Improved Potential Field Method for Unknown Obstacle Avoidance Using UAV in Indoor Environment

Thi Thoa Mac[1,2], Cosmin Copot[1], Andres Hernandez[1] and Robin De Keyser[1]
[1] Department of Electrical energy, Systems and Automation (EeSA), Ghent University, Belgium
[2] School of Mechanical Engineering, Hanoi University of Science and Technology, Vietnam
Email: {Thoa.MacThi, Cosmin.Copot, Andres.Hernandez, Robain.DeKeyser}@Ugent.be

***Abstract*—This paper proposes a solution to real-time collision-free path planning for an AR. Drone 2.0 UAV using only on-board visual and inertial sensing. The proposed solution consists in a modified potential field method to overcome the non-reachable goal problem. The approach comprises three key components: pattern-based ground for localization, proposed potential field method for path planning and PD controllers for steering commands. By applying the proposed method, the quadrotor is successful in avoiding known/unknown obstacles and reaching the target in complex indoor environment. The results demonstrate the feasibility of the proposed strategy, which opens new possibilities for the agent to perform autonomous navigation.**

I. Introduction

In recent years, research interest in Unmanned Aerial Vehicles (UAVs) has emerged due to their potential use for a wide range of applications such as: autonomous indoor flying [1], obstacle avoidance [2], formation control of UGVs using an UAV [3], habitat mapping [4]. In order to accomplish the above mentioned missions, a prerequisite requirement is that the UAV should be able to implement real-time autonomous navigation in known and unknown environment.

The navigation of autonomous agent consists of four essential requirements known as: 1) *perception*, the agent uses its sensors to extract meaningful information; 2) *localization*, the agent determines its location in the working space; 3) *cognition and path planning*, the agent decides how to steer to achieve its goal; 4) *motion control*, the agent regulates its motion to accomplish the desired trajectory [5]. Among them, the path planning is the most challenging task, therefore, bringing attention of researchers. The potential field method (PFM) is peculiarly attractive since it has a simple structure, low computational complexity and easy to implement. In literature, there has been a significant amount of work based on this method applied to ground agents path planning [6][7][8][9][10]. However, there are only a few implementations of this method on UAVs.

An interesting work on implementing and flight testing of this approach on an UAV is studied in [11]. In order to operate in real-time, the authors developed a layered approach in uncharted terrain: plan globally and react locally. The global planner is based on an implementation of Laplaces equation that generates a potential function with a unique minimum at the target. The local planner is based on the modification of conventional potential field method in which not only the position of the UAV (as in the traditional PFM) but also the relative angles between the goal and obstacles are taken into account. More than 700 successful obstacle avoidance experiments were performed. However, this approach sometimes encounters problems when the repulsion from obstacles exceeded the physical constraints of the UAV. In [12], a solution for formation flight of UAVs is presented. This is based on potential field using a virtual leader and considering the vehicle's velocities. The method works properly in simulation; however, it has not been yet implemented on the real system.

It is pointed out that the potential field method has several inherent limitations [6][13] in which the non-reachable target problem is the most serious one and is worth investigating since it causes an incomplete path in the navigation task.

In this study, we develop a real-time implementation for an AR. Drone 2.0 UAV navigation in indoor environment using only on-board visual and inertial sensing. The approach consists of three key components: pattern-based ground for localization, proposed potential field method for path planning and PD controllers for steering commands. The contribution of this work consists in providing a solution to solve the non-reachable target problem in conventional PFM. The new potential field method ensures that the goal is the unique global minimum. In addition, a realistic navigation approach in complex known and unknown indoor environment is proposed for the UAV. The motivation behind this research is to illustrate that autonomous navigation is feasible on low-cost UAV.

The paper is organized as follows: a brief description of the AR. Drone 2.0 UAV used in this study in section II, the system setup and localization are presented in detail in section III. The proposed path planning algorithm based on PFM is described in section IV and the control approach is given in section V. In section VI, the experimental results are presented, followed by section VII where the main outcomes of this work are summarized.

II. Hardware platform

A. Plant Description

The Ar. Drone 2.0, a commercial and low-cost micro UAV, is used in this research. It consists of four propeller blades

978-1-4673-8740-8/16/$31.00 ©2016 IEEE

arranged symmetrically around a central unit which holds the sensory equipment and the circuit board. There are four basic motions of this quadrotor: pitch, roll, throttle and yaw as shown in Fig.1.

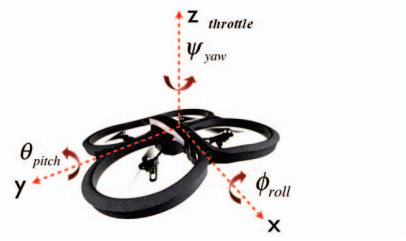

Fig. 1. The movements of an AR.Drone 2.0: Pitch θ, Roll ϕ, Yaw ψ, Throttle z .

The sensor system has several motion sensors which together form the Inertial Measurement Unit (IMU). Communication between the quadrotor and a command station is performed via Wi-Fi connection within a 50 meters range. AR. Drone 2.0 is equipped with two cameras in the bottom and the frontal parts. The bottom camera has an image size of 320 x 240 pixels at 60 frames per second (fps) which is used for localization purpose in this research. The frontal camera has a resolution of 640 x 360 pixels at 30 fps used to detect obstacles.

Several Software Development Kits (SDK) have been developed for Windows, Linux or iOS operating systems [14][15][16], thus enabling AR. Drone 2.0 to be manipulated from a computer, smart phone or tablet. In this work, the AR. Drone is controlled from a computer using Windows 7 with Visual Studio C++, OpenCV and AR. Drone libraries.

B. *Analysis of Inputs and Outputs*

The developed SDK mode allows to read from the quadrotor information the roll angle (rad), pitch angle (rad), the altitude (m), yaw angle (rad) and the linear velocities on longitudinal/ transversal axes (m/s). They are respectively denoted by $\{\theta_{out}, \phi_{out}, \zeta_{out}, \psi_{out}, \dot{x}, \dot{y}\}$. The system is affected by four inputs $\{V_{in}^x, V_{in}^y, \dot{\zeta}_{in}, \dot{\psi}_{in}\}$ which are the linear velocities on longitudinal/ transversal axes, vertical speed and yaw angular speed references as depicted in Fig. 2. The control parameters given to the internal controllers are floating point values between [-1, 1]. Those parameters are not directly the control parameters values, but a percentage of the maximum corresponding values are set in the drone.

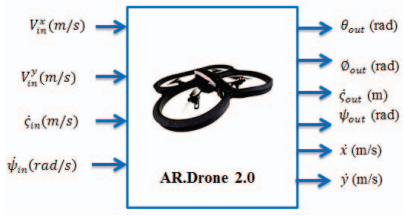

Fig. 2. Inputs and Outputs of an AR.Drone 2.0.

C. *System identification*

A quadrotor is a multi-variable and naturally unstable system. However, due to the internal low layer control implemented in the embedded operative system, AR. Drone 2.0 is considered as a Linear Time Invariant (LTI) System, which can be decomposed into multiple SISO loops. Transfer functions are obtained via parametric identification using the prediction error method and Pseudo-Random Binary Signal (PRBS) input signals [17]. A sampling time of 5 ms for yaw and 66 ms for other degrees of freedom are chosen based on the analysis of dynamics characteristic performed on the previous studies [18][19]. The identified transfer functions are:

$$\begin{aligned} H_x(s) &= \frac{\dot{x}(s)}{V_{in}^x(s)} = \frac{7.27}{(1.05s+1)}e^{-0.1s} \\ H_y(s) &= \frac{\dot{y}(s)}{V_{in}^y(s)} = \frac{7.27}{(1.05s+1)}e^{-0.1s} \\ H_{altitude}(s) &= \frac{\zeta_{out}(s)}{\dot{\zeta}_{in}(s)} = \frac{0.72}{s(0.23s+1)}e^{-0.1s} \\ H_{yaw}(s) &= \frac{\psi_{out}(s)}{\dot{\psi}_{in}(s)} = \frac{2.94}{s(0.031s+1)}e^{-0.1s} \end{aligned} \quad (1)$$

III. System Setup and Localization

Our approach consists of three major components running on a laptop connected to the quadrotor via wireless communication as shown in Fig.3. The first component is pattern-based localization, which allows Ar. Drone 2.0 to determine its location and orientation in a working space. The second component is IMU data process, which delivers and receives signals between the quadrotor and the ground station. For more explanation, please refer to section II. The last component is a cascade control, which guides the quadrotor to follow the designed trajectories. For details, see section V.

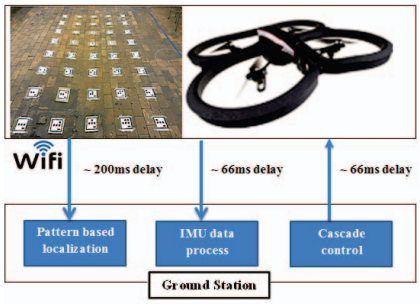

Fig. 3. Our navigation approach of an AR.Drone 2.0.

In order to perform the localization task, a grid of ground patterns is used to estimate the drone pose both in orientation and position. Each pattern represents a certain (x,y) coordinates inside this grid. The (x,y) coordinates are calculated based on the information in the first and second rows. Each row includes three bits, the bit values are zero and one corresponding to the white and black bit. Fig. 4 represents the

position of the ground pattern with (x, y) coordinates calculated as below:

$$\begin{aligned} x &= 2^0 x_0 + 2^1 x_1 + 2^2 x_2 \\ y &= 2^0 y_0 + 2^1 y_1 + 2^2 y_2 \end{aligned} \tag{2}$$

The rectangle at the right bottom of the pattern is utilized to approximate the orientation of the quadrotor. The distance between two patterns is 50 cm to ensure that there is always at least one pattern visible in one image.

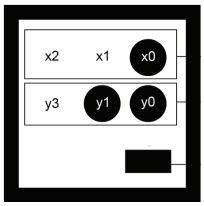

Fig. 4. A Ground pattern representing position (x = 1, y = 3).

In this study, the raw image must be further processed with OpenCV to achieve the quadrotor pose information. The image processing algorithm is depicted in flowchart as shown in Fig. 5. Briefly, the input image is converted into gray image, then applied a suitable threshold to find the contours. This threshold depends on the light condition in indoor environment and therefore has to be appropriately chosen. The drone orientation and position in the world coordinates are lastly calculated to control the quadrotor motion.

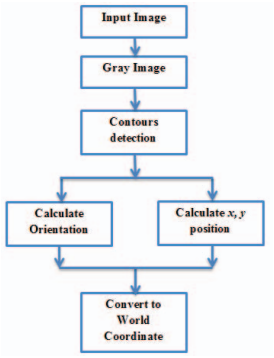

Fig. 5. Image process flowchart of pattern based localization.

IV. Path Planning

Path planning is defined as a designing collision-free path in a working environment with obstacles. A path is a set of configuration $\overrightarrow{q} = \{ q_0, q_1, ..., q_{goal} \} \in \mathscr{R}^n$ of the agent that connects the starting position q_0 and the final position q_{goal}.

In this section, conventional PFM is firstly introduced, then the non-reachable goal problem in PFM is described. Lastly, the proposed method termed modified potential field method (MPFM), is presented to overcome that problem. This method ensures that the target position is the unique global minimum of the total potential.

A. Conventional potential field method and its limitations.

In conventional PFM, the agent is considered as a point mass and moves in a two-dimensional working space. The obstacles exert repulsive forces while the target applies an attractive force to the agent. The total force determines the subsequent direction and speed of the agent. Conventionally, the attractive potential is defined as a function of the relative position between the agent and the target while the relative positions between the agent and the obstacles are used to calculated repulsive potential. The basic attractive and repulsive potential functions are:

$$\begin{aligned} U &= U_{att} + U_{rep} \\ U_{att} &= \frac{1}{2} k_1 d^2(q, q_{goal}) \\ U_{rep} &= \begin{cases} \frac{1}{2} k_2 (\frac{1}{d(q,q_{obs})} - \frac{1}{d_0})^2 & \text{if } d(q,q_{obs}) \le d_0 \\ 0 & \text{if } d(q,q_{obs}) > d_0 \end{cases} \end{aligned} \tag{3}$$

where k_1 and k_2 are positive coefficients; d_0 is the affected distance of obstacle; $d(q, q_{goal})$ is the distance between the agent and the target; $d(q, q_{obs})$ is the minimum distance between the agent and the obstacles.

The corresponding attraction force F_{att} and repulsion force F_{rep} are negative gradients of respective attraction $U_{(att)}(q)$ and repulsion $U_{rep}(q)$ whose formulas are:

$$\begin{aligned} F &= F_{att} + F_{rep} \\ F_{att} &= -k_1 d(q, q_{goal}) \\ F_{rep} &= \begin{cases} -k_2 (\frac{1}{d(q,q_{obs})} - \frac{1}{d_0})(\frac{1}{d(q,q_{obs})})^2 & \text{if } d(q,q_{obs}) \le d_0 \\ 0 & \text{if } d(q,q_{obs}) > d_0 \end{cases} \end{aligned} \tag{4}$$

Although the conventional PFM effectively generates smooth path, it encounters several problems such as [6][13]:

1) Non-reachable target.
2) No passage between closely spaced obstacles.
3) Oscillations in the presence of obstacles.
4) Oscillations in narrow passages.

Among the above mentioned problems, the non-reachable target problem is the most serious one and is worth investigating since it causes an incomplete path in the agent navigation task. This problem happens when the target is close to obstacles. In that case, when the agent approaches the target, it also approaches the obstacles. As a sequence, the attractive force reduces while the repulsive force increases. Therefore, the agent is trapped in local minima and oscillations might occur.

Fig. 6 (**Left**) illustrates the case that there are several obstacles located near the target.The repulsive force in this case is considerably larger than the attractive force, thus the agent is repulsed away rather than reaching the target.

Fig. 7 (**Left**) shows another case in which the attractive field and the repulsive field are collinear in opposite directions and the total force approximates zero. Using conventional PFM, the agent is trapped in local minima.

B. Proposed attractive and repulsive potential

The essential cause of the non-reachable target problem is that the goal position is not a global minimum of the total potential U in equation (3). When the agent reaches the target, attractive potential U_{att} is equal to zero; however, the repulsive potential U_{rep} is none-zero if there is at least one obstacle which satisfies the condition $d(q,q_{obs}) < d_0$. In order to overcome that limitation, a MPFM is proposed to ensure that the total potential field force has the unique global minimum at the target position.

Obviously, if the repulsive potential approaches zero as the agent reaches the target, the total potential attains the global minimum at the goal. As a consequence, it is necessary to introduce the relative distance between the agent and the target into the formula of repulsive potential. Furthermore, since the agent is unable to stop suddenly at the target position while it is moving at a high speed, the agent velocity term is taken into account in the proposed attractive potential formula.

By adding the term $d(q,q_{goal})$ in the formula of repulsive potential and the agent velocity $\dot{q}$ in the formula of attractive potential, the total potential $U = U_{att} + U_{rep}$ obtains the global minimum (0) if and only if $q = q_{goal}$ and $\dot{q} = 0$. In other words, it happens if and only if the agent arrives at the target and keep that position. The MPFM are formulated as follows:

$$\begin{aligned} &U = U_{att} + U_{rep} \\ &U_{att}(q,\dot{q}) = \rho_p d^2(q,q_{goal}) + \rho_v \dot{q}^2 \\ &U_{rep} = \begin{cases} \frac{1}{2}\alpha(\frac{1}{d(q,q_{obs})} - \frac{1}{d_0})^2 d^{\beta}(q,q_{goal}) & \text{if } d(q,q_{obs}) \le d_0 \\ 0 & \text{if } d(q,q_{obs}) > d_0 \end{cases} \end{aligned} \tag{5}$$

The attractive force and repulsive force are the negative gradients of attractive potential and repulsive potential as follows:

$$\begin{aligned} &F = F_{att} + F_{rep} \\ &F_{att} = -2\rho_p d(q,q_{goal}) - 2\rho_v \dot{q} \\ &F_{rep} = \begin{cases} F_{rep1} + F_{rep2} & \text{if } d(q,q_{obs}) \le d_0 \\ 0 & \text{if } d(q,q_{obs}) > d_0 \end{cases} \\ &F_{rep1} = -\alpha(\frac{1}{d(q,q_{obs})} - \frac{1}{d_0})\frac{d^{\beta}(q,q_{goal})}{d^2(q,q_{obs})} \\ &F_{rep2} = -\frac{\alpha\beta}{2}(\frac{1}{d(q,q_{obs})} - \frac{1}{d_0})^2 d^{\beta-1}(q,q_{goal}) \end{aligned} \tag{6}$$

In previous PFM studies, an obstacle is considered as a single point. This strategy only effectively works when the obstacle dimensions are small. Therefore, in this study, large obstacles are divided into several small obstacles before applying the proposed method.

Applying conventional PFM, the agent is not successful in autonomous navigation tasks when the obstacles are located near the target as mentioned in subsection IV. *A*. However, the proposed method can handle such problems properly because it reduces the repulsive force when the agent moves towards the target. Thus, the agent enables to reach the target successfully as shown in Fig. 6 (**Right**) and Fig. 7 (**Right**). In short, it indicates that the proposed method effectively solves the non-reachable target problem when the obstacles are located near the target .

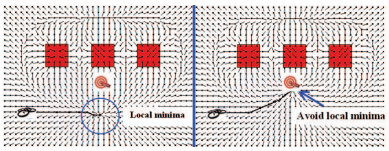

Fig. 6. The problem and the solution when the position of target is very close to obstacles. **Left:** the target non- reach problem of conventional PFM, **Right:** Avoid local minima solving the target non-reachable problem of MPFM.

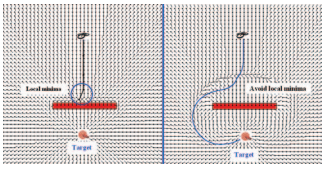

Fig. 7. The problem and the solution when the agent, obstacle and target are aligned in which the obstacle in the middle and the attractive force approximates the repulsive force. **Left:** the target non- reach problem of conventional PFM, **Right:** Avoid local minima solving the target non-reachable problem of MPFM.

The proposed approach is developed to appropriately work in known and unknown complex indoor environment. First, the global agent path planning is generated based on the proposed MPFM, then this path is renovated when the agent senses a new obstacle until reaching the target (local path). The algorithm is presented in Fig. 8.

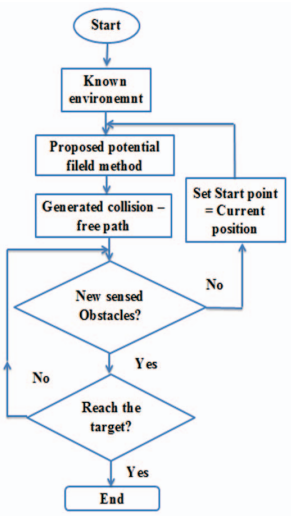

Fig. 8. The flowchart of the agent path planning in known and unknown environment based on MPFM.

C. Simulation results under complex environment

To validate the proposed algorithm, simulations are carried out under various complex environment conditions with known and unknown obstacles. There are several large obstacles with

different dimensions and shapes located in the way of the agent. Fig. 9 shows two different cases of complex indoor environment with completely known obstacles in which the agent arrives the target without collision with obstacles using MPFM.

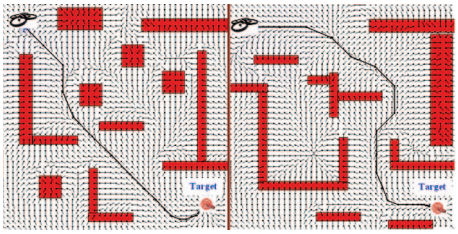

Fig. 9. Proposed MPFM under the complex environment.

The algorithm is also tested in complex scenarios with unknown obstacle as shown in Fig. 10. The known obstacle is presented in red while the unknown obstacle is presented in black. Since agent has no information about the black obstacle in advance, the trajectory is generated to avoid only red obstacles as shown in Fig. 10 (**Left**). However, it updates the path immediately (local path) when detecting a new obstacle (the obstacle changes its color to orange) as displayed in Fig. 10 (**Right**). The results prove the feasibility and effectiveness of the algorithm in complex environment with known and unknown obstacles.

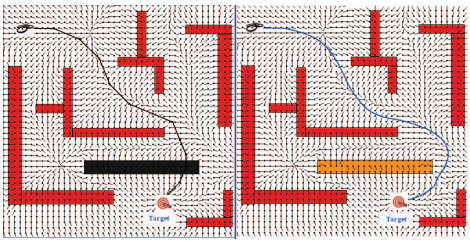

Fig. 10. Proposed MPFM under the complex partial known environment, **Left:** The agent collides black (unknown) obstacle. **Right:** The agent re-plans the path according to perceived environment.

V. Control Design

In this section, the control approach for the quadrotor navigation is presented. First, the overview of the control structure is introduced. Then, the compensation strategy for the quadrotor is proposed and PD controllers are designed.

In order to perform real experiments, a cascade control is designed (Fig. 11) such that the quadrotor follows the generated trajectories. There are two parts of the cascade controller: inner-loop controller and outer-loop controller. The inner-loop controller is performed inside the quadrotor as a black-box. The model identified in section II. C is used to identify the relationships between the inputs and the outputs of this black-box. The outer-loop controller is designed to implement position control.

In the outer-loop controller, the localization process provides the current quadrotor pose in the world coordinate based

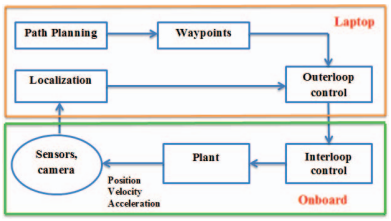

Fig. 11. Cascade control of the quadrotor.

on the ground patterns. The obtained free-collision path using MPFM is sent to the controller by a list of way-points. Because of missing communication, oscillating and uncertain noise, the quadrotor can be unable follow the designed path. Therefore, it is necessary to generate a compensation path, named, $\overrightarrow{Tar}$ that guides the quadrotor to return to the designed trajectory.

In order to perform the compensation, several definitions are introduced. "Path error" is defined as the distance between the drone and the designed path at a time during flight. Suppose the drone is currently in the position with the two nearest way-points, named, "Previous way-point" and "Next way-point". Let's define $\overrightarrow{Target}_{WP}$ as the vector from current position to the "Next way-point" and $\overrightarrow{Target}_{pathline}$ as the vector towards the perpendicular path as shown in Fig. 12. The compensation $\overrightarrow{Tar}$ is only executed when the "Path error" (L) is larger than threshold value L_0. The formula of the target vector is:

$$\overrightarrow{Tar} = \lambda * \overrightarrow{Target}_{WP} + (1-\lambda) * \overrightarrow{Target}_{pathline} \tag{7}$$

where λ is defined parameter as following:

$$\lambda = \begin{cases} 1-(L/L_0)^n & \text{if } L < L_0 \\ 0 & \text{if } L > L_0 \end{cases}$$

It is important to notice that the $\overrightarrow{Tar}$ should not significantly change when the drone is around the trajectory with a small "Path error" and should quickly react with a large "Path error". For our system, $n = 4$ and $L_0 = 0.5\ m$ has been chosen.

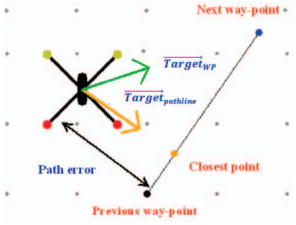

Fig. 12. Compensation strategy for the quadrotor.

The tuning of the controllers are implemented using Frequency Response Toolbox (FRTool) [20]. The following controllers were designed for navigation tasks:

$$\begin{aligned} C_{dotx}(s) &= 0.45(1.64s+1) \\ C_{doty}(s) &= 0.45(1.64s+1) \\ C_{altitude}(s) &= 4 \end{aligned} \tag{8}$$

VI. Experiments and Results

The virtual environment is developed in order to monitor on-line and facilitate off-line tests of the quadrotor navigation tasks. The prime principle of the virtual environment in this study is presented in Fig.13. In this figure, the comparison between the virtual environment (left) and actual environment (right) is shown. When the drone detects the real obstacle in actual environment, that one is displayed in the virtual environment as brown box. Then it is automatically enlarged into yellow box to create the safety area for the drone. The start point, the target and the dimensions of the working space are also shown in the virtual environment.

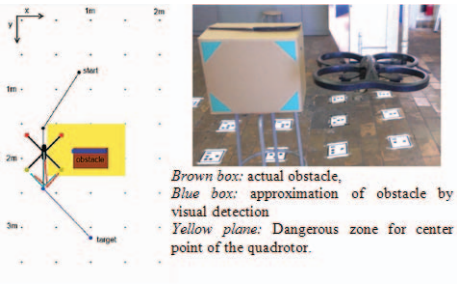

Fig. 13. Virtual vs. Actual environment in the real-time experiment.

The results obtained with pattern-based localization, MPFM and PD control in the real system are presented in Fig. 14. The bounded rectangle presents the walls of indoor working environment. The red obstacles are known obstacles and the black one is unknown obstacle. The quadrotor detects and avoids the obstacles successfully. The black path is reference path obtained by MPFM and the blue one is real path obtained by experiment. It is possible to observe that the quadrotor has few deviations to the desired trajectory, acceptable overshoot at the moment of performing sharp bends.

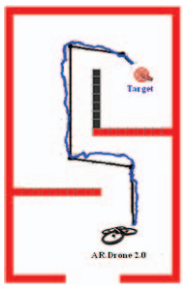

Fig. 14. Real-time obstacle avoidance of the AR. Drone 2.0 with pattern-based localization, MPFM and PD controller.

VII. Conclusion

In this study, we have presented the path planning approach for a low-cost commercial AR. Drone 2.0 during indoor flight using only on-board visual and internal sensing. The approach consists of three fundamental elements: pattern-based ground for localization, proposed potential filed method for path planning and PD controller for motion control. The proposed modified potential filed method (MPFM) enables to solve the non-reachable target problem and successfully avoids unknown obstacles. In addition, the virtual environment is designed to monitor on-line and test off-line the quadrotor navigation. The performed tests using the virtual environment and real-time experiment demonstrate the feasibility of the proposed strategy, which opens new possibilities of autonomous navigation for the mobile agent in complex known and unknown indoor environment. For future work, the approach is expected to be implemented in a dynamic working space.

References

[1] S. Grzonka, G. Grisetti, W. Burgard, Towards a Navigation System for Autonomous Indoor Flying, IEEE Int. Conf. Robotics and Automation, Kobe, 2009, pp. 2878−2883.
[2] J. Moon, J. V. R. Prasad, Minimum-time approach to obstacle avoidance constrained by envelope protection for autonomous UAVs, Mechatronics 21, pp. 861−875, 2011.
[3] A. Hernandez, C. Copot, J. Cerquera, H. Murcia, R. De Keyser, Formation Control of UGVs using an UAV as Remote Vision Sensor, Proceedings of the 19th IFAC World Congress, Cape Town, South Africa, 2014, pp. 618 −623.
[4] N. Dijkshoorn, Simultaneous localization and mapping with the AR.Drone, Master's thesis, University of Amsterdam, 2012.
[5] R. Siegwart, I. R. Nourbakhsh, Introduction to autonomous mobile robot, Massachusetts Institute of Technology press, Cambridge, U.S.A, 2004, ch.1.
[6] G. Li, A. Yamashita, H. Asama, Y. Tamura, An Efficient Improved Artificial Potential Field Based Regression Search Method for Robot Path Planning, IEEE International Conference on Mechatronics and Automation, Chengdu, China, 2012, pp. 1227−1232.
[7] J. Sfeir, M. Saad, H. S. Hassane, An Improved Artificial Potential Field Approach to Real-Time Mobile Robot Path Planning in an Unknown Environment, IEEE International Symposium on Robotic and Sensors Environments (ROSE), Montreal, QC, 2011, pp. 208−213.
[8] S. S. Ge, Y. J. Cui New Potential Functions for Mobile Robot Path Planning IEEE transactions on robotics and automation, Vol. 16, No. 5, pp. 615−620, 2000.
[9] F. A. Cosfo, M. A. P. Castaida, Autonomous Robot Navigation using Adaptive Potential Fields, Mathematical and Computer Modelling 40, pp. 1141−1156, 2004.
[10] L. Valbuena, Hybrid Potential Field Based Control of Differential Drive Mobile Robots H. G. Tanner, J Intell Robot Syst 68, pp. 307−322, 2012.
[11] S. Scherer, S. Singh, L. Chamberlain, M. Elgersma, Flying fast and low among obstacles: Methodology and experiments. International Journal of Robotics Research, 27(5), pp. 549−574, 2008.
[12] T. Paul, T. R. Krogstad, J. T. Gravdahl, Modelling of UAV formation flight using 3D potential field, Simulation Modeling Practice and Theory 16, pp.1453−1462, 2008.
[13] K. Borenstein, J. Borenstein, Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation, Proceedings of the IEEE Conference on Robotics and Automation, Sacramento, California, USA, 1991, pp. 1398−1404.
[14] S. Piskorski, N. Brulez, E. Eline, and F. D'Haeyer, Ar. drone developer guide-sdk 2.0, 2012.
[15] T. Krajnik, V. Vonasek, D. Fiser, and J. Faigl, Ardrone as a platform for robotic research and education, Research and Education in Robotics - Eurobot, Volume 161, pp.172−186, 2011.
[16] https://github.com/puku0x, cv drone free software, December 2013.
[17] L. Ljung, System identication: theory for the user, Prentice-Hall, 2007.
[18] T. Vlas, A. Hernandez, C. Cosmin, I. Nascu, R. De Keyser, Identication and Path Following Control of an of an AR.Drone quadrotor, IEEE 17th International Conference on system Theory, Control and Computing, Sinaia, Romania, 2013, pp. 583−588 .
[19] A. Hernandez, H. Murcia, C. Copot, R. De Keyser, Towards the Development of a Smart Flying Sensor: Illustration in the Field of Precision Agriculture, Sensors (Basel) 15(7), pp. 16688-16709, 2015.
[20] R. De Keyser, C. M. Ionescu, FRTool: a frequency response tool for CACSD in Matlab, in Proc. of the IEEE Conf. on Computer Aided Control Systems Design, Munich, Germany, 2006, pp. 2275−2280.