UAV Collision Detection Algorithm based on Formulafor Fast Calculation

Jung Kyu Park[1] and Jaeho Kim[2,*]

[1]Department of Computer Software Engineering, Changshin University, Korea
[2]Dept. of Aerospace and Software Engineering, Gyeongsang National University, Korea
*Corresponding author. E-mail: jaeho.kim@gnu.ac.kr

Abstract: Collision Avoidance System (CAS) is known as a pre-collision system or a forward collision warning system, and research has first begun as a vehicle safety system. In this paper, we propose an algorithm for collision detection of UAV. The proposed algorithm uses a mathematical method and detects the collision of the UAV by modeling it in a two-dimensional plane. Using the mathematical modeling method, it is possible to determine the collision location of the UAV in advance. Experiments were conducted to measure the performance and accuracy of the proposed algorithm. In the experiment, we proceeded assuming three environments and were able to detect an accurate collision when the UAV moved. By applying the algorithm proposed in the paper to CAS, many collision accidents can be prevented. The proposed algorithm detects collisions through mathematical calculations. In addition, the movement time of the UAV was modeled in a 2D environment to shorten the calculation time.

1.   Introduction

A drone was used for the first time for military use, but recently the market is rapidly growing for agricultural, service, and leisure purposes. As the market has grown rapidly, many hobby and leisure drones that can be purchased at low cost have become popular. However, as many drones are used, many accidents are occurring. For example, drone battery explosions, fires, injuries caused by collisions with drones, and safety accidents such as falls and collisions due to deviation from the adjustment distance are rapidly increasing. For this reason, a Collision Avoidance System(CAS) is essential in drones or UAV(Unmanned Aerial Vehicle) [1-4].

CAS is known as a pre-collision system or a forward collision warning system, and research has first begun as a vehicle safety system [5-7]. CAS research has been conducted in various fields as well as in the automobile field, and research is rapidly progressing in the field of aircraft and UAV. Although there are various studies related to CAS, the basic algorithm of CAS is to create a safe travel path without causing a collision from a starting point to a target point. The CAS structure is composed of three elements: search, recognition, and avoidance. Among the elements, search and recognition are responsible for collision detection. And the evasion performs a collision avoidance mission. For UAVs to perform their CAS mission, it is necessary to identify and collaborate with the surrounding environment when first flying. At this time, the obstacle information of the environment can be collected in real time through the mounted sensor of the board [8-10].

Since the UAV moves in autonomous driving without a pilot, a powerful CAS must be used. UAVs are much cheaper to operate than human-controlled airplanes. It is also more suitable for performing dangerous or repetitive tasks that are difficult for humans to perform. Because of the above advantages, UAVs can be used for civilian tasks such as mapping and rescue, or for military missions such as patrol and search. To perform this specific task, multiple UAVs can be used at the same time, and each UAV must establish a mobile sensor network. In addition, UAVs must have communication, navigation, and collision avoidance functions [11-14].

In this paper, we propose an algorithm for collision detection of UAV. The proposed algorithm uses a mathematical method and detects the collision of the UAV by modeling it in a two-dimensional plane. Using the mathematical modeling method, it is possible to determine the collision location of the UAV in advance. An additional reason to use mathematical models is performance. When the UAV driving mode is changed from a simple navigation mode to a collision detection mode, real-time processing performance of collision detection is important. Using the proposed algorithm, it is possible to determine the time until the collision of the UAV, the travel distance, and the collision point.

2.   Proposed Method

UAV collision detection algorithm proposed in this paper models UAVs moving in three dimensions in two dimensions and expresses them with equations. Some assumptions are made for fast computation of the algorithm. Because it aims to detect collisions between UAVs, the UAV is moving and the initial and final positions are not considered. Figure 1 shows the situation in which collision occurs in two dimensions.
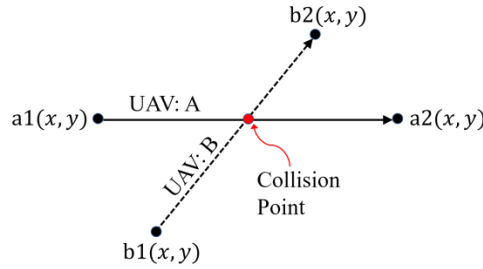
*Corresponding author:** Jaeho Kim
Dept. of Aerospace and Software Engineering, Gyeongsang National University, Korea
E-mail: jaeho.kim@gnu.ac.kr

Figure 1. Collision point of moving 2UAVs

2.1 Proposed Method

In this section, we describe the equations that form the basis of the algorithms we want to present in our study. This formula is used to detect collisions and calculate collision points.
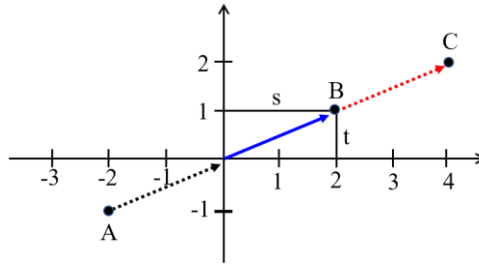

Figure 2. Movement equation with constant value

In Figure 2, assume that the UAV starts at point A and moves straight to point C. The UAV is currently located at point B, where $B = (s,t)$. Position B can be multiplied by a constant m to scale the position. As a formula, $P = m \cdot (s,t)$ can be expressed. That is, UAV is in a linear motion, so even if a specific coordinate value is multiplied by a constant m, the position becomes a point of a linear motion position. The constant m can be divided into four characters.

① $m > 1$: Point P quickly moves away from the starting point
② $m = 1$: Point P is in the current position (P==B)
③ $0 \leq m < 1$: Point P quickly approaches the origin (0,0)
④ $m < 1$: Point P represents a point in the direction that has passed before

Depending on the value of m, $m \cdot (s,t)$ is represented by a point on a straight line passing through the origin. This straight line passes through the origin, and if one point of the straight line is generalized to x and y, it is as the following Equation (1) that uses independent variables.

$$(x,y) = m \cdot (s,t)$$
$$= (m \cdot s, m \cdot t) \tag{1}$$

When this line passes through the point (q,r), the equation of the line can be expressed by Equation (2).

$$(x,y) = m \cdot (s,t) + (q,r) \tag{2}$$

Equation (2) using independent variables can be used to find the equation of a straight line passing through two points (s,t) and (q,r). At this time, the slope of the straight line can be described $m \cdot ((q,r) - (s,t))$. Using the slope obtained above, the equation for the straight line passing through two points can be expressed as Equation (3). In Equation (3), x can be calculated as $x = m(q - s) + q$, and y can be calculated as $y = m(r - t) + r$.

$$(x,y) = m \cdot ((q,r) - (s,t)) + (q,r) \tag{3}$$

2.2 Collision Point between UAVs

As shown in Figure 2, when the UAV is moving in a straight line, the equation of the straight line can be obtained using Equation (3).

$$P_A = m_A \cdot (a_2 - a_1) + a_1 \tag{4}$$
$$P_B = m_B \cdot (b_2 - b_1) + b_1 \tag{5}$$

P_A of equation (4) refers to the equation of the straight line of UAV A shown in Figure 1. Also, $P_A = (x,y)$ is a point on the straight line where UAV A moves. In Equation (4), x can be calculated as $x = m_A(a2.x - a1.x) + a1.x$, and y can be calculated as $y = m_A(a2.y - a1.y) + a1.y$. P_B of equation (5) refers to the equation of the straight line of UAV B shown in Figure 1. Also, $P_B = (x,y)$ is a point on the straight line where UAV B moves. In Equation (5), x can be calculated as $x = m_B(b2.x - b1.x) + b1.x$, and y can be calculated as $y = m_B(b2.y - b1.y) + b1.y$. m_A and m_B of equations (4), (5) mean the scale of point P_A and P_B.

To detect collisions between UAVs, the coordinate values of the two UAVs must be continuously monitored. That is, when two UAVs have the same coordinates ($P_A = P_B$), a collision occurs. Assuming that the coordinates of the two UAVs are the same, we can organize the expressions by $P_A.x = P_B.x$ and $P_A.y = P_B.y$. Summarizing the previous equation, the equation is (6).

$$m_A(a2.x - a1.x) + a1.x = m_B(b2.x - b1.x) + b1.x$$
$$m_A(a2.y - a1.y) + a1.y = m_B(b2.y - b1.y) + b1.y \tag{6}$$

Using the two equations in Equation 6, m_A and m_B can be obtained. In Equation 7, m_A and m_B are summarized.

$$m_A = \frac{(a2.x - a1.x)(b1.y - a1.y) - (a2.y - a1.y)(b1.x - a1.x)}{(a2.y - a1.y)(b2.x - b1.x) - (a2.x - a1.x)(b2.y - b1.y)}$$
$$m_B = \frac{(b2.x - a1.x)(b1.y - a1.y) - (b2.y - b1.y)(b1.x - a1.x)}{(a2.y - a1.y)(b2.x - b1.x) - (a2.x - a1.x)(b2.y - b1.y)} \tag{7}$$

In order to determine whether two UAVs collide, we need to make sure that both UAVs converge to one point. That is, it is possible to determine whether collision occurs by evaluating whether, m_A and m_B are values between 0 and 1. That is, when the two conditions in Equation (8) are satisfied, the collision point occurs in the UAV moving line.

$$0 < m_A < 1$$
$$0 < m_B < 1 \tag{8}$$

Although the predicted point of collision can be predicted by Equation (8), additional considerations must be taken to actually detect the collision. An additional consideration is the travel time of the UAV. The time at the current location of the two UAVs is compared, and if the difference is smaller than the threshold, collision avoidance should be performed. It is assumed that the current time of each UAV is t_A and t_B. When the difference between the two UAVs' current time is less than the threshold value, a collision occurs between the UAVs. At this time, the time difference must be less than the threshold value t_H and greater than 0. In this case, the threshold value should consider the operating environment of the UAV and the hardware characteristics of the UAV.

2.3 Collision Detection

Algorithm 1 describes a collision detection technique based on the section 2.2. This algorithm is performed when the UAV is on the move. Periodically, m_A and m_B values are obtained, and when these values are between 0 and 1, t_A and t_B values are additionally obtained. When the difference between the values of t_A and t_B is less than t_H, a collision avoidance operation is performed in anticipation of a collision.

	Algorithm 1: Collision Detection
1	**let,** t_H is a threshold of time
2	**while** *moving UAVs* **do**
3	calculate the m_A and m_B;
4	**if** ($0 < m_A < 1$) && ($0 < m_B < 1$)) **then**
5	get current time of two UAVs, t_A and t_B;
6	**if** ($0 \le |t_A - t_B| < t_H$) **then**
7	perform the collision avoidance;
8	**end**
9	**end**
10	**end**

3. Results and Discussion

In order to verify the proposed algorithm, a simulation experiment was conducted. The experiment was performed assuming that the UAV was moving in a 2D environment. The experiment was divided into three categories. 1) When there is no collision point, 2) When an intersection occurs and a collision is expected. 3) When an intersection occurs but there is no collision. As such, experiments were performed based on three scenarios, and the detailed experiment environment is shown in Table 1.

Table 1: Data set for experiment

		Category 1	Category 2	Category 3
UAV 1	Start position	(-10, 20)	(-10, 0)	(-10, 0)
	Destination position	(10, 20)	(10, 0)	(10, 0)

	Speed (m/s)	40km/h	40km/h	20km/h
UAV 2	Start position	(-10, 10)	(-10, -10)	(-10, -10)
	Destination position	(10, 10)	(10, 10)	(10, 10)
	Speed (m/s)	40km/h	40km/h	40km/h

Based on Table 1, experiments were performed by category, and as a result, Figures 3, 4, and 5 were obtained. Based on Table 1, experiments were performed by category, and as a result, Figures 3, 4, and 5 were obtained. In Figure 3, the two UAVs move horizontally at the same speed, so there are no intersections and collisions. That is, when ma and mb are calculated by Equation 7, the denominator becomes 0, so m_A and m_B cannot be obtained. This means that when two UAVs move, they do not intersect at the time t_1 and t_2.

Figure 4 shows the results of Category 1. Like Category 1, two UAVs are moving at the same speed. At this time, if the values of m_A and m_B are obtained using Equation (7), each is 0.5, which is greater than 0 and less than 1. It can be seen that a collision occurs at time t_1 at (0,0). Figure 5 shows the results of Category 3. The two UAVs are moving at different speeds. At this time, if the values of ma and mb are obtained using Equation 7, each is 0.5, which is the same as the result in Figure 3. Calculating the intersection point of the UAV is (0,0), but no collision occurs. That is, because two UAVs move at different speeds, each UAV is sufficiently far away at the time t_1
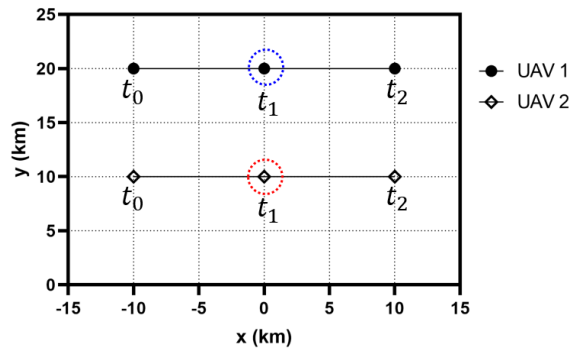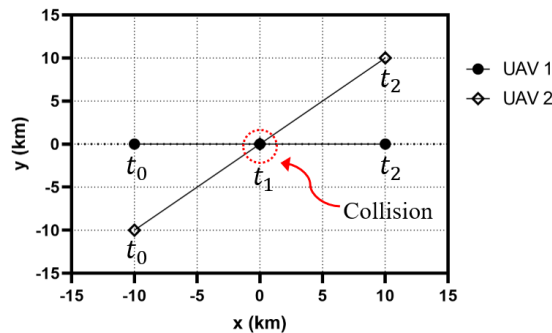

Figure 3. Result of category 1
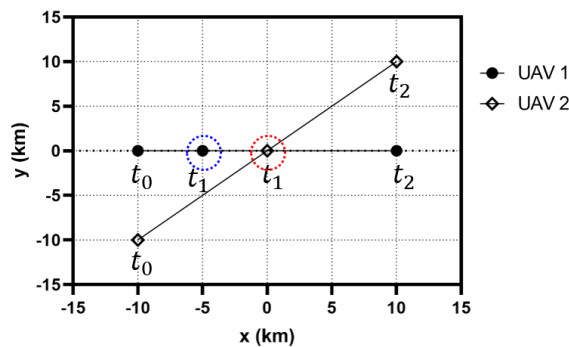

Figure 4. Result of category 2

Figure 5. Result of category 3

Jung Kyu Park[1] and Jaeho Kim[2,*]

4. Conclusions

In this paper, we propose an algorithm to detect collisions that may occur when UAVs move. The proposed algorithm detects collisions through mathematical calculations. In addition, the movement time of the UAV was modeled in a 2D environment to shorten the calculation time. To use the collision detection algorithm, two values require the current location and time of the UAV. In order to evaluate the proposed algorithm, a simulation experiment was conducted, and as a result, it was possible to accurately determine whether or not a collision occurred. Future studies will further refine the UAV crash situation and add more variables for calculation.

5. Acknowledgements

6. References

1. C. Xu, X. Liao, J. Tan, H. Ye, and H. Lu, (2020) Recent Research Progress of Unmanned Aerial Vehicle
2. Regulation Policies and Technologies in Urban Low Altitude. *IEEE Access 8*, 74175–74194.
3. Y. Ma, R. Zhao, E. Liu, Z. Zhang, and K. Yan, (2019) A Novel Method for Measuring Drogue-UAV Relative Pose in Autonomous Aerial Refueling Based on Monocular Vision. *IEEE Access* 7, 139653–139667.
4. J. K. Park and J. Kim. (2019) Collision Avoidance Method for UAV Using A* Search Algorithm. Proc. Of Advances in Intelligent, Interactive Systems and Applications. Hong Kong, 186–193.
5. G. Li, C. Zhuang, Q. Wang, Y. Li, X. Xu, and W. Zhou. (2019) A UAV Real-time trajectory Optimized Strategy for Moving Users. In: 2019 11th International Conference on Wireless Communications and Signal Processing (WCSP). Xi'an, China, 1–6.
6. Z. Chen, C.Wang, H.Wang, P. Li, Y. Li, and X.Wu. (2018) Object Detection for UAV Grasping: Solution and Analysis. Proc Of 2018 IEEE International Conference on Information and Automation (ICIA). Wuyishan, China, 1078–1083.
7. Ranjan, B. Panigrahi, H. K. Rath, P. Misra, and A. Simha. (2018) LTE-CAS: LTE-based criticality aware scheduling for UAV assisted emergency response. Proc of IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). HI, USA, 894–899.
8. Y. Sun, Z. Ding, and X. Dai, (2019) A User-Centric Cooperative Scheme for UAV-Assisted Wireless Networks in Malfunction Areas. *IEEE Transactions on Communications* 67(12), 8786–8800.
9. J. N. Yasin, S. A. S. Mohamed, M. Haghbayan, J. Heikkonen, H. Tenhunen, and J. Plosila, (2020) Unmanned Aerial Vehicles (UAVs): Collision Avoidance Systems and Approaches. *IEEE Access* 8, 105139–105155.
10. She, C. Liu, T. Q. S. Quek, C. Yang, and Y. Li, (2019) Ultra-Reliable and Low-Latency Communications in Unmanned Aerial Vehicle Communication Systems. *IEEE Transactions on Communications* 67(5), 3768–3781.
11. H. Kim and J. Ben-Othman, (2018) A Collision-Free Surveillance System Using Smart UAVs in Multi Domain IoT. *IEEE Communications Letters* 22(12), 2587–2590.
12. H. V. Abeywickrama, B. A. Jayawickrama, Y. He, and E. Dutkiewicz. (2017) Algorithm for energy efficient inter-UAV collision avoidance. Proc Of 2017 17th International Symposium on Communications and Information Technologies (ISCIT), QLD, Australia, 1–5.
13. J. Qu, X. Gan, G. Zhao, and S. Chen. (2019) Research on Detection and Warning of Air Threat Situation for UAV Collision Avoidance. Proc. Of 2019 International Conference on Intelligent Computing, Automation and Systems (ICICAS). Chongqing, China, 226–229.
14. M. S. Krämer and K.-D. Kuhnert. (2018) Multi-Sensor Fusion for UAV Collision Avoidance. Proc Of 2018 2nd International Conference on Mechatronics Systems and Control Engineering. ICMSCE 2018. Amsterdam, Netherlands, 5–12.
15. W. Zhang, D. Feltner, J. Shirley, D. Kaber, and M. S. Neubert, (2020) Enhancement and Application of a UAV Control Interface Evaluation Technique. *ACM Transactions on Human-Robot Interaction* 9(2), 1-20.