

# Feasible Computationally Efficient Path Planning for UAV Collision Avoidance

Han Wang, Muqing Cao, Hao Jiang and Lihua Xie

**Abstract**—This paper presents a robust computationally efficient real-time collision avoidance algorithm for Unmanned Aerial Vehicle (UAV), namely Memory-based Wall Following-Artificial Potential Field (MWF-APF) method. The new algorithm switches between Wall-Following Method (WFM) and Artificial Potential Field method (APF) with improved situation awareness capability. Historical trajectory is taken into account to avoid repetitive wrong decision. Furthermore, it can be effectively applied to platform with low computing capability. As an example, a quad-rotor equipped with limited number of Time-of-Flight (TOF) rangefinders is adopted to validate the effectiveness and efficiency of this algorithm. Both software simulation and physical flight test have been conducted to demonstrate the capability of the MWF-APF method in complex scenarios.

## I. INTRODUCTION

Unmanned Aerial Vehicle (UAV) is one of the most promising technologies in the 21st century. Thanks to its low cost and high mobility, UAV has the potential to replace human in jobs such as package delivery, forest rescue and so forth. Moreover, heavy lift drones have been employed as an innovative alternative of transportation. However, with the popularization of UAV, there are increasing reports of accidental vehicle crash of UAV due to poor pilot control. Safety concern is real and it becomes the bottleneck of UAV industry. Collision avoidance is the key for a successful navigation mission. In recent years, researchers have been working on collision detection and avoidance system (CDAS) that can be implemented on UAVs [1]–[3]. A good collision avoidance system is the key to further development on UAV such as swarm system [4].

A comprehensive collision avoidance solution includes two parts, namely obstacle detection and obstacle avoidance. Obstacle detection studies different approaches to detect the surrounding obstacles. Accuracy of the generated obstacle profile is the most crucial factor of obstacle detection. It has significant impact on the later part of obstacle avoidance. However, there is no uniform obstacle detection solution at the current stage. In the past decades, researchers are putting effort on Obstacle Avoidance system with various sensing units such as infrared sensor [1], vision sensor [2] and radar sensor [3]. Every method has its own advantages and

disadvantages [5]. Vision-based obstacle detection system is of great interest to research community due to the richness of environment information that can be extracted from digital images. However, the performance of such system is limited by the processing power of the on-board computer equipped by UAV. In contrast, a rangefinder such as infrared sensor, does not require high processing power and is able to provide very accurate feedback in terms of distance. The constraint of a rangefinder is the narrow field of view (FOV). Each rangefinder is able to cover 1 to 2 degree FOV only. As for radar-based system, it has much wider FOV. However, there is a higher chance for false detection to occur as the outline of obstacle cannot be well perceived.

Obstacle avoidance is the task of navigating robot subject to non-collision constraints. Different motion planning methods are used to achieve the task. One of the most classic motion planning methods is called artificial potential field (APF) method [6], which uses the idea of imaginary force. Previous study [7] has shown that APF can provide a computationally efficient motion planning for robot. However, one problem generated by APF is the Local Minimum Problem illustrated in Fig. 1. The robot may be trapped at local minimum points where the imaginary potential force equals to zero and unable to reach the target [7].

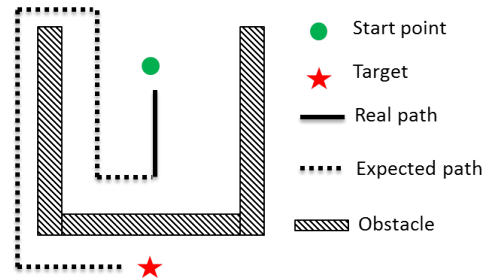


Fig. 1. Local Minimum of APF

Another popular motion planning method is called Bug Algorithm. It was firstly introduced by Lumelsky and Stepanov [8]. This method adopts the strategy to follow the path specified by the contour of an obstacle. Similar to APF, it is an effective and efficient solution for the obstacle avoidance. Research on the Bug Algorithm has made rapid progress in recent years and evolved into methods such as I-Bug Algorithm [9], Point Bug Algorithm [10], Insert Bug Algorithm [11], Vis Bug Algorithm [12] and Dist Bug Algorithm [13]. All these algorithms are categorized as Bug Family Algorithm (BFA). The Bug Family Algorithm usually has following three assumptions: i) the robot is a point, ii)

The work is partially supported by ST Engineering - NTU Corporate Lab under the NRF Corporate Lab @ University Scheme.

H. Wang, M. Cao and L. Xie are with the School of Electrical and Electronic Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore (e-mail: {wang.han, mqcao, elhxie}@ntu.edu.sg).

H. Jiang is with College of Electrical Engineering and Automation, Fuzhou University, Fuzhou, China (e-mail: jiangh@fzu.edu.cn) and was with the School of Electrical and Electronic Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore.

it has a perfect localization, and iii) its sensors are precise. Due to these assumptions, research on Bug Algorithm mainly focuses on theoretical studies rather than real life implementation. In the real application, we use the wall-following concept more often than bug family algorithm [14], [15]. Wall-Following Method (WFM) is developed based on the key idea of the bug family algorithm with the consideration of real life constraints such as robot size. WFM only requires to estimate the normal and tangent of the obstacle surface from real-time sensor input. UAVs with WFM always keep a safety normal distance from the obstacle and navigate in the tangent direction only. Feasible implementation makes WFM popular in robotics application. However, it is usually used in simple environment. One significant disadvantage of WFM is that it can lead to endless loop or repetitive trajectory in some complex cases shown in Fig. 2. If the robot always follows the right direction of the wall tangent when there are obstacles detected, it can never reach the target. It is described above that both methods have specific

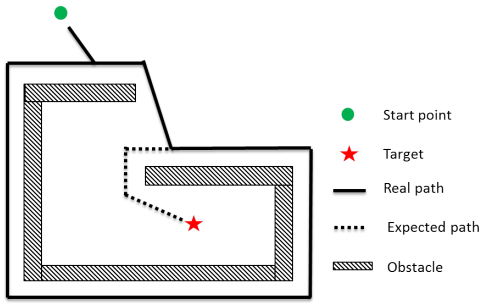


Fig. 2. Endless Loop Scenario for Wall-Following Method.

failure scenario. However, it is unlikely for both methods to fail at the same time. Hence, researchers have been looking for a method combining both WFM and APF features. These two methods will compensate each other once equipped with appropriate situation awareness. The failure of one algorithm will trigger another algorithm to take over path planning task. The research on this combined method has proven that it can result in better path than individual method [16]–[18].

## II. RELATED WORKS THAT INVOLVES COMBINED METHODS

As mentioned above, combined WFM and APF method aims to allow each stand-alone method to compensate another. The key point of this method is the situation awareness to make a right choice between two stand-alone methods. A proper situation awareness is able to recognize accurately when either WFM or APF has reached its failure mode. As a result, a better performance can be achieved.

### A. Early stage of situation awareness

The idea of combining WFM and APF was initially introduced by [18]. This work applied a simple situation awareness for robot to switch between two modes, namely VFF (Virtual Force Field, an extension from APF) and WFM.

To switch from VFF to WFM, the following logic was used to distinguish local minimum condition:

$$\|F_{total}\| \leq F_{th}, \quad (1)$$

where  $F_{total}$  is the total artificial potential force calculated at the current robot position,  $F_{th}$  is the tolerance for a point to be considered as a local minimum point. The condition checks if the robot reaches the local minimum point, that is, the failure of APF happens. If the condition occurs, the system will switch to WFM to escape from the local minimum problem. To switch to VFF from WFM the following condition was used:

$$|\theta_{goal} - \theta_0| > \frac{\pi}{2}, \quad (2)$$

where  $\theta_{goal}$  and  $\theta_0$  are the robot-to-target angle and current tangent of wall respectively. This condition triggers the mode to APF if the difference between robot-to-goal angle and wall-following angle is greater than  $\frac{\pi}{2}$ . In other words, it happens when robot is moving away from the target.

This work is useful in an environment where the obstacle profile is simple. However, the failure case happens for complex environment as shown in Fig. 3. A closed room with small exit opposite to the goal direction is given and the robot is supposed to escape from the room and to move to the goal. The dashed and solid line represents the expected path and resulted path respectively. In the end, the robot will keep endlessly circling along the solid trajectory and fail to navigate to the target point. The root cause of this type of failure is the lack of self-awareness of the looping path.

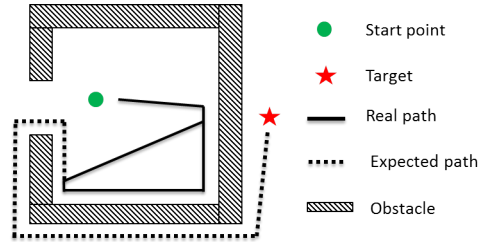


Fig. 3. Local Trap Issue for VFF-WFM method.

### B. Modified situation awareness with historical trajectory look-back

[17] reviewed the limitation of [18] and improved situation awareness with the look-back of historical trajectory. This paper argued that, without learning from the historical experience, the robot can make the same wrong decision repetitively. Hence in this work, the proposed situation awareness takes the visited locations into consideration. A new condition is added when switching from APF to WFM with comparison between the robot-to-target path and historical trajectory. The robot-to-target path is the straight line path from current position to the goal. The existence of intersection of the robot-to-target path and historical trajectory will prevent robot from switching to APF mode. Fig. 4 shows that the method can resolve the endless loop problem described in [18] with the same environment setup.

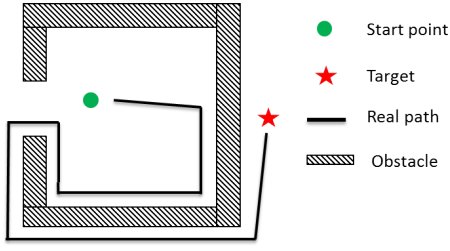


Fig. 4. Improved Path Planning with History Look-back.

However, this work does not make full use of the historical information, which can result in less efficient trajectory. An example is given in Fig. 5 where paths navigated by APF and WFM are represented by solid and dashed line respectively. When robot revisits this local minimum point, direction of wall following is changed so that the robot can move to the target point finally. However, repetitive path happens between point A and B marked in Fig. 5. This is because the above-mentioned situation awareness only checks the historical trajectory during WFM mode but not in APF mode. Hence, in the APF mode the robot does not perform checks for repeating decision. We believe that with a better decision mechanism, the trajectory can be further optimized to avoid the path between point A and B as shown in Fig. 5.

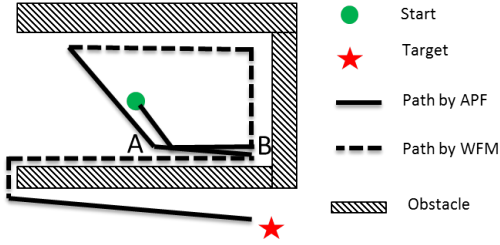


Fig. 5. Repetitive Path Generated by Improved Path Planning.

### III. PROPOSED MWF-APF METHOD

As discussed above, the look-back of historical trajectory plays an important role in the success of situation awareness system. Based on this, a new method is developed in this paper with memory based Wall-Following Method and Artificial Potential Field method, namely, MWF-APF method. More historical information will be considered in this enhanced situation awareness system.

#### A. Historical Trajectory Information

Similar to [17], the historical trajectory is stored as a series of key frames. Each key frame consists of time  $t$ , location information  $p(t)$  as well as current moving direction  $v(t)$ . In addition, a flag highlighting those key frames with local minimum of APF is added as extra information for the ease of processing. With storage space constraints in practical application, a new key frame will only be recorded when it satisfies any one of the three conditions below:

- i.  $\|p - p_{his}\| > d_{th}$ ,

- ii.  $\|p - p_{his}\| < d_{th}$  but  $|\angle(v, v_{his})| > \theta_{th}$ ,
- iii.  $\|F_{total}\| \leq F_{th}$ ,

where  $p_{his}$ ,  $v_{his}$  are the historical position and moving direction stored in the database and  $d_{th}$ ,  $\theta_{th}$  are the constants threshold to determine the data size. Constant  $d_{th}$  determines how frequently the key frame is updated. A frame is recorded when the distances between current robot position and all other key frame locations are greater than this threshold. If current position coincides with historical position but with moving direction difference greater than  $\theta_{th}$ , the current status will also be considered as a key frame. So when the robot revisits historical position but with a different moving direction, current status will be recorded as well. In addition, when local minimum point is found, the current status will be automatically considered as a key frame with the highlight of local minimum.

#### B. Artificial Potential Field

In APF mode, a virtual force is generated by a combination of both attractive and repulsive force. Assuming that the target position of the robot is  $P_t$  while the current position of the robot is  $P$ , then the attractive potential function takes a similar form as in [18]:

$$U_{att} = \begin{cases} \frac{1}{2} \zeta \|P - P_t\|^2, & \text{if } \|P - P_t\| \leq \rho, \\ \zeta \rho \|P - P_t\|, & \text{if } \|P - P_t\| > \rho. \end{cases} \quad (3)$$

Where  $\zeta$  and  $\rho$  are the coefficients to describe the strength of potential field. Similar to the calculation of potential field in physics, the attractive force is given by taking the partial derivative of the attractive potential field:

$$F_{att} = -\frac{\partial U_{att}}{\partial p} = \begin{cases} -\zeta (P - P_t), & \text{if } \|P - P_t\| \leq \rho, \\ -\zeta \rho \frac{P - P_t}{\|P - P_t\|}, & \text{if } \|P - P_t\| > \rho. \end{cases} \quad (4)$$

The resultant attractive force is constant when the robot is far away from the target. It decreases with decreasing distance to the target. Considering multiple obstacles surrounding the robot, let  $P_{obs,i}$  be the  $i^{th}$  obstacle position in the global frame, the repulsive potential can be calculated as

$$U_{rep,i} = \begin{cases} \frac{1}{2} \eta \left( \frac{1}{\|P - P_{obs,i}\|} - \frac{1}{d_c} \right)^2, & \text{if } \|P - P_{obs,i}\| \leq d_c, \\ 0, & \text{if } \|P - P_{obs,i}\| > d_c. \end{cases} \quad (5)$$

Where  $\eta$  is the repulsive potential field strength coefficient,  $d_c$  is the distance which the function is valid for. Similarly, repulsive force can be represented by

$$F_{rep,i} = \begin{cases} -\eta \left( \frac{1}{\|d\|} - \frac{1}{d_c} \right) \frac{d}{\|d\|}, & \text{if } \|d\| \leq d_c, \\ 0, & \text{if } \|d\| > \rho, \end{cases} \quad (6)$$

where  $d = P - P_{obs,i}$ . In real practice, global knowledge of environment is usually unavailable prior to the commencement of mission. Therefore, the position of obstacle is usually derived from the sensors mounted on the robot. For example,  $n$  laser rangefinders are mounted on a robot. The  $i^{th}$

rangefinder has an angle  $\theta_i$  with respect to the robot frame. For real-time processing, the obstacle can be estimated as

$$P_{obs,i} = P + d_i * \begin{bmatrix} \cos(\theta_i) \\ \sin(\theta_i) \end{bmatrix}, \quad (7)$$

where  $d_i$  is the reading of the  $i^{th}$  sensor. Finally, the overall artificial potential force is obtained by

$$F_{total} = F_{att} + \sum F_{rep,i}. \quad (8)$$

### C. Wall-Following Method

Wall-Following Method estimates the tangent and normal of the obstacle surface. The robot generates normal speed  $v_{normal}$  with a simple PID controller to keep a certain distance with the obstacle. The moving speed  $v_{tangent}$  is fixed with the tangent direction of obstacle surface. The velocity is hence given by

$$v_{total} = v_{normal} + v_{tangent}. \quad (9)$$

Another important factor in WFM is the decision on the wall-following direction. With historical trajectory look-back, same wall-following direction decision will be avoided when the robot revisits the position recorded in historical trajectory.

### D. Situation Awareness

With improvements from previous works described in the earlier sections, an enhanced situation awareness is proposed with capability to avoid the previously explained failure cases. We modified the existing conditions and added history look-back for both APF mode and WFM mode. The system will switch to APF from WFM with the simplified equations shown as follow:

- i.  $|\theta_{goal} - \theta_0| > \frac{\pi}{2}$ ,
- ii.  $\xi_t \cup \xi_{his} = \emptyset$ ,

where  $\xi_t$  is the robot-to-target straight line path and  $\xi_{his}$  is the historical trajectory. Both conditions have to be satisfied in order to trigger the WFM mode. When the robot is moving far away from the goal, the first condition tells that continuing Wall-Following Method may cause a longer trajectory. The set  $\xi_t$  simply draws a straight line from the current position to the goal. The robot will switch to APF mode only when the intersection between this line and historical trajectory is empty. Otherwise, the robot has a greater chance to revisit the previous path so switching to another mode is not suggested. An illustration when robot fails to meet the second condition is shown in Fig. 6. Switching conditions from APF to WFM are given by:

- i.  $F_{total} < F_{th}$ ,
- ii. If  $p = p(t_0)$  and  $v = v(t_0)$ ,  $t_0 < t_m$

where  $t_m$  records the time when the last local minimum appears,  $p$  is the current position,  $p(t_0)$  and  $v(t_0)$  are the position and moving direction at time  $t = t_0$ .

The WFM mode will be triggered when either condition is satisfied. The first condition checks the local minimum. Whenever it happens,  $t_m$  will be updated as the current time. The second condition is the key to avoid the revisit

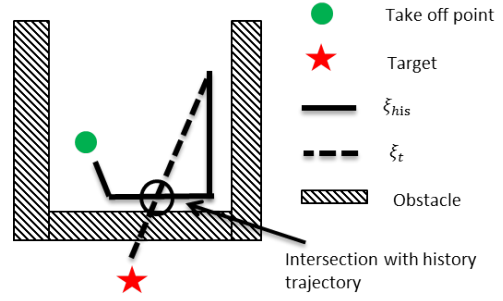


Fig. 6. obstacle-less APF path intersects with history trajectory.

of historical path in APF mode. If in this mode, the robot finds that its current status coincides (both position and moving direction) with the history frame at time  $t = t_0$ , then continuing APF mode should not navigate itself to the local minimum recorded at  $t = t_m$  again. In other words, the robot will switch to WFM if last local minimum with  $t = t_m$  happens after current position with history record at time  $t = t_0$ . If we do not trigger the mode in this case, the robot will likely make the same decision as it did from time  $t_0$  to  $t_m$ . Hence the repetitive path will happen.

## IV. SIMULATION RESULT WITH COMPLEMENTARY SENSORS

### A. Sensing System

In real life application, the information of the surrounding obstacles is often limited by the sensor constraints. In consideration of the implementation difficulty, cost and accuracy, we choose infrared sensor as sensing system. Infrared sensor is a time-of-flight(TOF) rangefinder with a small FOV. WLOG, the sensing system in the simulation contains 8 rangefinders equally distributed to get the 360-degree view of the surrounding environment.

### B. Surface Estimation in MWF Mode

In MWF mode, moving direction is determined by tangent of wall surface. The tangent of wall surface is estimated by selecting the tangent of nearest wall detected by the sensor. Each wall will be estimated by the following condition:

$$d_{ij} = \frac{\frac{1}{2}d_i d_j \sin(\theta_{ij})}{\sqrt{d_i^2 + d_j^2 - 2d_i d_j \cos(\theta_{ij})}}, \text{ if } \theta_{ij} < \frac{\pi}{2}, \quad (10)$$

where  $d_i$ ,  $d_j$  and  $\theta_{ij}$  are the readings of  $i^{th}$ ,  $j^{th}$  sensor and mounting angle between them. The minimum of  $d_{ij}$  indicates the wall closest to robot and hence the tangent of this wall is chosen as the moving direction. Fig. 7 gives an illustration of wall estimation.

### C. Simulation Result

The simulation is performed in RVIZ, a simulation tool based on Robotics Operating System (ROS). To validate the effectiveness of MWF-APF method, Fig. 8 gives two different cases. In Fig. 8(a), the obstacles leave small openings



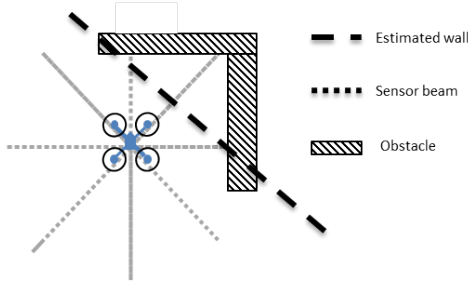


Fig. 7. Surface estimation with rangefinder.

in the corridor so that the robot can take a simple detour through the corridor. The planned path is straightforward as the heading of the robot does not change too much. In Fig. 8(b), it is more complex with the corridor closed. The robot has to make a u-turn and move outside corridor in order to navigate to the target. The robot is capable of planning different path based on the situation.

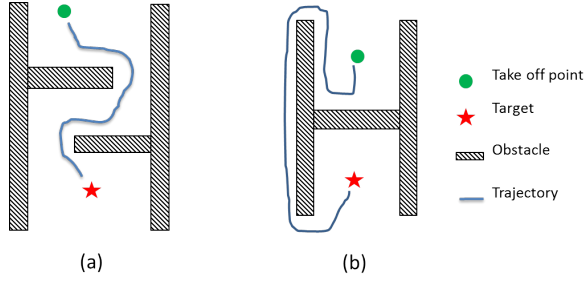


Fig. 8. (a) Result of path planning when there is a straight path (b) Result of H shape obstacle

A comprehensive comparison with the existing methods is shown in Fig. 9. In Fig. 9(a) and Fig. 9(b), we compare the traditional memory-less WFM-APF method with our MWF-APF method. The result shows that, with the new method the robot can navigate to the goal point while with the traditional method it fails. Fig. 9(c) shows the case where historical trajectory is only considered in WFM mode. Our method shown in Fig. 9(d) makes a shorter path than the method in Fig. 9(c).

## V. EXPERIMENT RESULT

In this section the proposed MWF-APF method is implemented on a quad-rotor. A short video of the experiment can be viewed at <https://youtu.be/E1h35GH9-tU>. A series of experiments are conducted in an indoor environment where motion capture system is used to provide position information of the UAV. The test field of the size 5m \* 5m is set up to form 3 obstacles in the center.

A quad-rotor is constructed that integrates distance sensors, a flight controller and an on-board computing unit. A photo of the quad-rotor is shown in Fig. 10. A group of 5 infrared time-of-flight (TOF) distance sensors is mounted above the UAV main frame for unobstructed distance measurement to 5 directions. The sensors are arranged such that

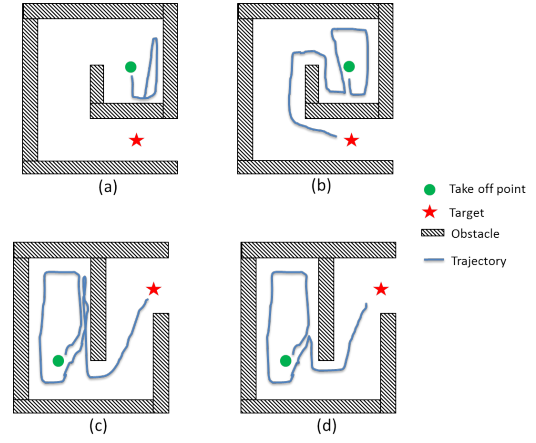


Fig. 9. (a)Result of traditional WFM-APF method. (b)Result of MWF-APF method. (c)Result of WFM-APF method with only historical look-back in WFM mode (d)Result of MWF-APF method

they face  $-90^\circ$ ,  $-45^\circ$ ,  $0^\circ$ ,  $45^\circ$  and  $90^\circ$  relative to vehicle front, so that a wide range of view is achieved.

The system overview of UAV is depicted in Fig. 11. The flight controller runs on-board PX4 firmware for position and attitude control. Position data is sent from an off-board motion capture system to the UAV through wireless XBee receiver. The distance sensor readings are sent to a 900MHz quad-core ARM embedded computer which runs Ubuntu and ROS. The proposed obstacle avoidance algorithm is executed in real time in ROS nodes which takes the UAV real-time position and distance sensor readings as inputs and outputs UAV's target velocity in lateral axes.



Fig. 10. UAV Mechanical Overview.

Multiple flight tests are conducted. During each flight, the UAV has to fly to a target position which is obstructed by multiple boxes. Obstacle profile is unknown prior to the flight and is detected online by the distance sensors. The UAV's lateral velocity is commanded by the MWF-APF algorithm while its altitude is maintained constant at 1m by the altitude controller. In each flight test, the UAV is able to reach the target position without colliding into the obstacles. Plots of some flight tests are shown in Figure.12. It is observed that the proposed method is able to switch between APF and WFM algorithms intelligently, allowing the UAV to effectively reach the target.

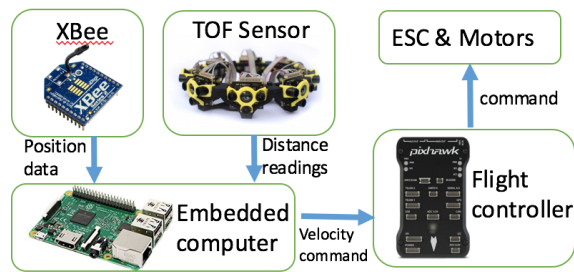


Fig. 11. UAV System Overview.

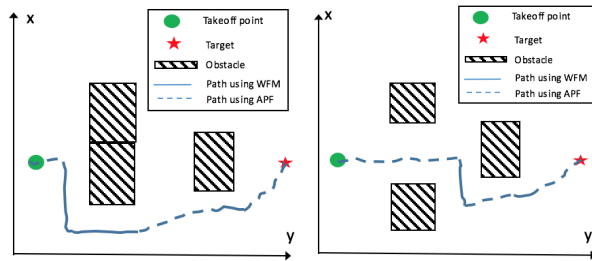


Fig. 12. Indoor experiment plot showing the UAV path in X/Y axes: UAV is able to switch between APF and WFM and reach target with an efficient path.

## VI. CONCLUSION

This paper presented a feasible and computationally efficient real-time path planning method for UAV with complementary sensors. The algorithm, namely memory based wall following-artificial potential field (MWF-APF) method, combines Wall-Following Method and Artificial Potential Field method with enhanced situation awareness. A comprehensive review of the previous path planning methods was given. To overcome the limitations of the existing methods, the new method includes the historical trajectory look-back on both APF and WFM modes. It provides an alternative solution to the endless loop problem of WFM and the local minimum problem of APF. The revised situation awareness is able to avoid repetitive path with guaranteed success of navigation compared to existing methods on WFM and APF. To the best of author's knowledge, the proposed situation awareness has better performance than existing works on method combining WFM and APF in terms of generated path length and success of navigation.

At the end of this paper, different simulation scenarios were given to show the robustness of the WFM-APF method in robot path planning. A collision-free UAV experiment was provided to validate its capability to complete real life mission. It was shown that only a few low-cost rangefinders are needed to execute the MFM-APF method and to complete the navigation task.

## ACKNOWLEDGEMENT

The author would like to thank Mr. Nguyen Pham Nhat Thien Minh for the great many suggestions during the planning and development of this research work.

## REFERENCES

- [1] N. Gageik, P. Benz, and S. Montenegro, "Obstacle detection and collision avoidance for a uav with complementary low-cost sensors," *IEEE Access*, vol. 3, pp. 599–609, 2015.
- [2] J. Park and Y. Kim, "Stereo vision based collision avoidance of quadrotor uav," in *Control, Automation and Systems (ICCAS), 2012 12th International Conference on*. IEEE, 2012, pp. 173–178.
- [3] Y. K. Kwag and C. H. Chung, "Uav based collision avoidance radar sensor," in *Geoscience and Remote Sensing Symposium, 2007. IGARSS 2007. IEEE International*. IEEE, 2007, pp. 639–642.
- [4] B. Zhu, L. Xie, D. Han, X. Meng, and R. Teo, "A survey on recent progress in control of swarm systems," *Science China Information Sciences*, vol. 60, no. 7, p. 070201, 2017.
- [5] H. Pham, S. A. Smolka, S. D. Stoller, D. Phan, and J. Yang, "A survey on unmanned aerial vehicle collision avoidance systems," *arXiv preprint arXiv:1508.07723*, 2015.
- [6] J. R. Andrews and N. Hogan, "Impedance control as a framework for implementing obstacle avoidance in a manipulator," Master's thesis, M. I. T., Dept. of Mechanical Engineering, 1983.
- [7] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on*. IEEE, 1991, pp. 1398–1404.
- [8] V. Lumelsky and A. Stepanov, "Dynamic path planning for a mobile automaton with limited information on the environment," *IEEE transactions on Automatic control*, vol. 31, no. 11, pp. 1058–1063, 1986.
- [9] K. Taylor and S. M. LaValle, "I-bug: An intensity-based bug algorithm," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 3981–3986.
- [10] N. Buniyamin, W. W. Ngah, N. Sariff, and Z. Mohamad, "A simple local path planning algorithm for autonomous mobile robots," *International journal of systems applications, Engineering & development*, vol. 5, no. 2, pp. 151–159, 2011.
- [11] Q.-L. Xu and G.-Y. Tang, "Vectorization path planning for autonomous mobile agent in unknown environment," *Neural Computing and Applications*, vol. 23, no. 7–8, pp. 2129–2135, 2013.
- [12] V. J. Lumelsky and T. Skewis, "Incorporating range sensing in the robot navigation function," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 20, no. 5, pp. 1058–1069, 1990.
- [13] I. Kamon and E. Rivlin, "Sensory-based motion planning with global proofs," *IEEE transactions on Robotics and Automation*, vol. 13, no. 6, pp. 814–822, 1997.
- [14] M. Finkelstein and B. Hunte, "Wall following with collision avoidance and mapping using a laser range finder," Tech. Rep., 2008.
- [15] I. Gavrilut, V. Tiponut, A. Gacsadi, and L. Tepelea, "Wall-following method for an autonomous mobile robot using two ir sensors," in *WSEAS International Conference. Proceedings. Mathematics and Computers in Science and Engineering*, no. 12. WSEAS, 2008.
- [16] M. Wang, Z. Su, D. Tu, and X. Lu, "A hybrid algorithm based on artificial potential field and bug for path planning of mobile robot," in *Measurement, Information and Control (ICMIC), 2013 International Conference on*, vol. 2. IEEE, 2013, pp. 1393–1398.
- [17] Y. Zhu, T. Zhang, and J. Song, "An improved wall following method for escaping from local minimum in artificial potential field based path planning," in *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*. IEEE, 2009, pp. 6017–6022.
- [18] X. Yun and K.-C. Tan, "A wall-following method for escaping local minima in potential field based motion planning," in *Advanced Robotics, 1997. ICAR'97. Proceedings., 8th International Conference on*. IEEE, 1997, pp. 421–426.