Proceeding of the 2015 IEEE
International Conference on Information and Automation
Lijiang, China, August 2015

Sense and Collision Avoidance of Unmanned Aerial Vehicles Using Markov Decision Process and Flatness Approach

Yu Fu and Xiang Yu, *Senior Member, IEEE*
Department of Mechanical and Industrial Engineering
Concordia University
Montreal, Quebec, H3G 1M8, Canada
f_yu12@encs.concordia.ca
xiangyu1110@gmail.com

Youmin Zhang, *Senior Member, IEEE*
Department of Information and Control Engineering
Xi'an University of Technology
Xi'an, Shaanxi, 710048, China
**Corresponding author on sabbatical leave from*
Concordia University: ymzhang@encs.concordia.ca

***Abstract*— This paper presents a new development of collision avoidance algorithm that ensures an Unmanned Aerial Vehicle (UAV) can avoid multiple intruders autonomously. Firstly, the Markov Decision Process (MDP) based approach generates the multiple threats resolution logic for the collision avoidance system. Secondly, the optimal trajectory is smoothed by the differential flatness technique where the constraints of the UAV dynamics are considered. In such a way, the planned trajectory is feasible for the UAV. The effectiveness of the developed scheme is illustrated by the numerical simulation studies.**

***Index Terms*— Collision avoidance; MDP; differential flatness; optimal trajectory; UAV.**

I. Introduction

A. Scope of Sense and Avoid

Unmanned Aerial Vehicles (UAVs) can carry out more complex civilian and military applications with less cost and more flexibility in comparision of manned aircraft. Mid-air collision thus becomes an important problem considering the safe operation of air transportation systems, once UAVs will be used more with various applications and share the same airspace with manned air vehicles. To ensure safe flights, Detect & Avoid (D&A) systems are equipped in manned systems, which are capable of detecting airplanes in airspace and performing necessary maneuvers to avoid collision threats. Unlike manned aircraft with a human pilot involved, UAVs have to configure Sense and Avoid (S&A) systems for guaranteeing the safety of flight. S&A systems are therefore one of the key components to safely integrate UAVs into the airspace [1]. As illustrated in Fig. 1, a complete S&A paradigm consists of four units: sensing, conflict detection, collision avoidance, and evasion maneuver generation, respectively. The conflict is an event in which two or more aircraft break the defined minimal separation criterion. The criterion of the en-route horizontal separation in civilian air-traffic is 5 NM and of the vertical separation is 1000 ft [2]. When a potential collision is predicted to take place with respect to the sensed data, the trajectory of UAV has to be re-planned for collision avoidance. The major collision avoidance approaches can be categorized into five categories: Rule-Based (RB) [3], Game Theoretical (GT) [4], Force Field (FF) [5], Geometric [6], and Numerical Optimization (NO) [7], respectively.

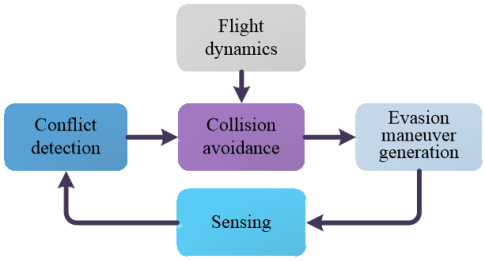

Fig. 1. The structure of S&A functionality in UAV

B. Overview of MDP Based Collision Avoidance

To generate certain maneuvers for a UAV to avoid collisions without incurring too much cost, the Markov Decision Process (MDP) based NO approach is considered to guide UAV in the case of multiple aircraft. By constructing a MDP in [8], an optimal path is derived to successfully avoid multiple guns and reach the target. In [9], using potential field and probabilistic map, the mechanism formulated by MDP fuses navigation decision for a team of UAVs to track targets, considering the uncertainty of the environment. Modeling UAV collision avoidance as continuous-state partially observable MDP, the threat resolution is generated by solving these models using Monte Carlo Value Iteration [10]. With the combination of a variety of sensor modalities, intruder behavior, aircraft dynamics, and cost function, the policies produced by MDP can be implemented to create an optimal avoidance strategy in real time [11]. A partially observable MDP based path planning for UAVs is developed to track targets while evading threats under wind disturbances [12]. The authors propose a discounted MDP using the Q-Learning algorithm to obtain optimal guidance policy without explicit knowledge of the system models and environmental conditions [13].

978-1-4673-9104-7/15/$31.00 ©2015 IEEE

C. The Contribution of This Paper

However, these proposed MDP-based algorithms for collision avoidance do not incorporate dynamic constraints into path planning while the flight envelope constraints have to be taken into account in practice. This study develops an optimal path through MDP to avoid **multiple threats**. Moreover, differential flatness-based algorithm smooths the planned path under the dynamic constraints on UAV motion. The main contribution of this paper includes: 1) presenting a method to model **multiple threats avoidance** as MDP which is appropriate for real-time applications; 2) taking the constraints of UAV into consideration when planning a path.

D. The Arrangement of This Paper

The remainder of the paper is organized as follows: Section II presents the problem formulation of collision avoidance. In Section III, MDP-based method is deployed to produce a trajectory directing a UAV from a start position to a destination evading multiple intruders threats. The generated trajectory is smoothened by flatness-based approach. Section IV evaluates the simulation results in MATLAB. Finally, conclusions are drawn in Section V.

II. Problem Formulation

The specific problem to be addressed is how to navigate a UAV from an initial position to a destination in the two-dimensional plane with capability of avoiding multiple intruders.

A. Flight Dynamics Modeling

In this work, a reduced-order nonlinear dynamic equations are used to model the aircraft motion. The equations of motion are as follows:

$$\begin{cases} \dot{x} = V_a \cos\psi \cos\gamma \\ \dot{y} = V_a \sin\psi \cos\gamma \\ \dot{h} = \sin\gamma \end{cases}, \tag{1}$$

where (x, y) is the position of UAV with respect to earth coordinate system, where the positive x-direction points east and positive y-direction points north. V_a, ψ, γ, and h represent the speed, heading angle, flight path angle, and altitude of UAV, respectively. Assuming the UAV is flying in two-dimensional plane with constant speed, the flight path angle equals to zero. The aforementioned motion equation can be subsequently written as:

$$\begin{cases} \dot{x} = V_a \cos\psi \\ \dot{y} = V_a \sin\psi \\ \dot{h} = 0 \end{cases}. \tag{2}$$

For the sake of UAV safely turning, it is required that dynamic constraints are not exceeded to prevent the UAV into an irreversible state. Therefore, a coordinated turn is considered to describe a turning maneuver. The coordinated turn condition is expressed by:

$$\dot{\psi} = \frac{g}{V_a} tan\phi, \tag{3}$$

where ϕ and g denote the roll angle for UAV and gravitational accelaration, respectively. ϕ is the control input of UAV, which is assumed to be bounded as $|\phi| \leq \phi_{max}$.

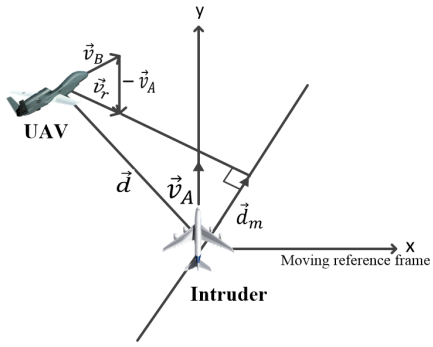

Fig. 2. The relative motion of UAV and Intruder

B. Formulation

As shown in Fig. 1, whether the trajectory of UAV has to be re-planned for avoiding collision or not depends on the feedback of the collision detection. Detecting the conflict is determined by the miss distance at the closet point approach [14].

The conflict detection between the UAV and multiple aircraft is only considered in a common horizontal plane. If the predicted distance between the UAV and the intruder is smaller than the minimum separation distance r_{safe} within a specific block of time T_{det}, the collision will occur. It is also assumed that intruders follow the initial trajectory without any avoidance maneuver. The velocity of the intruders can therefore be calculated. The relative motion of the UAV and one intruder is shown in Fig. 2. For the UAV, the closest approach distance, $\vec{d}_m$ can be derived as:

$$\vec{d}_m = \vec{v}_r \times (\vec{d} \times \hat{v}_r), \tag{4}$$

where $\vec{d}$ denotes the vector locating the intruder with respect to the UAV, and $\hat{v}_r$ is the unit vector in the direction of the relative velocity vector of the UAV with respect to the intruder:

$$\hat{v}_r = \frac{\vec{v}_r}{\|\vec{v}_r\|}. \tag{5}$$

The relative velocity of the UAV with respect to the intruder, $\vec{v}_r$, is obtained by:

$$\vec{v}_r = \vec{v}_B - \vec{v}_A, \tag{6}$$

where $\vec{v}_r$ is located at the vector direction of the time to the closest point τ. With the relation between $\vec{d}_m$ and $\vec{d}$, $\vec{d}_m$ can

be derived as:

$$\vec{d}_m = \vec{d} + \vec{v}_r \tau, \tag{7}$$

where the miss vector $\vec{d}_m$ and the relative velocity vector $\vec{v}_r$ are orthogonal, as shown in Fig. 2:

$$\vec{d}_m \cdot \vec{v}_r = 0. \tag{8}$$

Combining Eqs. (7) and (8), τ can be derived as:

$$\tau = -\frac{\vec{d} \cdot \vec{v}_r}{\vec{v}_r \cdot \vec{v}_r}. \tag{9}$$

If τ is positive and $\|d_m\|$ is smaller than d_{safe}, a collision will take place between the UAV and the intruder. Thus the conflicts between multiple intruders can be determined. In this case, the collision avoidance is of importance to be implemented to ensure the safe flight between aircraft.

III. Collision Avoidance Algorithm

The scheme of collision avoidance can be separated into two steps, including: 1) MDP-based path planning to produce an optimal trajectory, such that multiple threats can be avoided; 2) differential flatness-based algorithm with consideration of UAV dynamics to ensure the planned trajectory is feasible. In this work, the airspace is represented as a rectangular gridding system while each grid point is regarded as a waypoint. With the implementation of MDP, the optimal path of UAV is produced based on the current states of UAV and multiple threats. The collision avoidance is to select a set of waypoints for the UAV maximizing the safety distance. Following the MDP-based reference trajectory, UAV is required to avoid collision between multiple intruders, while the physical constraints are respected. A flatness-based path re-planner is proposed, which is considered as an appropriate method for the real-time calculation.

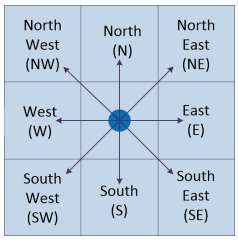

Fig. 3. The basic action of UAV

A. Markov Decision Process

MDP based on a stochastic model addresses the decision problem by discretizing the system into a finite number of states and actions [15]. MDP is defined as $M =< S, A, T, R >$, where S, A, T, and R denote the system's states, actions, transition function, and reward function, respectively. At each time step, the system takes an action $a \in A$ to move from a state $s \in S$ to $s' \in S$ and receives a real-valued reward $R(s, a)$. The reward function in this work is influenced by the position of the intruders and target. The goal of solving the collision avoidance problem is transformed to choose a sequence of actions that maximizes the expected total reward.

To generate an available trajectory, these units with respect to the flight requirement and environment needed to be defined. The parameters of MDP in this work are designed as:

- **States:** A flight range with $800m \times 800m$ is decomposed into a grid array by gridding the flight environment as $20m \times 20m$ square grid. Based on Cartesian coordinate method to identify grids, the initial point (0,0) is in the upper left corner of the grid array. The grid is then numbered in a positive direction on the x-axis (horizontally to the right) and in a positive direction on the y-axis (vertically going down).
- **Actions:** As shown in Fig. 3, UAV can move in eight directions, including: North, North-East, East, South-East, South, South-West, West, North-West.
- **Transition Function:** $S \times A \times S \rightarrow$ [0,1] is the state transition model for the discretized state space. The probability of transitioning from state s to s' after action a is written as:

$$P^a_{ss'} = Pr\{s_{t+1} = s' | s_t = s, a_t = a\}. \tag{10}$$

  The state-transition function specifies the next-state distribution given an action at a current state. Fig. 4 describes one scenario of probability transition. Assuming state s of a UAV at the center of this grid array, if the UAV is moving to the North with 0.9 probability, then the actions to NE and NW with 0.04 probability while to E and W with 0.01 probability, respectively. The UAV has eight probability transitions.
- **Reward Function:** The immediate reward when taking action a in state s is defined as:

$$R^a_{ss'} = E\{r_{t+1} | s_t = s, a_t = a, s_{t+1} = s'\}. \tag{11}$$

  The reward function denotes the reward of taking an action in a given state. It is convenient to define the reward function for each object separately. Comparing the safe distance with Euclidean distance between the UAV and the intruder, the reward of intruders is defined as:

$$r_1(s) = \sum_{i=1}^{n} (d_i - d_{safe}) \times w_1, \tag{12}$$

  where d_i is Euclidean distance between the UAV and the ith intruder, w_1 is a weighting factor for threats reward function. The reward function of the UAV to the goal is

$$r_2(s) = (D_{max} - D_{goal}) \times w_2, \tag{13}$$

  where D_{max} is the maximum sensing range of the sensor onboard the UAV, D_{goal} is Euclidean distance

between the UAV and the goal, w_2 is a weighting factor for the goal reward function. Eq. (13) implies a negative reward for the target not in the UAV's sensing range.

A policy π is a mapping from each state s and action a to the probability $\pi(s,a)$ of taking action a under the current state s. The time-accumulative value function under a policy π when starting from the state s is derived as:

$$\begin{aligned} V^{\pi}(s) &= E_{\pi}\{R_t|s_t=s\} \\ &= E_{\pi}\{\sum_{k=0}^{\infty}\lambda^k r_{t+k+1}|s_t=s\} \\ &= \sum_{a}\pi(s,a)\sum_{s'}P^a_{ss'}[R^a_{ss'}+\gamma V^{\pi}(s')], \quad (14) \end{aligned}$$

where $\gamma \in [0,1]$ is a discount factor to denote the effect of the future rewards on present decisions. $\pi(s,a)$ is the probability of taking action a in state s under policy π. $E_{\pi}\{\}$ denotes the expected value following policy π , t is a step time and s' is the next state followed by the current state s when action a is applied.

The expected value of taking action a in a state s under policy π is:

$$\begin{aligned} Q^{\pi}(s,a) &= E_{\pi}\{R_t|s_t=s,a_t=a\} \\ &= E_{\pi}\{\sum_{k=0}^{\infty}\lambda^k r_{t+k+1}|s_t=s,a_t=a\}. \quad (15) \end{aligned}$$

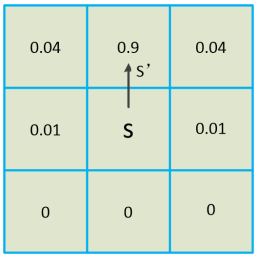

Fig. 4. The assignment of the transition probability

B. Policy Iteration

With respect to the initial policy and initial value functions, policy iteration is adopted to converge to an optimal policy. The policy iteration algorithm is carried out by the following steps:

1) Policy Evaluation: Policy evaluation is denoted as computing the state value function V^{π} for an arbitrary policy π. A sequence of approximate value funcions is obtained as:

$$\begin{aligned} V_{k+1}(s) &= E_{\pi}\{r_{r+1}+\gamma V_k(s_t+1)|s_t=s\} \\ &= \sum_{a}\pi(s,a)\sum_{s'}p^a_{ss'}[R^a_{ss}+\gamma V_k(s')]. \quad (16) \end{aligned}$$

The sequence $\{V_k\}$ can converge into V_{π} as $k \to \infty$ under the same initial conditions, by which the exsistence of V^{π} is ensured. To generate the successive approximation from V_k to V_{k+1}, the policy evaluation based operation is adopted to each state s: the new value obtained from the old value of the successor states of s' is substituted for the old value of s.

2) Policy Improvement: The process of making a new policy to obtain a higher value function is called policy improvement. Let π and π' be any pair of deterministic policies, if

$$Q^{\pi}(s,\pi'(s)) \geq V^{\pi}(s). \quad (17)$$

Moreover, the value under policy π', $V^{\pi'}(s)$, is not smaller than the value under policy π, $V^{\pi}(s)$. Then the greedy policy π' must be no worse than π. The greedy policy π' is given as:

$$\begin{aligned} \pi'(s) &= argmaxQ^{\pi}(s,a) \\ &= argmax\sum_{s'}P^a_{ss'}[R^a_{ss'}+\gamma V^{\pi}(s')]. \quad (18) \end{aligned}$$

Assuming that a better policy π_1 can be derived from the initial policy π_0 using V^{π}, an even better policy π_2 can be achieved by computing $V^{\pi'}$. A sequence of improving policies and value functions is produced as:

$$\pi_0 \xrightarrow{PE} V^{\pi_0} \xrightarrow{PI} \pi 1 \xrightarrow{PE} V^{\pi_1} ... \xrightarrow{PI} \pi^* \xrightarrow{PE} V^*,$$

where PE defines a policy evaluation and PI is a policy improvement. Owing to the finite number of MDP, this process must finally converge to an optimal policy and optimal value function in a finite number of iterations.

Based on the Bellman function [16], the optimal control policy π^* can be produced by the objective function, which maximizes the expected reward:

$$\pi^*(s) = argmax(R(s,a)+\lambda\sum_{s'\in S}(P(s'|s,a)V^*(s')), \quad (19)$$

where $V^*(s')$ is the optimal objective function as:

$$\begin{aligned} V^*(s) &= maxQ^{\pi^*}(s,a) \\ &= max\sum_{s'}P^a_{ss'}[R^a_{ss'}+\gamma V^{\pi}(s')]. \quad (20) \end{aligned}$$

C. The Integration of UAV Dynamics in the Planned Path

The optimal trajectory produced by MDP may require aggressive maneuvers for UAV to avoid intruders. However, UAVs have their physical constraints in practice, which are indicated by the maximal Euler angular rates, turning radius, and etc.. Based on the differential flatness approach, the profile of the trajectory is tuned to make sure that UAV follows the planned path smoothly. Due to the two-dimensinal flight scenario developed in this work, only the limit of the roll angle is taken into account.

A flatness-based nonlinear system is defined as:

$$\begin{cases} \dot{x} = f(x,u) \\ y = h(x) \end{cases}, \quad (21)$$

where $x \in R^n$ and $u \in R^m$ represent system states and control inputs, respectively. The system is flat only if flat outputs $F \in R^m$ fulfill the requirements such as:

$$\begin{cases} x = \Xi_1(F, \dot{F}, ..., F^{(n-1)}) \\ y = \Xi_2(F, \dot{F}, ..., F^{(n-1)}) \\ z = \Xi_3(F, \dot{F}, ..., F^{(n-1)}) \end{cases}, \quad (22)$$

where Ξ_1, Ξ_2, Ξ_3 are three smooth mapping and $F^{(i)}$ is the ith derivative of F. The parameterization of the flight envelope (here the roll angle is considered) in function of the flat outputs F is an important unit in the path smoothing problem: the flight envelope to be considered during a mission can be expressed in function of the desired trajectories. For the UAV model given in Eq. (2), the system is flat with flat outputs $F_1 = x, F_2 = y, F_3 = \psi$. Additionally, the parameterization of ϕ in the flat output is

$$\phi = atan\frac{V_a\dot{\psi}}{g}. \quad (23)$$

In this study, the reference trajectories are designed as:

$$F_i^*(t) = [1-(1+\omega_n t)e^{-\omega_n t}]R_i + R_i^0; \qquad i = 1, 2, 3 \quad (24)$$

where ω_n is the natural frequency. R_i $(i = 1, 2, 3)$ is the amplitude of x, y and ψ, respectively. R_i^0 is the initial position. According to Eq. (23) and the time derivative of Eq. (24), the roll angle ϕ^* is derived as:

$$\phi^* = atan(\frac{V_a R_3 \omega_n^2 t e^{-\omega_n t}}{g}). \quad (25)$$

According to [17], the relationship between the natural frequency ω_n and the settling time of the reference trajectory t_s can be approximated by:

$$\omega_n \approx \frac{5.83}{t_s}. \quad (26)$$

Thus, Eq. (25) can be rewritten as:

$$\phi^* = atan(\frac{5.83^2 V_a R_3 t e^{-\frac{5.83}{t_s}t}}{g t_s^2}). \quad (27)$$

And the time derivative of the roll angle is derived as:

$$\dot{\phi^*} = \frac{\frac{5.83^2}{t_s^2} R_3 V_a g e^{-\frac{5.83}{t_s}t}(1 - \frac{5.83}{t_s})}{\frac{5.83^4}{t_s^4} R_3^2 t^2 V_a^2 e^{-2\frac{5.83}{t_s}t} + g}. \quad (28)$$

To determine the time where the maximal roll angle is required, it is necessary to calculate the extrema of the roll angle by setting Eq. (28) to zero at $t = t_s/5.83$:

$$\phi_{ext}^* = atan(\frac{5.83 R_3 V_a e^{-1}}{g t_s}), \quad (29)$$

where the extrema denotes the maximal or the minimal of the function. It is of importance to check the value of the roll angle at the beginning and at the end of the mission. After comparing these three solutions, when $t = t_s/5.83$, the corresponding solution is the maximal one. Finally, to ensure that $|\phi^*| \leq \phi_{max}$, t_s has to satisfy:

$$t_s = \frac{5.83 R_3 V_a e^{-1}}{g tan(\phi_{max})}. \quad (30)$$

Based on the obtained solution t_s, the re-planned MDP-based optimal trajectory must be less than or equal to the maximal allowable constraints.

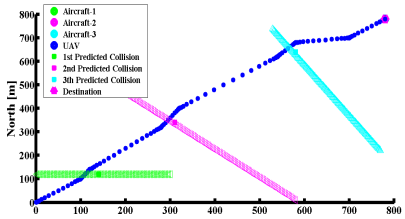

Fig. 5. The UAV - intruders collision avoidance scenario

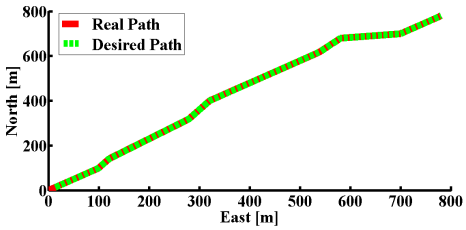

Fig. 6. Trajectory of the UAV

IV. Simulation and Results

The proposed algorithm is validated by numerical simulation in a PC that is configured with a 2.0 GHz Intel Core i7-4510U processor with 8 GB RAM. It is assumed that a UAV is cruising at $20m/s$, while three intruders are following the original trajectory at a speed of $15m/s$ without avoidance maneuver.

Fig. 5 shows the scenario in which three conflicts are detected, MDP-based algorithm produces a feasible trajectory for the UAV to avoid collisions. During the flight, the collision detection algorithm is adopted to determine when and where the collisions will occur. Based on the currently predicted conflict position and goal waypoint, the state, action, transition function and reward function of flight system are derived. MDP is therefore deployed to produce an optimal policy by calculating the policy interation. The blue path in Fig. 5 illustrates that the UAV with the developed S&A system successfully avoids three coming threats and reaches the destination finally.

Fig. 6 indicates the trajectory for the UAV using the differential flatness-based re-planning algorithm in the lateral direction. The plot shows the two-dimensional trajectory of the UAV (red line) with a desired trajectory (green line). As shown in Fig. 6, the red line closely follows the green line.

This result demonstrates that UAV can follow the desired trajectory quite well.

The tracking position errors in x direction and y direction, which are generated by comparing the real position with the desired position, are shown in Fig. 7 and Fig. 8, respectively. Fig. 7 exhibits that during the period of the entire flight, UAV follows x direction command within an error of ± 0.35 m, for 89% of the flight time. The peak error for x direction is approximately 0.55 m in this case. As can be seen from Fig. 8, y direction command is within an error of ± 0.8 m, for 85% of the flight time. The peak error for y direction is approximately 1.3 m. The sum of errors in x and y directions are illustrated in Fig. 9. The deviations from x and y directions are 26.8433 m and 67.0565 m within the entire duration. In summary, the simulation results exemplify the effectiveness of the developed S&A strategy.

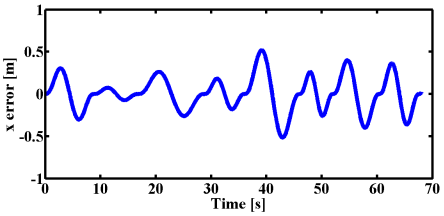

Fig. 7. x position error

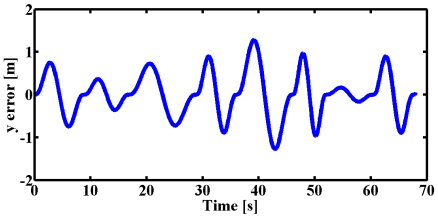

Fig. 8. y position error

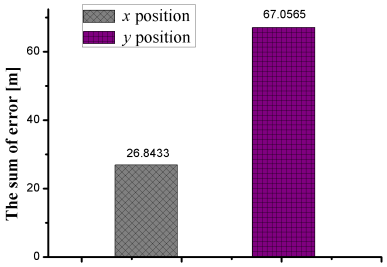

Fig. 9. The accumulative errors in x and y positions

V. Conclusion

In this study, a novel algorithm of trajectory optimization for UAVs Sense and Avoid (S&A) task in the presence of **multiple intruders** has been proposed. A unique feature of the proposed algorithm is to formulate the sense and the collision avoidance problem within a MDP framework for producing a desired trajectory in real-time. Considering the requirement of real-time computation, an optimal sequence of waypoints is selected to avoid collisions while minimizing the distance between targets. The feasible trajectory is smoothened by the flatness-based path planner without violating the physical limits of UAV. The effectiveness of the proposed algorithm is demonstrated through the numerical simulation. A possible future direction is to work on decreasing the number of states to reduce the computation costs and verifying the proposed algorithm in a real UAV testbed.

References

[1] X. Yu and Y. M. Zhang, "Sense and avoid technologies with applications to unmanned aircraft systems: Review and prospects," *Progress in Aerospace Sciences*, vol. 74, pp. 152-168, 2015.

[2] J. K. Kuchar and L. C. Yang, "A review of conflict detection and resolution modeling methods," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, no. 4, pp. 179-189, 2000.

[3] I. Hwang, J. Kim, and C. Tomlin, "Protocol-based conflict resolution for air traffic control," *Air Traffic Control Quarterly*, vol. 15, no. 1, pp. 1-34, 2007.

[4] A. Bayen, P. Grieder, G. Meyer, and C. Tomlin, "Lagrangian delay predictive model for sector based air traffic flow," *Journal of Guidance, Control, and Dynamics*, vol. 28, no. 5, pp. 1015-1026, 2005.

[5] J. H. Chuang and N. Ahuja, "An analytically tractable potential field model of free space and its application in obstacle avoidance," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 28, no. 5, pp. 729-736, 1998.

[6] M. A. Christodoulou and S. G. Kodaxakis, "Automatic commercial aircraft-collision avoidance in free flight: The three-dimensional problem," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 2, pp. 242-249, 2006.

[7] A. Bicchi and L. Pallottino, "On optimal cooperative conflict resolution for air traffic management systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1, no. 4, pp. 221-232, 2000.

[8] Z. T. Lian and A. Deshmukh, "Performance prediction of an unmanned airborne vehicle multi-agent system," *European Journal of Operational Research*, vol. 172, no. 2, pp. 680-695, 2006.

[9] A. G. Shem, T. A. Mazzuchi, and S. Sarkani, "Addressing uncertainty in UAV navigation decision-making," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 1, pp. 295-313, 2008.

[10] H. Bai, D. Hsu, M. J. Kochenderfer, and W. S. Lee, "Unmanned aircraft collision avoidance using continuous-state POMDPs," In *Proceedings of Robotics: Science and Systems*, Los Angeles, CA, USA, pp. 1-8, 2011.

[11] S. Temizer, M. J. Kochenderfer, L. P . Kaelbling, T. Lozano-Perez, and J. K. Kuchar. "Collision avoidance for unmanned aircraft using Markov decision processes," In *Proceedings of AIAA Guidance, Navigation, and Control Conference*, Toronto, Ontario, Canada, 2010.

[12] S. Ragi and E. K. P. Chong, "UAV path planning in a dynamic environment via partially observable Markov decision process," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 4, pp. 2397-2412, 2013.

[13] S. Ferrari and G. Daugherty,"Q-Learning approach to automated unmanned air vehicle (UAV) demining," *Journal of Defense Modeling and Simulation*, vol. 9, no. 1, pp. 83-92, 2012.

[14] J. W. Park, H. D. Oh, and M. J. Tahk, "UAV collision avoidance based on geometric approach," In *Proceedings of SICE Annual Conference*, Tokyo, pp. 2122-2126, 2008.

[15] M. Puterman, *Markov decision processes: discrete stochastic dynamic programming,* John Wiley & Sons, Hoboken, NJ, USA, 2005.

[16] R. S. Sutton and A. G. Barto, *Reinforcement learning: an introduction,* Cambridge, MA, MIT Press, 1998.

[17] A. Chamseddine, Y. M. Zhang, C. A. Rabbath, and D. Theilliol, "Trajectory planning and re-planning strategies applied to a quadrotor unmanned aerial vehicle," *Journal of Guidance, Control, and Dynamics*, vol. 35, no. 5, pp. 1667-1671, 2012.