

## 特輯論文

DOI: <http://dx.doi.org/10.5139/JKSAS.2012.40.11.979>

## 무인전투기 근접 지형추종을 위한 궤적생성 및 유도 항법

오경택\*, 서중보\*, 김형석\*, 김유단\*\*, 김병수\*\*\*

## Trajectory Generation, Guidance, and Navigation for Terrain Following of Unmanned Combat Aerial Vehicles

Gyeong-taek Oh\*, Joong-Bo Seo\*, Hyoung-Seok Kim\*, Youdan Kim\*\* and Byungsoo Kim\*\*\*

## ABSTRACT

This paper implements and integrates algorithms for terrain following of UCAVs (Unmanned Combat Aerial Vehicles): trajectory generation, guidance, and navigation. Terrain following is very important for UCAVs because they perform very dangerous missions such as Suppression of Enemy Air Defences while the terrain following can improve the survivability of UCAVs against from the air defence systems of the enemy. To deal with the GPS jamming, terrain referenced navigation based on nonlinear filter is chosen. For the trajectory generation, Voronoi diagram is adopted to generate horizontal plane path to avoid the air defense system. Cubic spline method is used to generate vertical plane path to prevent collisions with ground while flying sufficiently close to surface. Follow-the-Carrot and pure pursuit tracking methods, which are look-ahead point based guidance algorithms, are applied for the guidance. Numerical simulation is performed to verify the performance of the integrated terrain following algorithm.

## 초 록

본 논문에서는 무인전투기의 지형추종을 위한 궤적생성, 유도, 항법 알고리즘을 구현하고 이를 통합하였다. 적 방공망 제압과 같은 위험한 임무를 수행하는 무인전투기는 적의 대공방어망으로부터 생존성을 높이기 위해서 지면을 근접하여 비행하는 지형추종 알고리즘이 필수적으로 요구된다. 무인전투기가 지형추종 비행을 하기 위해서는 경로생성, 유도, 그리고 항법 분야 알고리즘이 통합되어야 한다. 본 논문에서는 항법 알고리즘으로 GPS가 교란된 상황을 대비하여 비선형 필터 기반의 지형참조 항법을 사용하였다. 경로생성을 위해 지형추종에 적합한 경로생성 기법으로 보로노이 다이어그램을 이용하여 적의 대공망을 회피하는 수평경로를 생성하고, Cubic Spline 기법을 사용하여 정해진 수평경로 상에서 지면과의 근접비행이 가능한 수직경로를 생성하였다. 유도 알고리즘으로 전방주시점 기반의 유도법칙인 Follow-the-Carrot 기법과 Pure Pursuit 기법을 사용하여 생성된 경로를 추종하게 하였다. 제안한 통합알고리즘의 성능을 검증하기 위하여 수치 시뮬레이션을 수행하였다.

**Key Words** : UCAV(무인전투기), Terrain Following(지형추종), Look-ahead Point(전방주시점), Terrain Referenced Navigation(지형참조항법)

† 2012년 8월 28일 접수 ~ 2012년 10월31일 심사완료

\* 정회원, 서울대학교 기계항공공학부

\*\* 정회원, 서울대학교 기계항공공학부

교신저자, E-mail : ydkim@snu.ac.kr

\*\*\* 정회원, 경상대학교 항공우주시스템공학과

## I. 서 론

현대전에서 공군력은 가장 핵심적인 요소로 꼽히는 전력이다. 공군에서 펼치는 많은 중요한 작

전 중에서도 가장 먼저 수행되어야 하는 작전은 적의 대공망을 무력화시키는 적 방공망 제압작전(SEAD, Suppression of Enemy Air Defenses)이다. 걸프전 당시 사막의 폭풍 작전에서 연합군의 최고 우선순위는 적 대공망의 파괴였다. 또한 최근 미군 전투기 출격 임무 중 약 25%가 SEAD 임무라는 사실은 현대전에서 SEAD가 가지는 중요성을 시사한다[1]. 그러나 SEAD를 수행하기 위해서는 적의 대공망을 미리 탐지해야 하는데, 이는 적 방공망의 사정거리 내에 놓이게 됨을 의미하므로 상당한 위험이 수반된다. 따라서 최근에는 인간 조종사를 대신하여 경제적이고 효율적인 공격이 가능한 무인전투기에 대한 요구가 증가하고 있다.

적의 레이더로부터 피탐될 확률을 낮추기 위해서는 비행체가 지면과 최대한 가까이 붙어서 비행하는 방법이 있다. 이는 곧 SEAD 임무를 수행하는 무인전투기의 생존률과 직결되는 요소이다. 전통적인 지형추종 알고리즘에 관한 연구는 1960년대에 이미 개발되어 B-1, F-111, Tornado, F-16, 토마호크 미사일 등에 적용된 바 있다.

Lu와 Pierson은 지형추종 문제를 크게 경로생성, 유도, 그리고 지형정보에 대한 센서융합으로 구성하였다[2]. 경로생성을 위해 Funk[3]는 Cubic Spline 기법을 활용하여 지형을 추종하는 경로를 생성하였고[3], Asseo는 구배 기법 parabolic fight segment를 활용하여 수평 및 수직 경로를 생성하였다[4]. Betts는 수치기법을 활용한 궤적 최적화에 관한 리뷰 논문을 발표하였고[5], Lu와 Pierson은 지형추종 문제에 관한 분석을 수행한 후 역동역학 기법을 활용하여 수치 최적화문제를 풀었다[2]. Sharma는 F-4 전투기 질점모델의 3차원 최적 경로생성 문제를 직접 기술(direct transcription) 방법을 활용하여 풀었다[6].

한편 전장에서는 관성항법 장치의 표류오차를 보정하기 위한 수단인 위성항법 장치가 전파교란을 받을 수 있기 때문에, 이를 대체할만한 독립적인 항법시스템이 필요하다. 따라서 지형의 정보를 활용하여 위치를 계산하는 지형참조항법(Terrain Reference Navigation)에 관한 연구가 수행되었다. 지형의 고도는 비선형성이 상당히 심하고, 때에 따라 나무에 의한 고도 또는 지면 고도가 측정될 수 있으므로 2개의 모드를 가지는 가우시안 혼합모델로 잡음을 모델링 하여야 한다. 이러한 이유로 인해 초기 지형참조항법은 확장형 칼만필터(Extended Kalman Filter)를 활용하여 비선형성을 극복하려 하였으나, 초기치가 부정확할 때 발산하는 등 불안정한 성능을 보이

곤 하였다. 최근에는 컴퓨터의 성능이 좋아짐에 따라 베이시안 필터들이 활발히 연구되었다. 한편, 최근에는 파티클 필터(Particle Filter)를 사용한 지형참조항법 연구가 수행되었다. 파티클 필터는 무수한 파티클을 확률공간상에 뿌린 후, 파티클들의 likelihood를 가중치로 삼아 그 가중평균을 추정치로 사용하는 필터를 말한다. 파티클 수가 많아질수록 파티클들이 확률밀도함수에 가깝게 근사되므로 오차제곱의 평균을 최소화하는 추정치가 된다. Bergman은 확률공간 전체에 임의로 뿌리는 파티클을 대신하여, 이산적인 확률공간을 입자들로 채우는 Point Mass Filter를 제안하였는데, 간단한 문제에서는 파티클 필터보다 적은 계산시간과 비슷한 정확도를 가진다[7]. Nordlund는 파티클 필터와 칼만필터를 결합하여 연산시간을 줄이는 Rao-Blackwellized Particle F(RBPF)를 제안하였다[8].

본 논문에서는 무인전투기의 지형추종을 위한 경로생성, 유도, 그리고 항법 알고리즘을 선정하고 통합하였다. 이를 위해 우선 통합 알고리즘의 입/출력 구조를 정의하였고, 그에 맞게 각 알고리즘을 구현하고 통합하였다. 경로생성 알고리즘으로는 전장에서 적의 레이더망을 회피하기 위한 수평경로생성을 위해 보로노이 다이어그램을 사용하였고, 생성된 수평경로 상에서 지형을 추종하기 위한 수직경로는 Cubic Spline 기법을 사용하였다. 유도 알고리즘으로는 전방주시점 기반 기법 중에서 2가지 기법인 Follow-the-Carrot 기법과 Pure Pursuit 기법을 이용하여 유도 알고리즘을 구현하였다. 항법 알고리즘으로는 RBPF를 구현하여, GPS가 교란당하는 상황에서 무인기의 위치를 계산하였다. 제안한 지형추종 유도항법 통합 알고리즘의 성능을 평가하기 위해서 수치 시뮬레이션을 수행하였다.

## II. 본 론

### 2.1 통합알고리즘 입출력 구조

지형추종 유도항법 통합 알고리즘을 위한 세부 구성요소와 각 구성요소 간에 필요한 입출력 데이터를 Fig. 1과 같이 정리하였다.

경로계획 단계에서는 목표물과 아군 무인전투기의 초기위치, 적 대공망의 위치를 통해 전장의 상황을 파악한다. 지형을 추종하는 궤적을 생성하기 위해서는 전장지형의 고도자료가 제공되어야 하는데, 본 연구에서는 DTED(Digital Terrain Elevation Data)를 사용하는 상황을 고려하였다.

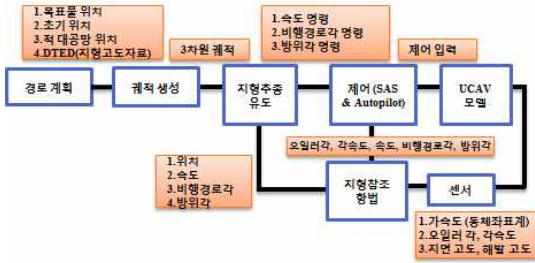


Fig. 1. 통합 알고리즘 입출력 구조

우선 지형참조항법 단계에서는 관성센서와 고도센서에서 측정된 가속도, 오일러각, 각속도, 지면/해발 고도 등을 이용하여 무인전투기의 위치와 속도를 추정한다. 전장의 정보가 주어지면, 궤적생성 단계에서 지형을 추종하며 적 대공망을 회피하는 동시에 목표지점으로 도달하는 3차원 궤적을 생성한다. 여기서 목표지점에서의 임무는 대공망 파괴 및 무력화, 주요시설 파괴, 정찰 및 감시 등의 임무를 고려할 수 있다. 주어진 3차원 기준궤적에 대해 지형추종 유도 단계에서는 궤적을 추종하기 위한 속도, 비행경로각, 방위각 명령을 계산한다. 각 단계의 상세한 알고리즘 설명은 다음 절에서 다룬다.

2.2 지형참조항법

2.2.1 문제 정의

지형참조항법은 비행을 하면서 측정하는 지형고도와 비행체 내부에 탑재된 지형고도 정보를 비교함으로써 비행체의 위치, 속도 및 가속도를 추정하는 방법이다. 관성항법장치만으로도 위치를 추정할 수 있으나, 적분으로 인해 발생하는 표류오차를 보정하지 않으면 신뢰도 있는 위치를 얻을 수 없다. 최근에는 위성항법장치가 이를 보정해주고 있으나, 전장상황에서는 전파교란에 의해 수신에 용이하지 않을 수 있다. 따라서 지형추종을 하는 무인전투기에는 독자적으로 기체의 위치를 추정함으로써 관성항법장치를 보정할 수 있는 수단이 필수적으로 요구된다. Fig. 2는 전파고도계와 기압고도계를 이용해 지형의 고도를 측정하고 이를 지형 지도와 비교하는 지형참조항법의 개념을 설명하고 있다.

이러한 문제는 결국 측정치가 지형의 고도이고 추정해야 하는 것이 위치와 속도인 문제이므로, 다음과 같은 이산화된 선형 운동학(kinematics) 방정식과 비선형 측정치 식으로 추정문제를 정의할 수 있다.



Fig. 2. 지형참조항법의 개념도

$$\begin{bmatrix} p_{k+1} \\ v_{k+1} \\ \delta a_{k+1} \end{bmatrix} = \begin{bmatrix} I & T_s \cdot I & T_s^2/2 \cdot I \\ 0 & I & T_s \cdot I \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} p_k \\ v_k \\ \delta a_k \end{bmatrix} + \begin{bmatrix} T_s^2/2 \cdot I \\ T_s \cdot I \\ 0 \end{bmatrix} a_k + \begin{bmatrix} T_s^3/6 \cdot I \\ T_s^2/2 \cdot I \\ T_s \cdot I \end{bmatrix} w_k \quad (1)$$

$$y_k = h(p_k) + e_k \quad (2)$$

여기서  $p_k, v_k, a_k$ 는 위치, 속도, 가속도계 측정값을 나타내고,  $\delta a_k$ 는 가속도계의 바이어스(bias),  $w_k$ 는 가속도계의 바이어스를 표현하기 위한 가우시안 잡음이다.  $T_s$ 는 이산 시스템의 샘플링 시간이고  $y_k$ 는 전파/기압 고도계에서 얻어진 지형의 고도,  $h(p_k)$ 는 DTED를 bilinear 보간 기법으로 얻은 해당 위치  $p_k$ 에서의 지형고도이다. 또한  $e_k$ 는 고도계의 측정오차, DTED와 실제 지형과의 차이, 보간오차를 포함하는 값으로, 본 연구에서는 전파고도계가 나무의 위와 지면을 측정하는 상황을 고려하여 2개의 모드를 가지는 가우시안 혼합 모델로 고려하였다. Fig. 3은 본 연구에서 시뮬레이션에 사용한 평균이 0, 12이고 표준편차가 2, 6인 가우시안 혼합 모델로부터 얻은 10,000개의 샘플의 상대빈도를 나타낸 것이다.

한편 가속도 센서의 바이어스는 일반적으로 가우시안 랜덤 워크(random walk)로 가정을 하는데, 가우시안 잡음이 계속해서 적분되는 형태를 가진다. 이를 수식으로 나타내면 다음과 같다.

$$a_{true} = a_k + \delta a_k, \delta a_k = \delta a_{k-1} + T_s * w_k \quad (3)$$

Fig. 4는 가우시안 잡음과 이를 적분한 값인 가

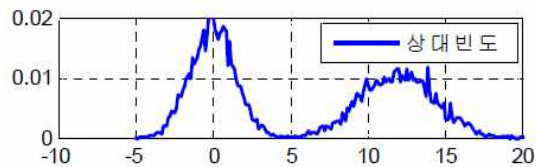


Fig. 3. 가우시안 혼합모델 샘플의 상대빈도

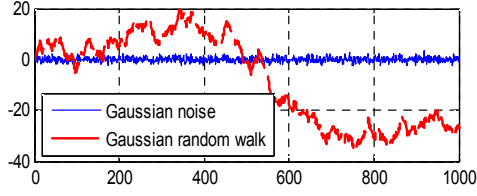


Fig. 4. 가우시안 잡음과 랜덤 워크

우시안 랜덤 워크, 즉 가속도 바이어스  $\delta a$ 의 모델을 나타낸다. 여기에서 바이어스 잡음 외의 가속도계의 측정 잡음은 상대적으로 크기가 작다고 가정하여 고려하지 않도록 한다.

### 2.2.2 Rao-Blackwellized Particle Filter

고전적인 지형참조항법은 1950년대부터 연구되어 실전에 사용되었으나, TERCOM이나 SITAN과 같은 초기의 알고리즘들은 일괄처리(Batch) 방식을 취하여 연속적인 값을 얻을 수 없거나 초기 위치오차가 클 경우에 발산을 하는 등의 단점을 내포하고 있었다. 본 연구에서는 상대적으로 많은 계산량을 필요로 하지만 초기 오차에 강건하고 지형의 비선형성과 비가우시안 측정잡음에도 우수한 추정성능을 가지는 베이시안 추정기법(Bayesian Estimation)을 사용하였다. 베이시안 추정기법에서는 사후(*a posteriori*) 확률밀도함수  $p(x_t | Y_t)$ 를 계산하고, 이를 바탕으로 추정치인  $\hat{x}_t$ 를 계산한다. 재귀적인 방법으로 확률밀도함수를 구하는 식은, 사후 확률밀도함수를 베이시안 규칙을 사용하여 다음과 같이 나타낼 수 있다.

$$p(x_t | Y_t) = \frac{p(y_t | x_t)p(x_t | Y_{t-1})}{p(y_t | Y_{t-1})} \quad (4)$$

이때 분모는 다음의 식을 통해 구할 수 있다.

$$p(y_t | Y_{t-1}) = \int_{R^n} p(y_t | x_t)p(x_t | Y_{t-1})dx_t \quad (5)$$

한편 이들을 재귀적으로 구해나가기 위해서는  $p(x_{t+1} | Y_t)$ 가 필요한데, 이는 다음의 식을 통해 구할 수 있다.

$$p(x_{t+1} | Y_t) = \int_{R^n} p(x_{t+1} | x_t)p(x_t | Y_t)dx_t \quad (6)$$

위의 적분을 실제로 수행하는 것은 매우 어렵기 때문에 상태변수 공간상에 많은 수의 샘플을 임의로 생성하여 적분을 합으로 치환하는 기법들이 제시되었는데, 파티클 필터와 Point Mass Filter들이 그 예이다[7]. 본 연구에서는 파티클 필터에

칼만 필터를 결합함으로써 계산량을 줄일 수 있는 Rao-Blackwellized Particle Filter(RBPF)를 적용하였다[8,9]. RBPF를 적용하기 위해서는 모델 방정식과 측정치 방정식은 다음과 같은 형식으로 표현될 수 있어야 한다.

$$\begin{aligned} x_{k+1}^1 &= f(x_k^1) + A_k^1 x_k^2 + B_k^1 w_k^1 \\ x_{k+1}^2 &= A_k^2 x_k^2 + B_k^2 w_k^2 \\ y_k &= h(x_k^1) + e_k \end{aligned} \quad (7)$$

RBPF에서는  $x_k^1$ 은  $f$ ,  $h$ 와 같은 비선형 함수와 연관이 되었으므로 비선형 필터인 파티클 필터를 사용하고,  $x_k^2$ 는 선형의 관계만을 가지므로 칼만 필터를 적용한다. 이때 파티클에 속하는 상태변수의 수를 줄일 수 있으므로 계산량이 줄어든다. 앞에서 정의한 지형참조항법 문제를 위와 같은 형식으로 표현하기 위해 상태변수를 다음과 같이 관성항법장치의 오차로 정의한다.

$$\begin{aligned} \delta p &= p_{true} - p_{INS} & x_k^1 &= \delta p_k \\ \delta v &= v_{true} - v_{INS} \\ \delta a &= a_{true} - a_{INS} & x_k^2 &= [\delta v_k, \delta a_k]^T \end{aligned} \quad (8)$$

위의 정의에 의해 지형참조항법 문제는 다음과 같이 표현될 수 있다.

$$\begin{aligned} \begin{pmatrix} x_{k+1}^1 \\ x_{k+1}^2 \end{pmatrix}_{6 \times 1} &= \begin{pmatrix} I_{2 \times 2} & A^1 \\ O_{4 \times 2} & A^2 \end{pmatrix}_{6 \times 6} \begin{pmatrix} x_k^1 \\ x_k^2 \end{pmatrix}_{6 \times 1} + \begin{pmatrix} B^1 \\ B^2 \end{pmatrix}_{6 \times 2} w_k \\ A^1 &= (T_s I_{2 \times 2} | (T_s^2 / 2) I_{2 \times 2}), & B^1 &= (T_s^3 / 6) I_{2 \times 2}, \\ A^2 &= \begin{pmatrix} I_{2 \times 2} & T_s I_{2 \times 2} \\ O_{2 \times 2} & I_{2 \times 2} \end{pmatrix}, & B^2 &= \begin{pmatrix} (T_s^2 / 2) I_{2 \times 2} \\ T_s I_{2 \times 2} \end{pmatrix}, \\ p(w_k) &= p(w_k^n, w_k^e) = N \left( \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} Q & 0 \\ 0 & Q \end{pmatrix} \right). \end{aligned} \quad (9)$$

Table 1에 위의 정의한 변수들을 이용하여 세부적인 RBPF 알고리즘을 정리하였다.

Fig. 5는 Table 1의 1번 과정에서 초기에 생성된 파티클과 참값, 그리고 초기 추정치를 보여주고 있다. 실제 상황에서는 초기위치로 설정된 값이 참값이 아니기 때문에, 고도를 측정하는 실제 위치와 필터 상의 초기 추정치와의 차이가 존재한다. 따라서 필터는 초기 추정치 주변에서 생성될 수밖에 없다. Fig. 5에서도 참값은 (0,0)에 존재하나, 초기 추정치인 (2,2)를 중심으로 정규분포를 갖는 파티클을 생성하였다. 그러나 공산(likelihood)을 바탕으로 가중치를 계산하는 5번

Table 1. 위치를 추정하기 위한 RBPf 알고리즘

1.  $k=0$ .  $N$ 개의 파티클을 현재 추측되는 초기 위치 주위에서 생성한다. 이때 초기 위치 오차의 공분산을 이용한다. 또한 칼만필터를 초기화하고, 파티클 필터의 가중치들을 초기화시킨다.

$$\{\hat{x}_0^{1,i}\}_{i=1}^N \sim N(\hat{x}_0^1, P_0^1)$$

$$\{\hat{x}_0^{2,i}\}_{i=1}^N \sim N(\hat{x}_0^2, P_0^2)$$

$$\{w_0^i\}_{i=1}^N = 1/N$$

2. 모델 방정식을 이용하여 다음 시간에서의 파티클을 예측한다.

$$x_{k+1}^{1,i} \in N(x_k^{1,i} + A^1 \hat{x}_{k|k-1}^{1,i}, A^1 P_{k|k-1}^2 (A^1)^T + B^1 Q(B^1)^T)$$

3. 칼만 필터를 업데이트 시킨다.

$$S_k = A^1 P_{k|k-1}^2 (A^1)^T + B^1 Q(B^1)^T$$

$$K_k = P_{k|k-1}^2 (A^1)^T S_k^{-1}$$

$$P_{k|k}^2 = P_{k|k-1}^2 - K_k A^1 P_{k|k-1}^2$$

$$\hat{x}_{k|k}^{2,i} = \hat{x}_{k|k-1}^{2,i} + K_k (x_{k+1}^{1,i} - x_k^{1,i} - A^1 \hat{x}_{k|k-1}^{2,i})$$

4. 다음 시간에서의 칼만 필터 추정치를 예측한다.

$$\hat{x}_{k+1|k}^{2,i} = A_2 \hat{x}_{k|k}^{2,i}$$

$$P_{k+1|k}^2 = A_2 P_{k|k}^2 (A_2)^T + B_2 Q(B_2)^T$$

5. 가중치를 계산하고 정규화시킨다.

$$w_{k+1}^i = \frac{p(y_{k+1} | x_{k+1}^{1,i}) w_k^i}{\sum_{j=1}^N p(y_{k+1} | x_{k+1}^{1,j}) w_k^j}, i=1, \dots, N$$

6.  $N_{eff} < N_{threshold}$  이면 리샘플링을 수행한다.

$$\hat{N}_{eff} = 1 / \sum_{i=1}^N (w_{k+1}^i)^2$$

$$x_{k+1}^{1,i} \in \{x_{k+1}^{1,j}\}_{j=1}^N, \text{ where } p(x_{k+1}^{1,i} = x_{k+1}^{1,j}) = w_{k+1}^j$$

$$w_{k+1}^i = 1/N, i=1, \dots, N$$

7. 최소평균제곱 추정을 수행한다.

$$\hat{x}_{k+1}^{1,MMS} = \sum_{i=1}^N w_{k+1}^i x_{k+1}^{1,i}$$

$$\hat{x}_{k+1}^{2,MMS} = \sum_{i=1}^N w_{k+1}^i x_{k+1|k}^{2,i}$$

$$\hat{p}_k = p_{INS} + \hat{x}_{k+1}^{1,MMS}$$

$$[\hat{v}_k \hat{a}_k]^T = [v_{INS} a_{INS}]^T + \hat{x}_{k+1}^{2,MMS}$$

8.  $k=k+1$ , 2로 돌아간다.

과정이 있고, 이를 통해 가중평균으로 추정을 하는 7번 과정이 존재하므로, 실제 위치 주변의 파티클들이 높은 가중치를 가지게 되고, 추정치는 점차 참값에 근접하게 된다.

통합 시뮬레이션에서 필터의 초기위치 오차의 표준편차는 200m, 공정 잡음  $w_k$  의 표준편차는

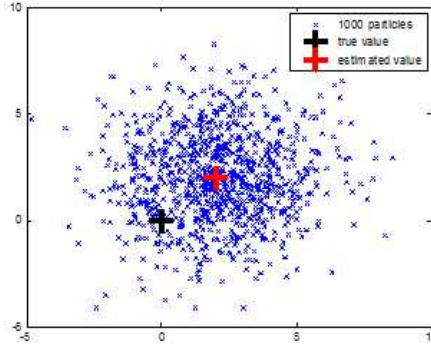


Fig. 5. 초기 파티클 생성 과정

0.0001, 파티클의 수는 1,000개,  $N_{threshold}$  는  $N$  의 2/3으로 설정하였다.

### 2.3 지형추종 궤적생성

#### 2.3.1 수평궤적 생성

본 연구에서는 위협을 회피하면서 목표지점에 도달하기 위한 궤적을 생성하기 위해 수평/수직 분리기법을 사용하였다. 대공망을 피해가는 궤적을 생성하기 위해 보로노이 다이어그램을 이용하여 수평면의 궤적을 생성한 뒤, 해당 궤적 상에서 지형을 추종할 수 있도록 수직 궤적을 생성하였다. Fig. 6은 본 연구에서 설정한 전장을 보여주고 있다. Fig. 6에 검은 실선으로 보로노이 다이어그램과 대공망을 회피하는 수평궤적(붉은 점선)을 보여주고 있으며, 등고선은 DTED 정보를 사용하여 나타내었다.

#### 2.3.2 수직궤적 생성

수직궤적은 생성된 수평궤적 상에서 Cubic Spline 기법으로 계산하였다. Cubic Spline 기법은 수평궤적을 길이방향에 따라 일정한 간격으로

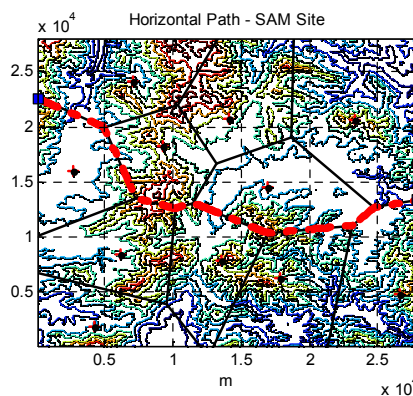


Fig. 6. 생성된 수평 궤적과 전장 정보

나누고, 각 구간의 경계점 고도들이 다음과 같은 3차 다항식을 만족하도록 함으로써 부드러운 수직궤적이 생성되도록 한다.

$$H_i(R) = h = a_i R^3 + b_i R^2 + c_i R + d_i \quad (10)$$

여기서  $R$  은 수평궤적의 길이방향 거리,  $H_i(R)$  는  $i$ 번째 구간에서 해당 위치  $R$  에서의 해발 고도이며,  $a_i \sim d_i$  는  $i$ 번째 구간에서의 3차 다항식의 계수이다.

생성되는 궤적은 수직고도가 지면으로부터 설정된 안전 고도를 지켜야하며, 기울기가 무인전투기의 상승/하강 한계치를 초과하지 않아야한다. 이를 수식으로 정리하면 표 2와 같다.

Fig. 7은 지면으로부터의 안전 고도를 50m로 설정했을 때 생성된 수직궤적을 나타낸다. Fig. 7에서 실선은 지형고도이고 위의 점선은 Cubic Spline을 이용한 궤적이다.

### 2.4. 지형추종 유도 알고리즘

#### 2.4.1 질점 모델과 전방주시점

본 논문에서는 제어기 성능이 충분히 빠르게

Table 2. 수직궤적 생성 알고리즘

Find	$H_i(R) = a_i R^3 + b_i R^2 + c_i R + d_i \quad i = 0, \dots, N-1$
subject to	$H_{i-1}(R_i) = H_i(R_i) = h_i,$
	$\left. \frac{d}{dR} H_{i-1}(R) \right _{R_i} = \left. \frac{d}{dR} H_i(R) \right _{R_i},$
	$\left. \frac{d^2}{dR^2} H_{i-1}(R) \right _{R_i} = \left. \frac{d^2}{dR^2} H_i(R) \right _{R_i},$
	$\left  \left. \frac{d}{dR} H_i(R) \right _{R_i} \right  < s,$
$h_i$ 는 $i$ 번째 목표고도, $s$ 는 상승/ 하강 한계치	

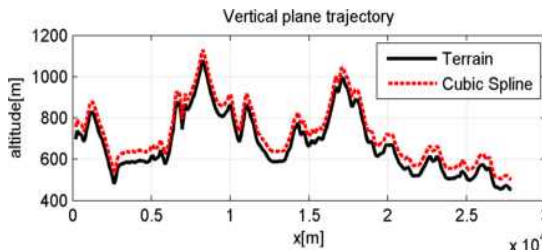


Fig. 7. 생성된 수직궤적

설계되어 비행경로각과 방위각 명령을 즉시 추종한다고 가정하였으며, 지형추종 성능평가를 위해 다음과 같은 질점 운동학 모델을 고려하였다.

$$\dot{x} = V \cos \gamma \cos \chi \quad (11)$$

$$\dot{y} = V \cos \gamma \sin \chi \quad (12)$$

$$\dot{z} = -V \sin \gamma \quad (13)$$

$$\dot{\gamma} = \dot{\gamma}_{cmd} \quad (14)$$

$$\dot{\chi} = \dot{\chi}_{cmd} \quad (15)$$

여기서  $x, y, z$ 는 위치벡터의 성분이며,  $V$ 는 속력,  $\gamma$ 는 비행경로각, 그리고  $\chi$ 는 방위각이다.

본 연구에서는 계산시간이 적은 유도 알고리즘을 선정하였는데, 이는 험난한 산악지형이 많은 국내 상황을 고려할 때 추종해야 할 기준궤적의 고도변화가 상대적으로 심함에 따라 알고리즘의 효율성이 중요하기 때문이다. 전방주시점 기반의 유도 알고리즘은 기준궤적에 전방주시점을 설정함으로써 가상의 움직이는 점을 계속해서 추종하다보면 기준궤적을 쫓아가는 원리를 이용한다. 전방주시점의 정의는 Fig. 8과 같다. Fig. 8에서  $L$ 은 전방주시 거리로, 설계 변수에 해당한다.

전방주시점 기반 유도 알고리즘 중에서 2차원 유도 문제에서 상당히 우수한 성능을 보여주는 Follow-the-Carrot 기법과 Pure Pursuit 기법을 지형추종에 적용하였다. 2.3절에서 구한 궤적정보에는 길이방향 이동거리  $R$ 이 함께 정의되어 있으므로, 무인기 위치  $X'(t)$ 로부터 기준궤적으로 정사영 시키고, 정사영된 지점  $X'_L(t)$ 으로부터 궤적의 길이 방향으로 전방주시 거리만큼 떨어진 지점을 전방주시점  $X'_L(t)$ 으로 계산하였다.

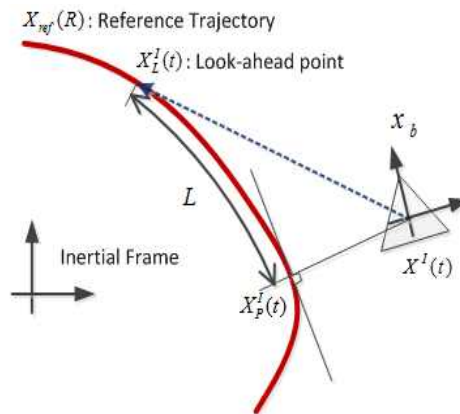


Fig. 8. 전방주시점의 정의

### 2.4.2 Follow-the-Carrot

Follow-the-Carrot 알고리즘은 비행체의 이동방향을 전방주시점으로 향하도록 하여 기준궤적을 추종하도록 하는 알고리즘이다. 무인기의 위치  $X^I(t)$ 로부터  $X_L^I(t)$ 를 향하는 목표벡터  $T^I(t)$ 를 다음과 같이 정의한다.

$$T^I(t) = X_L^I(t) - X^I(t) = [T_x \quad T_y \quad T_z] \quad (16)$$

관성좌표계에 대한  $T^I(t)$ 의 고도각(elevation)과 방위각(azimuth)는 다음과 같다.

$$\begin{aligned} \gamma_T(t) &= \tan^{-1}\left(\frac{-T_z}{\sqrt{T_x^2 + T_y^2}}\right) \\ \chi_T(t) &= \tan^{-1}\left(\frac{T_y}{T_x}\right) \end{aligned} \quad (17)$$

시간  $t$ 에서의 비행경로각과 방향각은  $\gamma(t), \chi(t)$  이므로, 비행경로각과 방향각의 오차는 다음과 같다.

$$\begin{aligned} e_\gamma(t) &= \gamma_T(t) - \gamma(t) \\ e_\chi(t) &= \chi_T(t) - \chi(t) \end{aligned} \quad (18)$$

이때 비례제어 이득에 의해 기준입력은 다음과 같이 결정된다.

$$\begin{aligned} \dot{\gamma}_{cmd}(t) &= K_{p,\gamma} e_\gamma(t) \\ \dot{\chi}_{cmd}(t) &= K_{p,\chi} e_\chi(t) \end{aligned} \quad (19)$$

### 2.4.3 Pure Pursuit Path Tracking

Pure Pursuit 알고리즘은 유도탄 유도제어에 활용되는 알고리즘에 기반하고 있다. 유도탄 제어에서의 Pure Pursuit 유도기법에 의한 비행이 부드러운 곡선을 그린다는 점을 반영하여  $X^I(t)$ 에서  $X_L^I(t)$ 로 원형궤적을 가질 수 있도록 유도 명령을 계산한다[9]. Fig. 9는 무인기 동체의 X-Y 좌표계에 대해서 Pure Pursuit 유도기법의 개념도를 나타내고 있다.

Fig. 9에서  $\kappa_\chi$ 는 곡률을 나타낸다. 동체좌표계의 XYZ 축은 진행방향을 X축으로 정의하고, 수직 직교하는 두 축을 Y축, Z축으로 정의한다. 따라서 동체좌표계는 관성좌표계의 Z축에 대해  $\chi$ 만큼 회전시키고, 이어서 Y축에 대해 다시  $\gamma$ 만큼 회전한 좌표계가 된다. 이때 동체좌표계에 대한 전방주시점의 위치  $X_L^B(t)$ 는 방향코사인 행렬

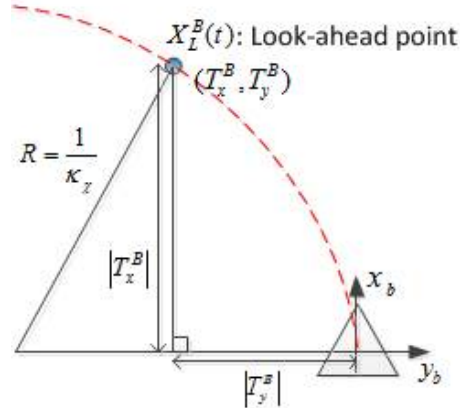


Fig. 9. Pure Pursuit 개념도

$R_L^B(t)$ 에 의해 다음과 같이 정의된다.

$$X_L^B(t) = [T_x^B \quad T_y^B \quad T_z^B] = R_L^B(t) T^I(t) \quad (20)$$

이때 간단한 기하학적 관계에 의해 곡률  $\kappa_\chi$ 는 다음과 같이 구해진다.

$$\kappa_\chi = \frac{2|T_y^B|}{(T_x^B)^2 + (T_y^B)^2} \quad (21)$$

같은 방식으로 동체의 X-Z축에 대해 적용하면 다음 식을 얻을 수 있다.

$$\kappa_\gamma = \frac{2|T_z^B|}{(T_x^B)^2 + (T_z^B)^2} \quad (22)$$

한편 회전 방향은  $T_y^B, T_z^B$ 에 의해 결정되며, 동체의 X-Z축 상에서 유도명령을 생성할 경우 X-Y축 상에서의 방향과 반대가 되어야 하므로 각속도 명령은 다음과 같이 얻어진다.

$$\begin{aligned} \dot{\gamma}_{cmd} &= -\kappa_\gamma V \operatorname{sgn}(T_z^B) \\ \dot{\chi}_{cmd} &= -\kappa_\chi V \operatorname{sgn}(T_y^B) \end{aligned} \quad (23)$$

## 2.5 수치 시뮬레이션 결과

본 연구에서 정의한 입출력 구조를 바탕으로 개별 알고리즘을 통합하여 수치 시뮬레이션을 수행하였다. 수치 시뮬레이션은 10Hz로 수행되었고, 유도 알고리즘은 Follow-the-Carrot 기법과 Pure Pursuit 유도기법을 선택적으로 사용할 수 있도록 하였다. 질점모델의 속도는 200m/s으로 일정하다고 가정하였다. Fig. 10은 약 90m의 해

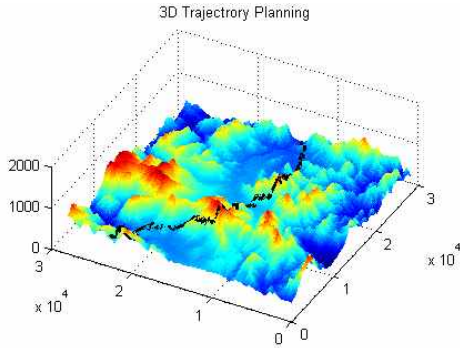


Fig. 10. 3차원 궤적과 추종 결과

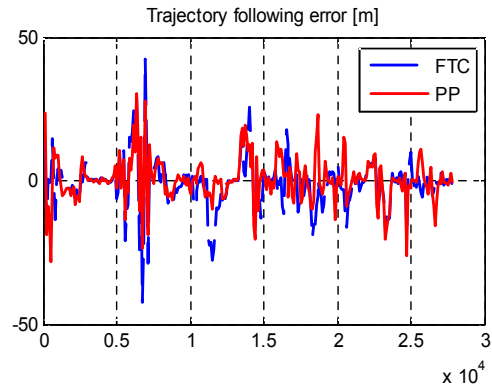


Fig. 12. 수직궤적 추종 오차

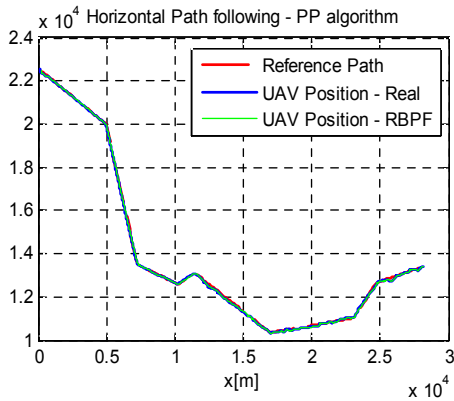


Fig. 11. 수평궤적 추종 결과

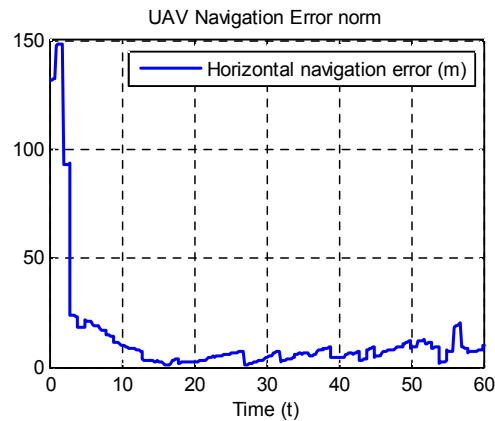


Fig. 13. 지형참조항법의 성능 (수평거리오차)

상도를 가지는 DTED level 1을 이용한 3차원 궤적과 추종 결과를 보여주고 있다. Fig. 11은 Pure Pursuit 기법을 이용한 수평궤적의 추종결과와 지형참조항법을 통해 추정된 궤적을 함께 표시하였다. 궤적이 복잡한 형태가 아니므로 Follow-the-Carrot 기법과 Pure Pursuit 유도기법 모두 우수한 성능을 보여주었다.

수직궤적의 추종결과는 전방주시 거리의 설정에 민감하게 변하였다. 전방주시 거리가 짧아질수록 가까운 거리만을 생각하므로 궤적에 지연이 커졌다. 전방주시 거리를 늘리면 미리 예측을 하기 때문에 궤적 지연은 줄어들으나, 구불구불한 경로는 부드러운 경로로 통과하는 등의 지름길 현상이 발생하였다. 전방주시 거리가 50m 일 때, Follow-the-Carrot 기법과 Pure Pursuit 유도기법의 수직궤적 추종 오차는 Fig. 12에 나타내었다.

초기에 기준궤적에 진입하기 전에는 Follow-the-Carrot 기법이 우수한 성능을 보이나, 궤적 진입 후에는 전반적으로 Pure Pursuit 유도기법이 더 우수한 성능을 가졌다. 안전거리를 50m로 설정하였으므로 두 알고리즘 모두 지면과의 충돌

은 발생하지 않았다.

지형참조항법의 성능은 시뮬레이션 상에서 별도로 계산하는 참위치와 추정위치의 수평거리 오차로서 나타낼 수 있다. Fig. 13은 시간에 따른 항법오차를 나타내고 있다. Fig. 13에서 보듯이 시간이 갈수록 초기의 위치오차가 줄어드는 것을 확인할 수 있다. 그러나 지형참조항법의 특성상 지형이 평평하다거나 자기유사성(self-similarity)을 가질 경우, 지형으로부터 위치정보를 알아낼 수 없으므로 추정성능은 지형의 험준도에 따라 달라질 수 있다.

### III. 결 론

본 논문에서는 무인전투기의 지형추종을 위해 지형참조항법, 궤적생성, 그리고 지형추종 유도 알고리즘을 구현하고 통합하였다. 보로노이 다이어그램과 Cubic Spline을 통해 생성된 궤적은 전



방주시점 기반의 유도법칙에 의해 추종되었고, 위성항법장치의 정보 없이도 Rao-Blackwellized Particle Filter를 통해 독자적인 위치추정을 할 수 있는 지형참조항법이 적용되었다. 수치 시뮬레이션의 결과, 지형참조항법, 궤적생성 및 유도 알고리즘은 입출력 구조에 따라 성공적으로 통합되었으며, 만족할만한 성능을 보임을 확인할 수 있었다. 본 연구 결과는 무인전투기의 적 대공망 제압 작전을 수행하기 위한 지면추종 임무에 활용될 수 있을 것이며, 다수의 무인전투기 임무에도 적용이 가능할 것이다.

### 후 기

본 연구는 방위사업청과 국방과학연구소의 지원으로 수행되었습니다. (계약번호UD100048JD)

### 참고문헌

1) Bolkcom, C., "Military Suppression of Enemy Air Defenses (SEAD): Assessing Future Needs," CRS Report for Congress, RS21141, Congressional Research Service, Library of Congress, 2005.

2) Lu, P., and Pierson, B. L., "Optimal Aircraft Terrain Following Analysis and Trajectory Generation," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 3, 1995, pp. 555-560.

3) Funk, J. E., "Optimal-Path Precision Terrain-Following System," *Journal of Aircraft*, Vol. 14, No. 2, 1977, pp. 128-134.

4) Asseo, S. J., "Terrain Following/Terrain Avoidance Path Optimization Using the

Method of Steepest Descent," *IEEE National Aerospace and Electronics Conference*, Dayton, OH, May 1988.

5) Betts, J. T., "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 193-207.

6) Sharma, T., Williams, P., Bil, C., and Eberhard, A., "Optimal Three Dimensional Terrain Following and Collision Avoidance," *Australia and New Zealand Industrial and Applied Mathematics Journal*, Vol. 47, 2007, pp. 695-711.

7) Bergman, N. Recursive Bayesian Estimation - Navigation and Tracking Application, Ph. D. Dissertation, Department of Electrical Engineering, Linkoping University, Linkoping, Sweden, 1999.

8) Nordlund, P. J., and Gustafsson, F., "Sequential Monte Carlo Filtering Techniques Applied to Integrated Navigation Systems," *American Control Conference*, Arlington, VA, June 2001.

9) Coulter, R. C., "Implementation of the Pure Pursuit Path Tracking Algorithm," Technical Report. CMU-RI-TR-92-01, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 1992.

10) Gustafsson, F., Bergman, N., Forsell, U., Jansson, J., Karlsson, R., and Nordlund, P.-J., "Particle Filters for Positioning, Navigation, and Tracking," *IEEE Transactions on Signal Processing*, Vol. 50, No. 2, 2002, pp. 425-437.