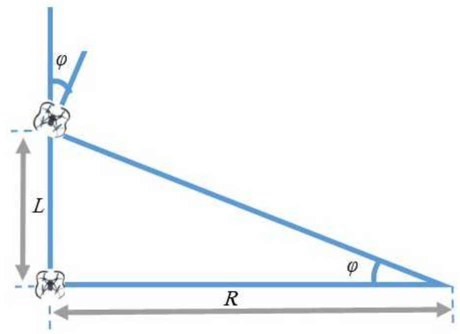

Fig. 3. Ackerman geometric model.

$$\varphi = arctan\left(\frac{L}{R}\right) \quad (4)$$

Here, L symbolizes the vehicle's wheelbase. The robot's steering angle can be derived by transforming Equation (3) into Equation (4).

$$\varphi = arctan\left(\frac{2L.sin\alpha}{l_d}\right) \quad (5)$$

The Ackerman geometric model refers to a steering linkage configuration used in cars and other vehicles. It addresses the challenge of wheels needing to follow circles of varying radii on the inside and outside of turns [21].

IV. Obstacle Avoidance System

The Obstacle Avoidance System (OAS) aims to set off a warning if the quadcopter is about to collide with an obstruction. For identification and tracking of UAVs near obstacles like tall buildings, power line crossings, telecom towers, or wind turbines, OAS employs low power sensors. In this work, we used LiDAR sensor with vector field histogram algorithm (VFH+) in our Obstacle avoidance system. Data from the sensors and other algorithms must be merged into one system in order to process sensor input data to create steering angle movements in the quadcopter [22]. A LiDAR sensor included into the quadcopter enables the identification of environmental items that were not included in the initial 3D model. Prior to following the planned course, the UAV gathers samples of its surroundings to look for impediments. If any are found, it records its location for a certain period of time. If a new barrier is found, a suggested control algorithm, PID in the first scenario and pure pursuit in the second scenario, is carried out to estimate its future position by fitting the sensor values. Additionally, a security distance is determined based on the object's size and speed. The control system is informed of the location of this new obstacle, and it recomputes the new trajectory profile taking into account its presence and utilizing the planned trajectory as a starting point for the iterative procedure. Once the UAV reaches the last waypoint on the trajectory, the procedure is complete.

A. Vector Field Histogram plus

The Vector field histogram (VFH) algorithm was created by Borenstein and Koren initially for real-time local obstacle avoidance with mobile robots in 1991. The VFH+ oobstacle avoidance algorithm is an enhanced version of VFH that operates on a similar principle as the original VFH algorithm. This enhanced algorithm takes into consideration the dimensions of the robot. A map grid of the immediate area, known as the histogram grid and based on the earlier certainty grid and occupancy grid approaches, serves as the input to this algorithm[23].

The new direction of motion is computed using the VFH+ technique through a four-stage data reduction procedure. These stages are the primary polar histogram, binary polar histogram, masked polar histogram, and steering direction selection [24].

V. Proposed Scenarios And Simulation Result

Two scenarios were proposed and tested to control the movement of the aircraft during flight along a predetermined path within an environment filled with obstacles.

The suggested approaches have been investigated under various conditions using the MATLAB simulation tool. Initially, a set of waypoints is defined with missing information about existing obstacles, as shown in Fig. 1. The quadcopter base is depicted in green, while the quadcopter is displayed in blue. A single LiDAR sensor with specified attributes was mounted on the quadcopter system for obstacle detection.

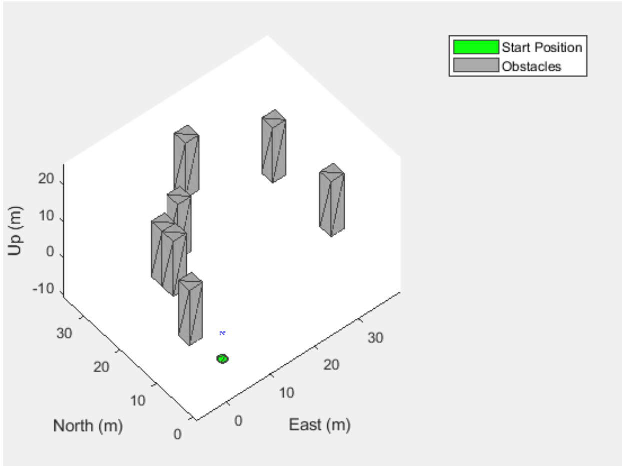

Fig. 4. The scenario environment with obstacles and a quadcopter at the start position.

A. First scenario using PID approach

Based on the current quadcopter state and set of waypoints, the obstacle avoidance system determines the required position and yaw clear of obstacles. The next waypoint's direction is computed as the look-ahead point for the quadcopter. This obtained look-ahead point is updated using the 3D VFH+ method to determine an obstacle-free direction and yaw for collision-free navigation. The desired location is derived by employing the look-ahead distance-constant block, with its value being multiplied by the unit vector in the desired direction and added to the current quadcopter position. The "Controller and Plant" subsystem uses the look-ahead point to generate control instructions and update the quadcopter's status. The roll, pitch, yaw, and thrust control instructions are formulated by the controller to guide the quadcopter in the intended direction. In this first scenario, position control is executed through several PID loops. Using these control commands, the quadcopter plant updates the UAV status. The subsystem handles data and coordinates transformations, extracting the location and orientation from the quadcopter state for display. When using longer look-ahead distances, the quadcopter flies faster, but the risk of collision increases. Conversely, shorter distances result in slower flight but reduced collision likelihood. With a LiDAR sensor and the VFH algorithm combined with PID controls, the quadcopter navigates from its starting point, passing through waypoints while dodging obstacles.

The flowchart detailing the motion of the quadcopter from start to goal in an obstacle-laden environment under uncertain conditions is displayed in Fig. 5. The second scenario is akin to the first, with the distinction being the use of a PPA in place of the PID controller.

The simulation results presented in Table I reveal a minimal position error between the actual and desired paths. Furthermore, there is a narrow gap between the obstacle and the real path, making this method suitable for applications demanding comprehensive area coverage. Table II displays the results achieved using this approach, highlighting the position error between the actual and desired paths at each waypoint.

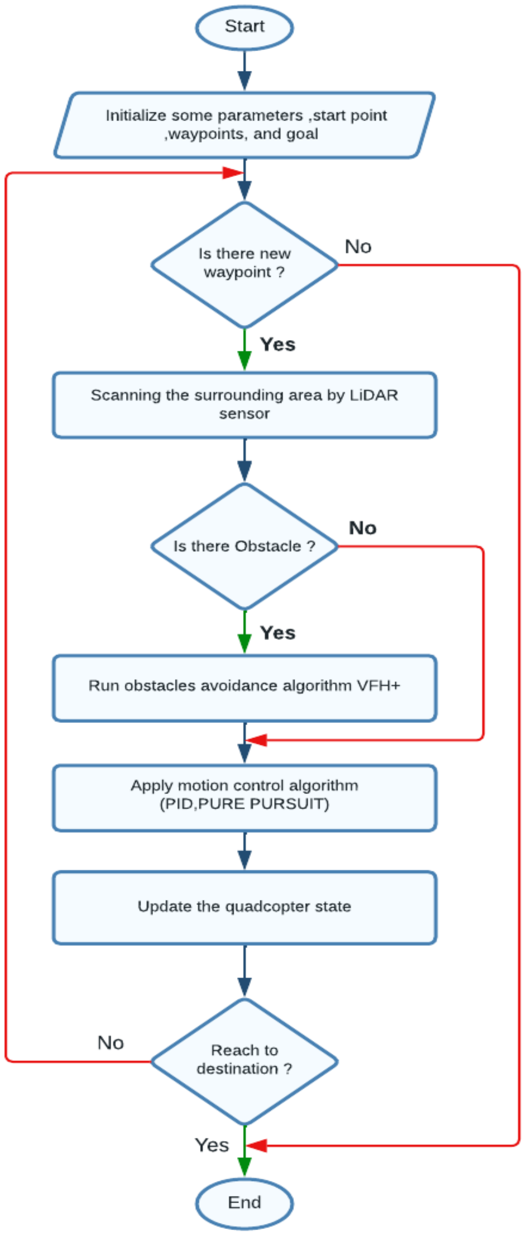

Fig. 5. Flowchart of the steps of suggested scenario.

The simulation using this methodology are shown in Fig.6, Fig.7, Fig.8 and Fig.9.

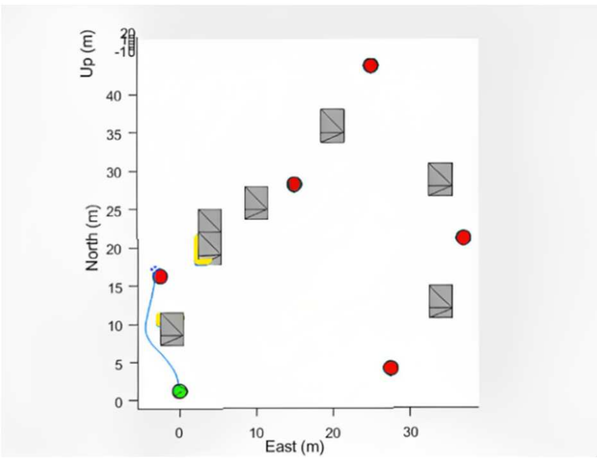

Fig. 6. Quadcopter pass the first obstacle and reach the first waypoint during the navigation using PID approach.

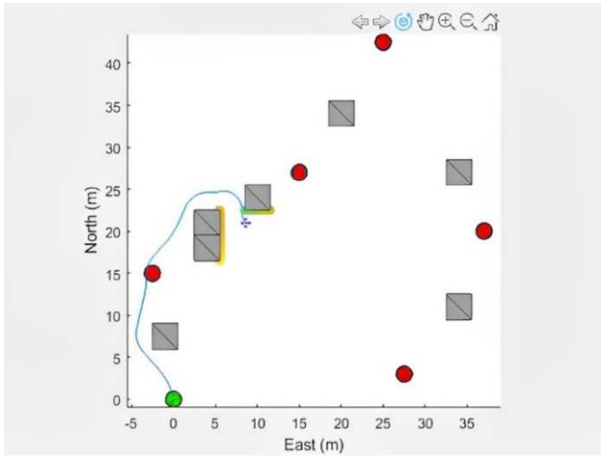

Fig. 7. Quadcopter facing obstacle during the navigation from waypoint to the next point.

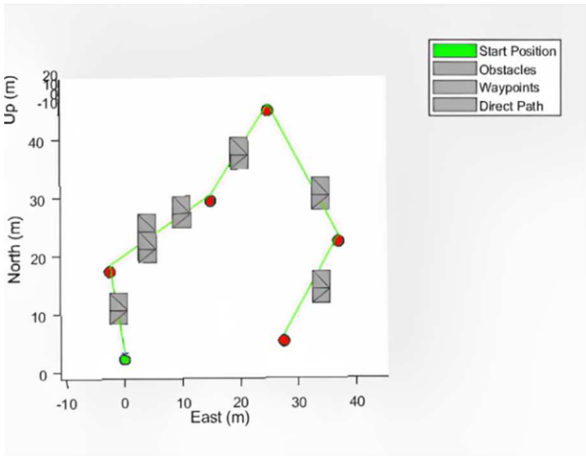

Fig. 8. The shortest direct path from the starting point to the target, passing through the waypoints.

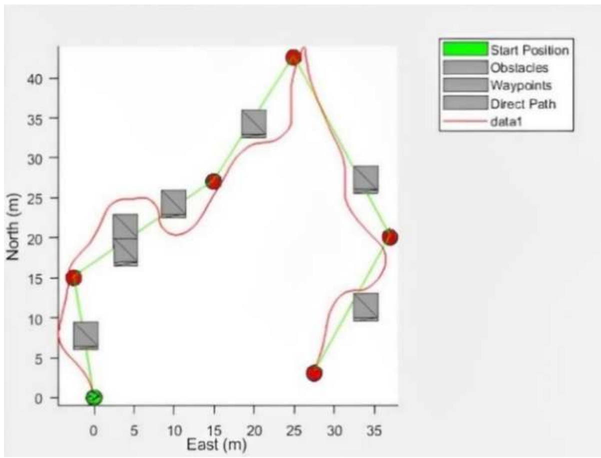

Fig. 9. Shows the direct path and the actual path at the end of simulation using PID approach.

TABLE I. SHOWS THE POSITION ERROR BETWEEN THE ACTUAL PATH AND THE SHORTEST PATH DURING FACING THE FIRST THREE OBSTACLES.

waypoint	Error in meter		
	Position x error	Position y error	Distance to the obstacle
1st	2.8	0	1.3
2nd	0	2	0.5
3rd	0	1.9	0.4

TABLE I. SHOWS THE ERROR BETWEEN THE ACTUAL PATH AND THE SHORTEST PATH AT EACH WAYPOINTS USING PID APPROACH.

Waypoint	Error in meter	
	Position x error	Position y error
1st	0	0.01
2nd	0.05	0.02
3rd	0	0.04
4th	0.08	0.18
5th	0	0

Simulation results as in Table I show that the position error between the actual path and the desired path is very small, Additionally, there is not much space between the obstacle and the actual path. This makes the approach useful for a variety of applications that call for thorough coverage of the region.

Table II shows the results obtained using this method related to the position error between the actual path and the desired path at each waypoints of the path..

A. Second scenario using the Pure Pursuit algorithm

An engineering technique called the Pure Pursuit algorithm works by figuring out the curvature the vehicle will take to reach the target path point [14]. The curvature is generated connecting the vehicle's current position to the next point on the desired track. The PPA works more effectively whenever the deviation of the drone's direction to the next point is smaller than the threshold angle. In this case, the UVA should shift slowly toward the next point, then the arc r should be calculated. If the deviation is greater than the threshold angle in this case, the drone must stop at the current position and turn its direction toward the next point, then the arc r must be calculated. The last case is considered one of the disadvantages of this method. The second scenario was implemented using the pure pursuit controller with VFH+ algorithm. As in the simulation in the first scenario, the percentage of error in trajectory tracking was measured at predetermined trajectory points in addition to measuring the distance from the obstacle when the aircraft encountered an obstacle during its trajectory flight to compare the two methods.

The second scenario was implemented using the pure pursuit controller with VFH+ algorithm. As in the simulation in the first scenario, the percentage of error in trajectory tracking was measured at predetermined trajectory points in addition to measuring the distance from the obstacle when the aircraft encountered an obstacle during its trajectory flight to compare the two methods.

The simulation results show the position error between the actual path and the planned path to the same extent as in Table I and Table II. Compared to the first technique, the proportion of departure from the planned path is higher. Additionally, this approach requires less time to do the necessary work than the first technique using PID, making it beneficial in applications that need to finish the task in the shortest period of time feasible.

Simulation results using this method are shown in Fig.10 and Fig.11.

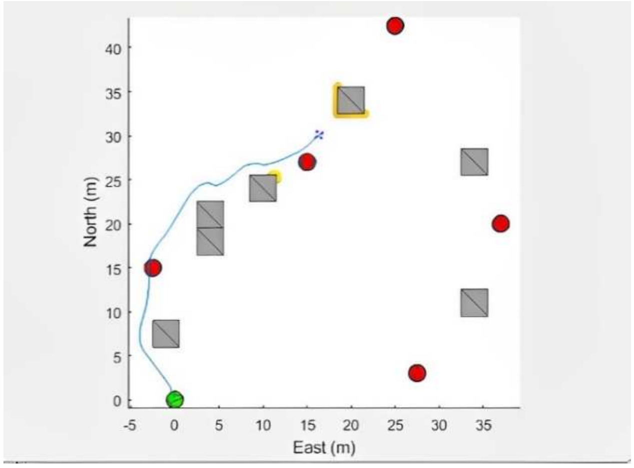

Fig. 10. Quadcopter facing obstacle during the navigation from waypoint to the next point.

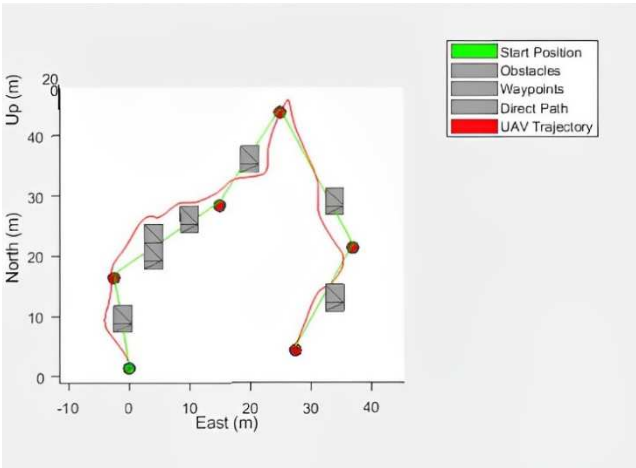

Fig. 11. Shows the direct path and the actual path at the end of the simulation using pure pursuit approach.

Table I below shows that the distance from the obstacles is nearly constant since this approach moves the quadcopter away from the obstacle at a predetermined safety distance, which is helpful for ensuring the quadcopter's safety.

Table II shows the results obtained using PPA with VFH+ related to the position error between the actual path and the desired path at each waypoint of the path.

TABLE I. SHOWS ERROR BETWEEN THE ACTUAL PATH AND THE SHORTEST PATH WHILE FACING THE FIRST THREE OBSTACLES USING THE PURE PURSUIT APPROACH.

Waypoint	Error in meter		
	Position x error	Position y error	Distance to the obstacle
1st	4	1.5	2.5
2nd	4	1.5	2.5
3rd	4	1.5	2.5

TABLE II. SHOWS THE ERROR BETWEEN THE ACTUAL PATH AND THE SHORTEST PATH AT EACH WAYPOINT.

Waypoint	Error in meter	
	Position x error	Position y error
1st	0	0.04
2nd	0.05	0
3rd	0	0.04
4th	0.08	0
5th	0	0

VI. CONCLUSION

In this study, a powerful motion planning for quadcopters was studied using various simulation scenarios. PPA and PID were the two control algorithms used for the position control of the quadcopter. An improved VFH+ algorithm was used to implement the quadcopter system's motion as it traveled to the intended destination position from a predetermined start location. The quadcopter successfully avoided colliding with an obstacle. The efficiency of the obstacle avoidance algorithm was further evaluated by adding more obstacles to the surrounding area. The quadcopter system successfully navigated the surroundings. The implementation was carried out several times, and we concluded from the repeated implementation that the time taken by the second method was shorter than the first method, but the percentage of deviation from the required path was greater than the first method. The second method was also superior in terms of path length and path smoothness when compared to the first method, due to the use of the PPA.

REFERENCES

[1] M. Pouzesh and S. Mobayen, "Event-triggered fractional-order sliding mode control technique for stabilization of disturbed quadcopter unmanned aerial vehicles," Aerospace Science and Technology, vol. 121, p. 107337, 2022.

[2] K. Kogiso and M. Miyamoto, "Development and Examination of Secure Quadcopter Control System with Partially Homomorphic Encryption," in 2023 IEEE/SICE International Symposium on System Integration (SII), 2023: IEEE, pp. 1-6.

[3] A. Kapnopoulos and A. Alexandridis, "A cooperative particle swarm optimization approach for tuning an MPC-based quadcopter trajectory tracking scheme," Aerospace Science and Technology, vol. 127, p. 107725, 2022.

[4] Y. Gan, "Search-based motion planning algorithms for quadcopters," in International Conference on Artificial Intelligence and Industrial Design (AIID 2022), 2023, vol. 12612: SPIE, pp. 263-273.

[5] F. Noroozi, M. Daneshmand, and P. Fiorini, "Conventional, Heuristic and Learning-Based Robot Motion Planning: Reviewing Frameworks of Current Practical Significance," Machines, vol. 11, no. 7, p. 722, 2023.

[6] D. H. Shim, H. Chung, and S. S. Sastry, "Conflict-free navigation in unknown urban environments," IEEE robotics & automation magazine, vol. 13, no. 3, pp. 27-33, 2006.

[7] B. Sinopoli, M. Micheli, G. Donato, and T.-J. Koo, "Vision based navigation for an unmanned aerial vehicle," in Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164), 2001, vol. 2: IEEE, pp. 1757-1764.

[8] H. T. Nguyen, T. V. Quyen, C. V. Nguyen, A. M. Le, H. T. Tran, and M. T. Nguyen, "Control algorithms for UAVs: A comprehensive survey," EAI Endorsed Transactions on Industrial Networks and Intelligent Systems, vol. 7, no. 23, pp. e5-e5, 2020.

[9] D. A. Ramadhan, A. T. Rashid, and O. T. Rashid, "Two Dimensional Path Planning with Static Polygon Obstacles Avoidance," Iraqi Journal for Electrical And Electronic Engineering, no. 3RD, 2020.

[10] A. T. Rashid, A. A. Ali, M. Frasca, and L. Fortuna, "Path planning with obstacle avoidance based on visibility binary tree algorithm," Robotics and Autonomous Systems, vol. 61, no. 12, pp. 1440-1449, 2013.

[11] G. Hoffmann, H. Huang, S. Waslander, and C. Tomlin, "Quadcopter helicopter flight dynamics and control: Theory and experiment," in AIAA guidance, navigation and control conference and exhibit, 2007, p. 6461.

[12] M. K. Joyo, S. F. Ahmed, D. Hazry, M. H. Tanveer, and F. A. Warsi, "Position controller design for quad-rotor under perturbed condition," Wulfenia Journal, vol. 20, no. 7, pp. 178-189, 2013.

[13] J. H. Gillula, H. Huang, M. P. Vitus, and C. J. Tomlin, "Design of guaranteed safe maneuvers using reachable sets: Autonomous quadcopter aerobatics in theory and practice," in 2010 IEEE international conference on robotics and automation, 2010: IEEE, pp. 1649-1654.

[14] A. Tayebi and S. McGilvray, "Attitude stabilization of a four-rotor aerial robot," in 2004 43rd IEEE conference on decision and control (CDC)(IEEE Cat. No. 04CH37601), 2004, vol. 2: Ieee, pp. 1216-1221.

[15] A. Tayebi and S. McGilvray, "Attitude stabilization of a VTOL quadcopter aircraft," IEEE Transactions on control systems technology, vol. 14, no. 3, pp. 562-571, 2006.

[16] C. D. Krainer, JNavigator: an autonomous navigation system for the JAviator quadcopter helicopter. na, 2009.

[17] R. P. Borase, D. Maghade, S. Sondkar, and S. Pawar, "A review of PID control, tuning methods and applications," International Journal of Dynamics and Control, vol. 9, pp. 818-827, 2021.

[18] J. Morales, J. L. Martínez, M. A. Martínez, and A. Mandow, "Pure-pursuit reactive path tracking for nonholonomic mobile robots with a 2D laser scanner," EURASIP Journal on Advances in signal Processing, vol. 2009, pp. 1-10, 2009.

[19] R. Wang, Y. Li, J. Fan, T. Wang, and X. Chen, "A novel pure pursuit algorithm for autonomous vehicles based on salp swarm algorithm and velocity controller," IEEE Access, vol. 8, pp. 166525-166540, 2020.

[20] K. Lee, S. Jeon, H. Kim, and D. Kum, "Optimal path tracking control of autonomous vehicle: Adaptive full-state linear quadratic Gaussian (LQG) control," IEEE Access, vol. 7, pp. 109120-109133, 2019.

[21] P. A. Simionescu and D. Beale, "Optimum synthesis of the four-bar function generator in its symmetric embodiment: the Ackermann steering linkage," Mechanism and Machine Theory, vol. 37, no. 12, pp. 1487-1504, 2002.

[22] I. P. Sary, Y. P. Nugraha, M. Megayanti, E. Hidayat, and B. R. Trilaksono, "Design of obstacle avoidance system on hexacopter using vector field histogram-plus," in 2018 IEEE 8th International Conference on System Engineering and Technology (ICSET), 2018: IEEE, pp. 18-23.

[23] J. Borenstein, "Vfh+: Reliable obstacle avoidance for fast mobile robots," in Proc. IEEE Int. Conf. on Robotics and Automation, Leuven, 1998, 1998.

[24] S. Vanneste, B. Bellekens, and M. Weyn, "3DVFH+: Real-time three-dimensional obstacle avoidance using an Octomap," in MORSE 2014 Model-Driven Robot Software Engineering: proceedings of the 1st International Workshop on Model-Driven Robot Software Engineering co-located with International Conference on Software Technologies: Applications and Foundations (STAF 2014), York, UK, July 21, 2014/Assmann, Uwe [edit.], 2014, no. 1319, pp. 91-102.