

A Dual-path Adaptive Selection Method for Unmanned Aerial Vehicles

Sooyoung Moon*

*Researcher, Convergence and Open Sharing System Immersive Media, Pai Chai University, Daejeon, Korea

[Abstract]

Path planning methods play a major role (for example, collision avoidance and time/cost reduction) in unmanned aerial vehicles (UAV) based services. Path planning methods for ground robots cannot be directly applied to UAVs since they only consider 2D free space. A UAV should be able to fly over or fly around obstacles to reduce flight distance. The proposed path planning method is able to compute a 'fly-over' path with consideration of location, width and height of obstacles. The proposed path planning method is more efficient than existing Wavefront method when the obstacle height is less than half the width of the obstacle.

▶ **Key words:** Unmanned aerial vehicles (UAVs), Path planning, Collision avoidance, Wavefront algorithm, Software-in-the-loop simulation (SITL)

[요 약]

무인기 응용 서비스에서는 충돌 회피 및 운용 비용 절감을 위한 경로 계획 방법이 핵심적인 역할을 담당한다. 지상 로봇 분야에서 사용되어 온 기존 경로 계획 방법은 2차원 공간을 가정하여 장애물 측면 우회 경로만 생성하므로 3차원 공간에서 운용되는 무인기에게는 비효율적이다. 본 논문에서는 장애물의 위치, 너비, 높이 특성을 고려하여 상공 통과 또는 측면 우회 중 최적 경로를 지능적으로 선택하는 이중 경로 적응 선택 기법(Dual-Path Adaptive Method, DPAM)을 제안한다. 제안된 방법은 장애물 상공 통과 경로와 측면 우회 경로를 각각 생성한 후 비행 거리를 비교하여 더 짧은 경로를 선택한다. PX4 기반 SITL 시뮬레이션을 통한 성능 평가 결과, 제안 방법은 장애물 높이가 너비의 절반 미만일 때 기존 Wavefront 알고리즘 대비 10-20%의 비행 거리 단축 효과를 나타냈다. 구체적으로 너비 30m, 50m, 70m 장애물에 대해 높이가 상대적으로 낮은 경우 각각 약 14%, 19%, 25%의 비행 거리 절감 효과를 확인하였다.

▶ **주제어:** 무인기, 경로 계획, 충돌 회피, Wavefront 알고리즘, 시뮬레이션

• First Author: Sooyoung Moon, Corresponding Author: Sooyoung Moon
*Sooyoung Moon (symoon@pcu.ac.kr), Convergence and Open Sharing System Immersive Media, Pai Chai University
• Received: 2025. 09. 22, Revised: 2025. 10. 11, Accepted: 2025. 10. 20.

I. Introduction

무인기는 조종자가 탑승하지 않고 임무를 수행하는 비행체를 의미한다 [1]. 최근 센서, 통신, 카메라, 비행 제어 SW/HW 등 관련 기술들의 발달로 항공촬영, 측량 등 다양한 분야에서 무인기가 폭넓게 활용되고 있다 [2-4]. 이와 같은 무인기 서비스에 있어 경로 계획 방법은 다음과 같은 이유로 매우 중요하다 [5]

- 비가시권 비행 (Beyond the Visual Line Of Sight : BVLOS)에서는 조종자에 의한 수동 조종이 불가하다.
- 무인기는 비행 중에 지형지물과 충돌하지 않아야 한다.
- 무인기 서비스 제공 비용 절감을 위해 비행 시간·거리 면에서 효율적인 경로를 제안할 수 있어야 한다.

최근 무인기 3차원 경로 계획 연구는 크게 강화학습 기반 방법 [11], 메타휴리스틱 최적화 기법 [12, 14], 그리고 에너지 효율 최적화 방법 [15] 등으로 분류할 수 있다. Wang et al. [11]은 미지의 복잡한 환경에서 강화학습 기반 충돌 회피 알고리즘(APPA-3D)을 제안하였으며, 입자 군집 최적화(PSO)를 개선한 연구 [12]에서는 게임 이론을 활용하여 다중 UAV 경로 계획 성능을 향상시켰다. Zhang et al. [13]은 고밀도 장애물 환경을 위한 RFA-star 알고리즘을 제시하였고, Levy Pelican 최적화 알고리즘 [14]은 산악 환경에서의 경로 계획 문제를 다루었다. 또한 에너지 최적화 관점의 연구 [15]에서는 UAV의 제한된 배터리 용량 문제를 해결하기 위한 궤적 계획 방법을 제안하였다.

그러나 이러한 최신 연구들은 대부분 복잡한 학습 과정이 필요하거나 [11], 계산 복잡도가 높은 메타휴리스틱 기법 [12, 14]을 사용하며, 특정 환경(미지의 환경, 고밀도 장애물, 산악 지형 등)에 특화되어 있다 [11, 13, 14]. 반면, 사전에 장애물 정보가 주어진 일반적인 임무 환경에서는 계산 효율성과 실시간 적용 가능성이 더 중요할 수 있다.

지상 로봇 기반 서비스에서 사용되는 경로 계획 방법은 대부분 2차원 공간을 가정하므로 무인기 서비스에는 적합하지 않다. 만약 지상 로봇 경로 계획 알고리즘을 무인기에 적용하면 장애물이 존재할 경우 항상 장애물 측면으로 우회하는 경로만 선택하게 되어 비행 시간·거리 면에서 비효율적인 상황이 발생하게 된다. 따라서 무인기 임무 수행을 위한 경로 계획 방법은 다음과 같은 요구사항을 만족해야 한다.

- 무인기는 장애물의 위치와 너비, 높이를 고려하여 장애물이 위치한 지점의 상공으로 통과하거나 장애물 측면으로 우회할 수 있어야 한다.

본 논문에서는 위 요구사항을 만족하는 무인기 경로 계획 방법을 제안한다. 제안 경로 계획 방법은 주어진 (출발지, 목적지) 쌍에 대해 장애물의 위치, 너비, 높이를 고려하여 비행 경로를 계산한다. 만약, 출발지와 목적지 사이의 직선 경로 (Line Of Sight) 상에 장애물이 위치하는 경우, 제안 경로 계획 방법은 최대/최소 비행 고도, 장애물의 넓이, 높이 등을 고려하여 2개의 가능한 비행 경로, 즉 (i) 해당 장애물의 상공으로 통과하는 경로와 ii) 측면으로 우회하는 경로를 계산하고 예상 이동 거리가 더 짧은 경로를 선택한다. 제안 경로 계획 방법과 기존 경로 계획 방법 (즉, Wavefront 알고리즘) 간 성능 비교를 위해 본 논문에서는 PX4 [6] 기반 software-in-the-loop (SITL) 시뮬레이션을 수행하였으며 장애물의 너비·길이에 비해 높이가 상대적으로 낮은 경우, 제안 기법이 Wavefront 알고리즘에 비해 더 짧은 비행 거리를 갖는 경로를 제안할 수 있다는 것을 확인하였다.

본 논문의 구성은 다음과 같다. 2장에서는 기존 경로 계획 방법인 Wavefront 알고리즘을 요약하고 3장에서는 무인기 서비스에 필요한 경로 계획 문제를 정의한다. 4장에서는 제안된 경로 계획 방법을 단계별로 설명하고 5장에서는 시뮬레이션을 통한 성능 평가 결과를 제시한다. 6장에서는 결론을 내리고 추후 연구 계획을 설명한다.

II. Background

1. Wavefront algorithm

Wavefront 알고리즘 [7, 8]은 A. Zelinsky 에 의해 제안되었으며 무인기가 임무를 수행하는 영역의 맵 데이터가 존재하고 모든 장애물의 위치가 고정이라는 가정 하에 출발지에서 목적지까지 최단 경로 또는 커버리지 경로 (coverage path) [9] 계산이 가능하다. Wavefront 알고리즘은 다음과 같이 3단계로 진행된다.

- 1) 전체 영역을 격자로 분할하고 출발지와 목적지 식별
- 2) 목적지에서 너비 우선 탐색 (BFS) 알고리즘 수행
- 3) 출발지 셀에서 목적지 셀까지 경로 계산 -

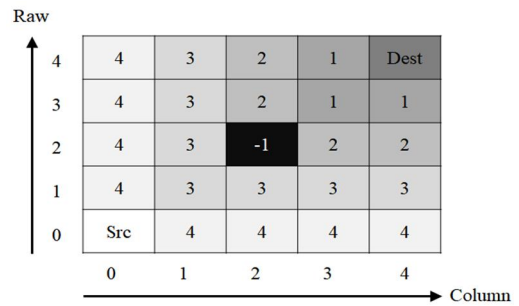
최단 경로 또는 커버리지 경로 계산

2)에서, Wavefront map 구성 영역을 구성하는 각 셀은 양의 정수인 레이블 (Label)을 할당 받으며 이는 해당 셀에서부터 목적지까지 이동하기 위해 거쳐야 하는 셀의 최소 개수를 의미한다. 그러므로 각 셀의 레이블은 탐색 트리 상에서 부모 셀의 레이블 값에 1을 더한 값으로 결정된다. Table 1은 Wavefront 경로 계획 알고리즘의 동작 과정을 나타낸다.

Table 1. Wavefront Path planning algorithm

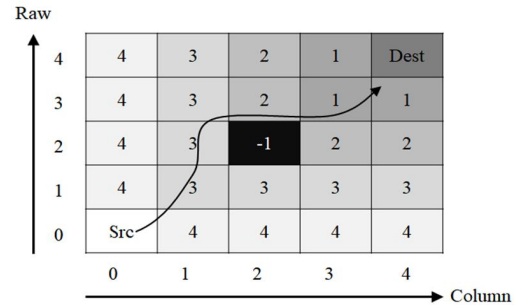
<p>Inputs:</p> <ol style="list-style-type: none"> 1) M: Map data 2) S: Source, 3) D: Destination, 4) O: Obstacle data
<p>Output:</p> <p>P: A path from source S to destination D $P = \langle \text{Cell}_1(=S), \text{Cell}_2, \dots, \text{Cell}_N(=D) \rangle$</p>
<p>$WaveFront(M, S, D, O)$:</p> <ol style="list-style-type: none"> 1. Wavefront Map Creation <ol style="list-style-type: none"> ① Divide M into a grid of cells of size $m \times n$, where each cell is denoted as (i, j) ② Determine the grid cells corresponding to S (source) and D (destination) ③ Mark obstacle cells based on O (obstacle data) by assigning them a label of -1 ④ Assign the destination cell a label of 0 ⑤ Perform a Breadth-First Search (BFS) starting from the destination cell to propagate distance labels <ol style="list-style-type: none"> i) Initialize an empty queue Q ii) Mark all non-obstacle cells as 'undiscovered' iii) Set label[D] = 0, Enqueue D into Q iv) While Q is not empty: Dequeue the front cell c from Q For each neighbor n of c: If n is 'undiscovered' and not an obstacle: Set label[n] = label[c] + 1 Mark n as 'discovered' Enqueue n into Q 2. Path Computation <ol style="list-style-type: none"> ① Initialize path P with the source cell: $P \leftarrow [S]$ ② Set current $\leftarrow S$ ③ Repeat until current $\neq D$ Among all neighbors (w/o obstacles) of current: Select the neighbor n with the smallest positive label Append n to path P Set current $\leftarrow n$ 3. Return the computed path P

Table 1에서 Wavefront 알고리즘의 입력은 1) 맵 데이터 (M), 2) 출발지 위치 (S), 3) 목적지 위치 (D), 그리고 맵 영역에 속한 장애물의 위치 정보 (O)이다. 또한 해당 알고리즘의 출력은 출발지 S 에서 목적지 D 까지의 이동 경로 (P)이다. Wavefront 알고리즘은 크게 1) Wavefront 맵 생성과 2) 경로 계산의 순서로 진행된다. Wavefront 맵 생성 단계에서는 먼저 맵 영역을 격자 형태로 분할하고 출발지 셀과 목적지 셀을 계산한다. 이후 장애물 데이터로부터 각 장애물의 위치에 해당하는 셀의 레이블을 -1 으로 설정한다. 이는 해당 셀로 이동할 수 없음을 나타내기 위함이다. 또한 목적지 셀의 레이블을 0 으로 할당한 후 너비 우선탐색을 통해 목적지 셀에서부터 모든 셀의 레이블을 계산한다. 맵 생성과 그에 따른 레이블 계산이 완료되면 출발지에서 목적지까지의 경로를 계산한다. 먼저 출발지 셀만 포함하는 임시 경로 변수를 생성한 후, 현재 셀의 이웃 셀들 중 레이블 값이 가장 작은 셀을 선택하고 이를 목적지 셀을 찾을 때까지 반복하면 출발지에서 목적지까지의 최단 경로가 도출된다. Fig.1은 Wavefront 알고리즘에 의한 최단 경로 계산 과정의 예를 나타낸다.



The cell with obstacle: (2, 2)

(a) Wavefront map creation



Path $P = \langle (0, 0), (0, 1), \dots, (3, 4), (4, 4) \rangle$

(b) Path computation

Fig. 1. Path Computation based on Wavefront algorithm

Fig. 1. (a) 에서 Wavefront map 계산을 위해 영역이 5-by-5 격자로 분할된 후 출발지 셀의 위치 (0, 0), 목적지 셀의 위치 (4,4) 가 각각 계산된다. 또한 목적지 셀에 레이블 값 (0) 을 할당하고 너비 우선 탐색을 통해 각 셀의 레이블 값을 계산한다. 장애물이 위치한 셀의 경우 -1 값이 할당된다.

Fig. 1. (b)에서는 변수 P에 출발지 셀 정보를 추가한 후 출발지 셀의 이웃 셀들 중 가장 레이블 값이 작은 셀 (1, 1) 을 변수 P에 추가한다. 변수 P에 목적지 셀 정보가 추가될 때까지 위 과정을 반복한다. 경로 계산이 완료되면 해당 경로 정보가 반환된다.

Wavefront 알고리즘은 너비 우선 탐색 알고리즘을 기반으로 하므로 출발지에서 목적지까지 최단 경로가 존재한다면 이를 반드시 찾을 수 있다는 장점이 있는 반면 영역의 크기가 증가함에 따라 시간 복잡도와 공간 복잡도가 지수적으로 증가한다는 단점을 갖는다. 또한 2차원 공간을 가정하므로 출발지에서 목적지 사이에 장애물이 존재할 경우 항상 해당 장애물의 측면으로 우회하는 경로만을 선택하게 되어 경우에 따라 해당 장애물의 상공을 통과하는 경로에 비해 이동 거리 면에서 상대적으로 비효율적인 경로를 선택하게 될 가능성이 존재한다.

2. Recent Advances in UAV 3D Path Planning

최근 UAV 3D 경로 계획 연구는 다양한 접근 방법을 통해 발전하고 있다. 강화학습 기반 접근법에서 Wang et al. [11]의 APPA-3D는 동적 보상 함수와 최적화된 행동 탐색 전략을 통해 미지의 복잡한 환경에서 자율적 경로 계획을 수행한다. 이 방법은 환경 정보가 사전에 주어지지 않은 상황에서 강점을 보이지만, 학습에 상당한 시간과 계산 자원이 필요하다는 한계가 있다.

메타휴리스틱 최적화 기법 연구로는 개선된 PSO [12], Levy Pelican 최적화 알고리즘 [14], 그리고 RFA-star 알고리즘 [13]이 있다. PSO 기반 방법 [12]은 게임 이론의 내쉬 균형을 활용하여 다중 UAV의 경로 최적화에서 우수한 성능(32.57% 수렴 시간 개선)을 보였으나, 반복적 최적화 과정으로 인한 계산 부하가 크다. RFA-star [13]은 고밀도 장애물 환경에서 특징 주의 메커니즘을 활용하여 안정적인 경로를 생성하며, Levy Pelican 알고리즘 [14]은 산악 환경의 복잡한 지형에 대응하도록 설계되었다.

에너지 효율 최적화 관점에서 [15]는 "에너지 맵" 기반 궤적 계획을 통해 제한된 배터리 용량 문제를 다루었다. 이는 장거리 임무 수행 시 매우 중요한 요소이나, 본 연구에서 다루는 짧은 거리의 장애물 회피 상황에서는 비행 거리 자체가 주요 최적화 목표가 된다.

이러한 최신 연구들과 비교하여 본 논문에서 제안하는 방법은 다음과 같은 차별점을 갖는다:

- 사전 학습이나 복잡한 최적화 과정 없이 실시간 경로 계획 가능
- 장애물 정보가 주어진 환경에서 계산 효율성 우수
- 간단한 비용 비교를 통한 명확한 의사결정 구조
- 기존 Wavefront 알고리즘을 확장하여 구현 용이성 제공

III. The Proposed Scheme

1. Problem definition

1.1 Definitions and assumptions

본문에서 사용된 주요 용어들은 다음과 같다.

- 영역 (Area) : 무인기가 임무를 수행하는 지리적 범위
- 셀 (Cell) : 영역을 격자(grid) 형태로 나눈 기본 단위
- 출발지 셀 (Start Cell) : 출발지 좌표를 포함하는 셀
- 목적지 셀 (Destination Cell) : 목적지 좌표를 포함하는 셀
- 경로 (Path) : 출발지에서 목적지까지 연결된 셀 목록
- 웨이포인트 (Waypoint) : 경로 내 셀들의 위치 좌표 (x, y, z)
- 경로 비용 (Cost of a path): 무인기가 제안 경로를 따라 비행했을 때의 총 비행 거리
- 최대 비행 고도 (Maximum altitude): 비행 가능한 최대 고도 (지표면 기준)
- 최소 비행 고도 (Minimum altitude): 무인기가 특정 셀의 상공으로 통과할 경우, 충돌 방지를 위해 유지해야 하는 최소 고도

본 논문의 주요 가정사항은 다음과 같다.

1. 사전에 무인기 임무 수행 영역의 맵 데이터가 제공된다
2. 무인기 임무 수행 영역 내 장애물들의 위치와 크기, 높이 정보가 주어진다
3. 장애물들의 위치는 임무 수행 기간 동안 변경되지 않는다
4. 지형지물 충돌 방지를 위한 최대/최소 비행고도가 존재한다

2. Dual-path adaptive method

2.1 Limitation of existing path planning methods

Wavefront 알고리즘은 2차원 공간을 가정하므로 출발지에서 목적지 사이에 장애물이 위치할 경우 해당 장애물

을 우회하는 경로만 선택 가능하다. 그러나 무인기 임무 수행 중에는 장애물의 넓이에 비해 높이가 상대적으로 낮을 경우 해당 장애물의 상공으로 통과하는 경로가 더 효율적인 상황이 존재할 수 있다. 위와 같은 이유로 본 논문에서는 장애물의 위치와 넓이, 높이를 고려하여 상황에 따라 해당 장애물의 상공으로 통과하거나 측면으로 우회할 수 있는 Wavefront 기반 경로 계획 방법을 제안한다.

2.2 Proposed method

Table 2는 제안된 경로 계획 알고리즘을 나타낸다.

Table 2. Dual-path adaptive method

<p>Inputs:</p> <ol style="list-style-type: none"> 1) M: Map data 2) S: Source, 3) D: Destination, 4) O: Obstacle data 5) A_{Max}: Maximum altitude 6) A_{Min}: Minimum altitude
<p>Output:</p> <p>P: A path from source S to destination D $P = \langle \text{Cell}_1(=S), \text{Cell}_2, \dots, \text{Cell}_N(=D) \rangle$</p>
<p><i>Dual-pathAdaptiveMethod</i>($M, S, D, O, A_{Max}, A_{Min}$):</p> <ol style="list-style-type: none"> 1. Wavefront Map Creation <ol style="list-style-type: none"> ① Divide M into a grid of cells of size $m \times n$, where each cell is denoted as (i, j) ② Determine the grid cells corresponding to S (source) and D (destination) ③ Make a list of the cells with obstacles (CWO) ④ Assign the destination cell a label of 0 ⑤ Perform a Breadth-First Search (BFS) starting from the destination cell to propagate distance labels <ol style="list-style-type: none"> i) Initialize an empty queue Q ii) Mark every cell in the area as 'undiscovered' iii) Set label[D] = 0, Enqueue D into Q iv) While Q is not empty: <ol style="list-style-type: none"> Dequeue the front cell c from Q For each neighbor n of c: <ol style="list-style-type: none"> distance = label[c] + 1 If n is 'undiscovered' and $n \in CWO$, If $\text{Obstacle_height} < (A_{Max} - A_{Min})$: label[n] = distance Mark n as 'discovered' Enqueue n into Q Else: label[n] = -1 Mark n as 'unreachable'

<p>If n is 'undiscovered' and $n \notin CWO$, label[n] = distance Mark n as 'discovered' Enqueue n into Q</p>
<p>2. 'Fly-over' path (P_{FO}) computation</p> <ol style="list-style-type: none"> ① Initialize path P with the (source cell, min. alt.): $P_{FO} \leftarrow [(S, A_{MIN})]$ ② Set current $\leftarrow S$ ③ Repeat until current $\neq D$ <ol style="list-style-type: none"> Among all neighbors (not 'unreachable') of current: Select the neighbor n with the smallest positive label if $n \in CWO$ then altitude = obstacle_height(next) + A_{MIN} else altitude = A_{MIN} Append (n, altitude) to P_{FO} Set current $\leftarrow n$
<p>3. 'Fly-around' path P_{FA} computation</p> <p>Among all neighbors (w/o obstacles) of current: Select the neighbor n with the smallest positive label Append (n, A_{MIN}) to path P Set current $\leftarrow n$</p>
<p>4. Path selection</p> <p>if $\text{cost}(P_{FO}) \leq \text{cost}(P_{FA})$ then $P \leftarrow P_{FO}$ else $P \leftarrow P_{FA}$ return P</p>

Table 2는 제안된 이중 경로 선택형 경로 계획 알고리즘 (Dual-path Adaptive Method; 이하 DPAM) 을 나타낸다. 제안된 알고리즘의 입력은 1) 맵 데이터 (M), 2) 출발지 위치 (S), 3) 목적지 위치 (D), 4) 장애물 위치 정보 (O), 5) 최대 비행 고도 (A_{Max}) 및 6) 최소 비행 고도 (A_{Min})이다. DPAM 알고리즘의 출력은 출발지-목적지 경로를 구성하는 (셀 ID, 비행 고도) 목록이다. DPAM 알고리즘은 크게 1) Wavefront 맵 생성, 2) 장애물 상공 통과 경로 (Fly-over path) 계산, 3) 장애물 우회 경로 (Fly-around path) 계산, 그리고 4) 두 경로의 비용 비교 및 경로 선택의 순서로 진행된다. Wavefront 맵 생성 단계에서 DPAM 알고리즘은 맵 격자 분할 및 출발지/목적지 셀 식별을 진행하며 이는 기존 알고리즘과 동일한 방식으로 진행된다. 이후 DPAM 알고리즘은 장애물을 포함하는 셀 목록 (CWO) 을 생성하며 이는 각 셀의 레이블을 계산하고 이후 경로 계산 단계들에서 활용하기 위함이다. 레이

블 계산 과정에서 임의 셀이 CWO에 포함된 경우, 다음과 같이 조건에 따라 처리된다.

1) 해당 장애물의 높이가 ($A_{MAX} - A_{MIN}$)보다 작으면 상공 통과가 가능하므로 정상적으로 레이블을 계산하고 비행 경로에 포함될 수 있도록 'discovered'로 상태를 설정한다

2) 만약 장애물의 높이가 ($A_{MAX} - A_{MIN}$)보다 크면 최대 비행 고도 제한으로 인해 상공 통과가 불가능하므로 비행 경로에 포함되지 않도록 'unreachable'로 상태를 설정한다.

장애물 상공 통과 경로 ('Fly-over' path) 계산 단계에서는 영역 내 존재하는 장애물의 상공으로 진행되는 경로 (P_{FO})를 계산한다. 기존 Wavefront 알고리즘과 다른 점은 경로를 구성하는 셀 ID 외에 각 셀을 지날 때의 비행 고도를 함께 경로에 포함시킨다는 점이며 이는 무인기가 3차원 공간에서 장애물의 상공으로 통과하기 위함이다.

장애물이 존재하지 않는 셀의 비행 고도는 최소 비행 고도인 A_{MIN} 로 설정되며, 장애물이 존재하지만 상공으로 통과 가능한 셀의 경우 비행 고도는 장애물 높이에 최소 비행 고도 A_{MIN} 를 더한 값으로 결정된다. 이는 무인기가 해당 셀의 상공으로 통과할 때 장애물과 충돌을 방지하기 위함이다.

장애물 우회 경로 ('Fly-around' path) 계산 단계에서는 영역 내 존재하는 장애물의 측면 우회 경로(P_{FA})를 계산한다. 우회 경로 계산 시 장애물이 존재하는 셀을 제외한 나머지 셀들만을 포함하여 비행 경로를 구성하게 된다.

최종 경로 선택 단계에서는 이전 단계들에서 계산된 장애물 상공 통과 경로 (P_{FO})와 장애물 우회 경로 ('Fly-around' path)의 비용을 각각 계산 후 더 비용이 적은 경로를 반환한다. 각 경로의 비용은 무인기가 해당 경로를 따라 비행했을 때의 총 이동 거리로 정의된다.

IV. Simulation-based Performance Evaluation

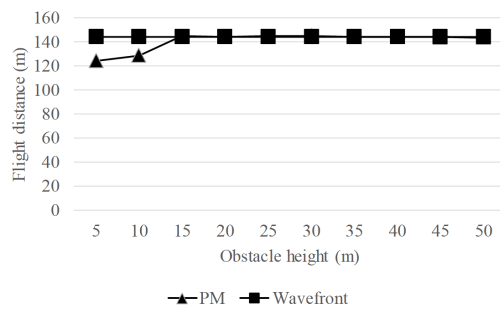
본 논문에서 제안된 무인기 경로 계획 방법의 성능을 예상하기 위해 기존 경로 계획 방법인 Wavefront 알고리즘과 시뮬레이션을 통한 비교 평가를 수행하였으며 세부 실험 환경은 Table 3과 같다.

Table 3. Simulation environment variables

Environment variables	Value (Range)
Area size	100 * 100 (m ²)
Cell size	5 * 5 (m ²)
Obstacle height	5 ~ 50 (m)
Obstacle width	30 / 50/ 70 (m)
Maximum altitude (A_{Max})	50 (m)
Minimum altitude (A_{Min})	5 (m)

영역 크기는 (100m × 100m)는 도심 건물 블록, 소규모 농경지 등 실제 무인기 임무가 빈번한 환경의 대표적 규모이며, 한국 항공안전법상 비가시권 비행 가능 범위(반경 500m) 내에 해당한다. 셀 크기(5m × 5m)는 일반 쿼드콥터 크기에 해당하면서도 실시간 처리가 가능한 계산 복잡도 (400개 셀)을 유지하도록 설정하였다. 최대/최소 비행 고도(50m/5m)는 항공안전법의 고도 제한(150m 이하) 범위 내에서 도심 저고도 임무(항공촬영, 건물점검 등)의 일반적 운용 고도를 반영했으며, GPS 오차와 지상 장애물을 고려한 최소 안전 이격 거리를 적용하였다. 장애물 높이와 너비는 실제 도심환경의 다양한 구조물들을 대표하도록 설정하였으며 이를 통해 제안 기법의 성능을 검증하였다.

Software-in-the-loop (SITL) simulation 환경 구축을 위해 SITL 버전의 PX4 펌웨어와 jMAVSim 시뮬레이터, 지상관제 소프트웨어인 QGroundControl을 사용하였다. 출발지 (0, 0)와 목적지 (100, 100) 위치를 고정한 상태에서 기존 기법 (Wavefront algorithm)과 제안기법 (DPAM)을 사용하여 도출된 경로를 따라 모의환경에서 무인기를 비행시킨 후 이동 거리를 비교하였다. Fig.2 (a)-(c)는 서로 다른 너비를 갖는 장애물이 존재하는 환경에서 장애물 높이 증가에 따른 기존/제안 경로 계획 알고리즘의 성능 (비행 거리) 비교 결과를 나타낸다.



(a) (Obstacle width = 30 (m))

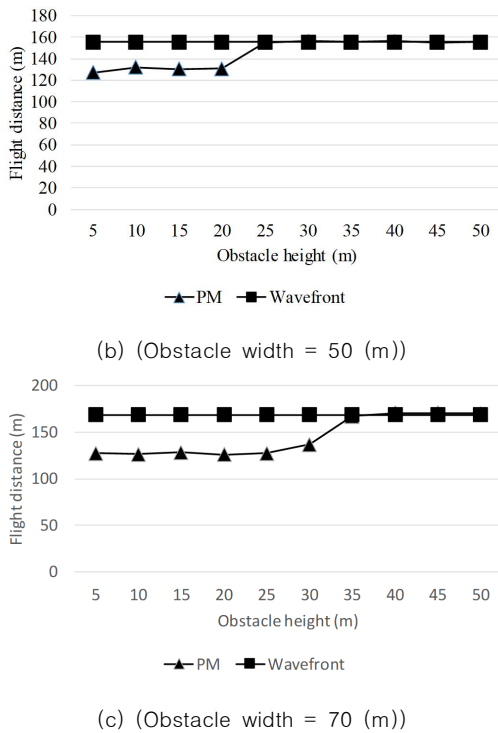


Fig. 2. Performance comparison between Wavefront and the proposed path planning algorithms

Fig.2에서 PM은 제안된 경로 계획 방법, Wavefront는 기존 Wavefront 알고리즘을 나타낸다. 또한 x 축은 장애물의 높이 변화를 나타내며 y 축은 장애물의 높이 증가에 따른 비행 거리 변화를 나타낸다. 제안 경로 계획 방법은 장애물의 위치와 너비, 높이를 고려하여 비행 경로가 결정된다. 따라서 장애물 너비 변화에 따른 (a), (b), (c)는 각각 장애물의 너비가 30, 50, 70m 일 때 비행 거리 측정 결과를 보여 준다. (a), (b), (c) 공통적으로 기존 Wavefront 알고리즘은 비행 거리 측면에서 장애물 높이에 관계 없이 일정한 값을 보이는데 이는 2차원 공간을 가정하여 항상 장애물의 측면으로 우회하는 경로를 선택하기 때문이다. 반면 제안된 경로 계획 방법은 장애물의 높이가 장애물 너비의 1/2 이하일 때는 해당 장애물의 상공으로 통과하는 경로를 선택하여 우회 경로에 비해 더 짧은 비행 거리를 갖게 된다. 그림 1 (a)에서 볼 수 있듯이 장애물의 너비가 30m인 경우 장애물의 높이가 30m 이하일 때는 제안 기법의 비행 거리가 Wavefront 방식에 비해 약 86% 수준으로 단축할 수 있다. 또한 장애물의 너비가 50m, 70m일 경우 (그림 1 (b), (c)), 각각 81% 75% 수준으로 비행 거리를 단축할 수 있음을 확인하였다. 단 장애물의 높이가 장애물 너비의 1/2 이상일 경우 제안 방법 역시 해당 장애물을 측면으로 우회하는 경로를 선택하며 따라서 기존 Wavefront 알고리즘과 동일한 비행 거리를 보인다. 위 결

과를 종합하면 제안된 경로 계획 방법은 장애물의 너비에 비해 상대적으로 장애물의 높이가 낮은 환경에 유리한 것을 알 수 있다.

V. Conclusions

무인기는 3차원 공간에서 임무를 수행하므로 지상 로봇과 달리 장애물의 상공으로 통과하거나 측면으로 우회하는 2가지 선택지가 존재한다. 지상 로봇을 위한 기존 경로 계획 기법인 Wavefront 알고리즘의 경우, 항상 장애물의 측면으로 우회하는 경로만 제공함으로써 더 효율적인 경로를 제공하지 못 하는 한계점이 존재했다. 따라서 본 논문에서는 기존 경로 계획 방식인 Wavefront 알고리즘을 확장하여 영역 내 존재하는 장애물의 위치, 너비, 높이 등을 고려한 이중 경로 선택 기법을 제안하였다. 제안된 이중 경로 선택 기법은 1) 장애물 상공 통과 경로와 2) 장애물 측면 우회 경로 계산 후 두 경로의 이동 거리를 비교하여 둘 중 더 짧은 경로를 제공한다. 또한 SITL 시뮬레이션 결과결과 제안 기법이 기존 알고리즘에 비해 비행 거리 면에서 10~20%의 성능향상 효과를 보임을 확인하였다. 본 연구의 한계점은 기존 경로 계획 알고리즘인 Wavefront 알고리즘의 확장이라는 것과 이로 인해 무인기 임무수행 영역 내 3D 공간 특성을 반영하는데 어려움이 있다는 점이다. 또한 비록 장애물의 높이와 너비를 다양하게 설정하였으나 다양한 위치에 존재하는 정적/동적 장애물 등을 고려하지 않아 필드 적용에 제한점이 존재한다. 그러나 장애물의 위치와 크기를 미리 알 수 있는 환경에서는 간단한 구현을 통해 이동 거리 단축 및 효과적 임무 수행에 기여 가능하다. 본 연구를 더 발전시키기 위해서는 추후 지표면 높이 변화, 다양한 위치와 크기의 장애물, 동적 객체 등을 고려한 경로 계획 방법에 대한 추가 연구가 필요하다.

REFERENCES

- [1] TTA terms - terms.tta.or.kr
- [2] A Survey of Quadrotor Unmanned Aerial Vehicles
- [3] F. Nex and Fabio Remondino. "UAV for 3D mapping applications: a review." *Applied geomatics*, Vol. 6, No. 1, pp. 1-15 Mar. 2014.
- [4] J. T. K. Ping et al. "Generic unmanned aerial vehicle (UAV) for civilian application-A feasibility assessment and market survey on civilian application for aerial imaging." *Proc. of Sustainable Utilization and Development in Engineering and Technology*

(STUDENT) 2012, pp. 289-294

- [5] I. Lim, S. R, "Waypoints Altitude Planning for Terrain Collision Avoidance and Maneuverability of an Unmanned Aerial Vehicles," Journal of KIIT, Vol. 16, No. 9, pp. 31-41 Sep. 2018 (in Korean)
- [6] L. Meier and D. Honegger, and Marc Pollefeys. "PX4: A node-based multithreaded open source robotics framework for deeply embedded platforms." Proc. of IEEE International Conference on Robotics and Automation (ICRA) 2015, pp. 6235-6240
- [7] V. Shivashankar et al. "Real-Time Planning for Covering an Initially-Unknown Spatial Environment." Proc. of FLAIRS Conference 2011
- [8] A. Zelinsky et al. "Planning paths of complete coverage of an unstructured environment by a mobile robot." Proc. of international conference on advanced robotics 1993. pp. 533-538
- [9] E. Galceran and M. Carreras. "A survey on coverage path planning for robotics." Robotics and Autonomous systems, Vol. 61, No. 12, pp. 1258-1276, 2013.
- [10] C. Ryu, S. Ra, "Mission Planning and Control Analysis Algorithms for the Safe Flight of an Unmanned Aerial Vehicle," Journal of KSAS, Vol. 33, No. 3, pp. 71-77 Mar. 2005 (in Korean)
- [11] J. Wang, Z. Zhao, J. Qu et al. "APPA-3D: an autonomous 3D path planning algorithm for UAVs in unknown complex environments." Scientific Reports, Vol. 14, No. 1231, Jan. 2024.
- [12] "A robust multiple Unmanned Aerial Vehicles 3D path planning strategy via improved particle swarm optimization." Physica A: Statistical Mechanics and its Applications, Sep. 2023.
- [13] W. Zhang, J. Li, W. Yu, P. Ding, J. Wang, X. Zhang. "Algorithm for UAV path planning in high obstacle density environments: RFA-star." Frontiers in Plant Science, Vol. 15, Oct. 2024.
- [14] "Path Planning of UAV Using Levy Pelican Optimization Algorithm in Mountain Environment." Taylor & Francis, Jun. 2024.
- [15] "Energy-Optimized 3D Path Planning for Unmanned Aerial Vehicles." Applied Sciences (MDPI), Vol. 14, No. 16, Aug. 2024.

Authors



Sooyoung Moon received the B.S., M.S. and Ph.D. degrees in Computer Engineering from Sungkyunkwan University, Korea, in 2007, 2009 and 2015, respectively.