

비행 운동 시뮬레이션을 위한 좌표계 변환

The Coordinate Conversion for Flight Dynamics Simulation

백중환*, 황수찬*, 김철영*

Joong-Hwan Baek*, Soo-Chan Hwang*, and Chil-Yong Kim*

요 약

비행 시뮬레이터는 엔진 모듈, 항법장치 모듈, 계기 모듈로 구성된다. 여기에는 두 가지의 문제점을 가지고 있다. 첫째는, 각 모듈별로 사용하는 좌표계가 서로 다르다는 점이다. 이를 해결하기 위해 본 논문에서는 각기 독자적인 좌표계를 일치시키기 위한 방법을 고안하였다. 둘째는, 지구 형태가 완전한 구 모양을 따르고 있다고 가정하고 구 삼각공식을 이용하여 거리와 방위각을 계산하게 되면 많은 연산량이 요구된다. 본 논문에서는 근거리 비행시 구 삼각공식 대신 평면 근사화 방식을 사용함으로써 우수한 결과를 얻음을 확인하였다. 연산 속도는 구 삼각공식 사용시 4.95×10^{-4} 초에서 평면 근사화 방식을 사용시 1.648×10^{-4} 초로 향상됨을 확인하였다.

Abstract

A flight simulator is composed of engine, navigation systems and instrument modules. However, two problems exist here. First, the coordinate of each independent module is not same. To solve this problem, we design a method that mutual coordinates are capable of transformation each other. Second, the distance and bearing between two points on the earth are computed in a sphere shape using the spherical trigonometry. However, the computing time is very severe. In this paper, we project the sphere into the planar to reduce the computing time. An experimental result shows that the performance of the proposed method is excellent to both distance and bearing calculations in close region. Also, the computing time is reduced from 4.95×10^{-4} seconds to 1.648×10^{-4} seconds.

I. 서 론

비행 운동 모델링 부분은 비행기의 운동 방정식을 효과적으로 해석하기 위해 동체 좌표계를 이용하여 비행기에 작용하는 힘과 모멘트에 대한 물리적 의미를 해석하게 된다. 그리고, 항법전자장치부분은 구 공간상에서 비행기의 위치와 거리 및 방위각(bearing)을 획득하여 비행기의 비행 상황인 피각주치(pitch), 롤(roll), 요우(yaw) 및 속도, 고도(altitude) 등을 표

시하게 된다. 즉, 비행기의 운동 모델링은 동체 좌표계인 직각 좌표계를 따르고 있고, 항법전자장치는 구 공간을 표현하기 위한 경·위도 좌표계 형태이다. 비행기의 운동을 표현하는 좌표계와 실제 비행기의 이동 경로 등을 표현하는 좌표계가 일치하지 않는다는 문제점을 가지고 있다.

또한, 구 공간상에서 거리 및 방위각 해석법은 구 삼각 공식[3]을 이용하게 된다. 이와 같은 연산법의 문제점은 많은 \cos , \sin 및 실수연산이 존재한다는 것이다. 그러므로, CBT(Computer-Based Trainer)

* 한국항공대학교(Hankuk Aviation Univ.)

· 논문번호 : 99-2-5

· 접수일자 : 1999년 10월 13일

에서 실시간 연산에 어려움을 가지고 있다. 따라서, 근거리에서 구 공간상의 연산을 대신한 직각 좌표계 (cartesian coordinates)로 사상(mapping)하여 거리와 방위각을 구하였다.

본 논문의 구성은 2장에서 비행 운동 시뮬레이션, 3장에서는 각 모듈별에 사용되는 좌표계 변환을 설명하였다. 4장은 항법 전자 장치에서 사용되는 거리 및 방위각 계산을, 5장에서는 제안한 방법을 이용한 실험 및 검토를 하고, 6장에서 결론을 맺는다.

II. 비행 운동 시뮬레이션

비행기용 시뮬레이터에서 엔진 모델링은 6자유도 비행운동 모델링과 엔진 모델로 크게 구성된다. 비행기는 조종원과 엔진 등에 대한 입력을 시작으로 지상에서 이륙하여 상승, 순항, 선회, 하강, 착륙 등의 운동을 한다. 이와 같은 운동은 입력에 따른 운동 방정식으로 나타낼 수 있다.

먼저 비행 운동 모델링은 비행기의 운동방정식을 임의의 직교 좌표계에 대해 표현할 수 있고, 임의의 적합한 좌표계를 선택함에 따라 방정식의 단순화 및 비행기에 작용하는 힘과 모멘트에 대한 물리적 의미를 부여할 수 있게 된다. 본 논문에서 시뮬레이션 해석 시에 사용되는 좌표계는 네 종류로 동체 좌표계 (body axes system), 관성(inertia) 좌표계, 안정 (stability) 좌표계 그리고, 비행경로(flight path) 좌표계 등이다[1],[4].

그리고, 비행 중 비행기에 작용하는 힘과 모멘트를 계산하기 위해서는 여러 변수들의 값을 파악해야 하며, 정확한 공력계수 값들을 설정하게 된다. 이와 같은 공력 계수값과 좌표계를 이용하여 6자유도 비행운동을 모델링하게 된다[4].

그림 1은 계기 비행 훈련용 시뮬레이터에 적용되는 비행 운동에 대한 흐름도를 나타낸 것이다. 즉, 비행 운동 모델링에서 사용되는 각 공력 계수와 비행기에 영향을 주는 외부 환경에 따라 비행기의 고도, 속도, 비행거리와 기수 방향 등을 산출하게 된다. 그러나, 비행 운동을 해석하기 위해서 자신의 독자적인 좌표계로 해석을 하고 있으므로 경·위도 좌표계와는 차이가 있게 된다.

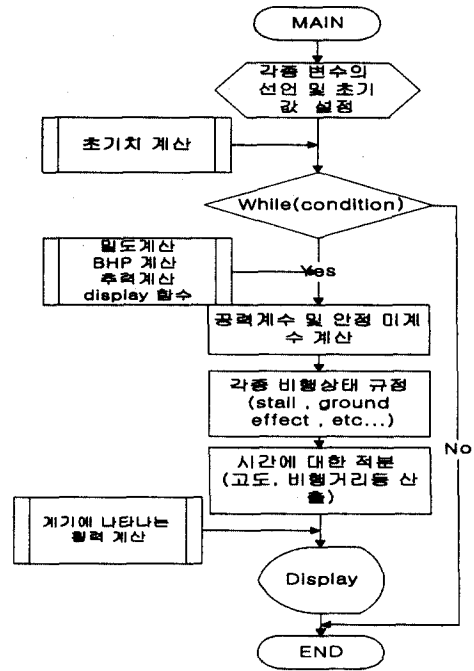


그림 1. 계기 비행 훈련용 CBT 흐름도

Fig. 1. Flow chart of CBT for flight navigation simulator.

III. 좌표계 변환

비행 운동 좌표계에서는 그림 2에서와 같이 동체 좌표계 (body axes, [feet])를 사용하고 있다. 그리고, 실제 경·위도 좌표계와는 독립된 형태로 운동 방정식을 유도하게 된다.

그림에서 볼 수 있듯이 동체 좌표계는 비행기의 진행 방향을 x 축으로 설정하고 있다. 즉, 비행기가 θ 방향으로 출발하였다면, 두 좌표계와의 관계는 그림 3과 같은 형태로 나타나게 될 것이다.

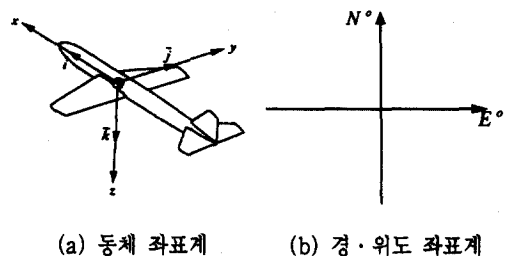


그림 2. 동체 좌표계와 경·위도 좌표계

Fig. 2. Body axes and longitude-latitude axes.

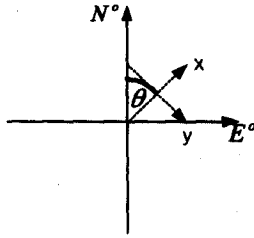


그림 3. 동체 좌표계와 경·위도 좌표계 간의 관계도
Fig. 3. Relationship between body axes and longitude-latitude axes.

그림 3에서 보는 것과 같이 좌표 축 변환에 의해서 두 좌표계를 일치시킬 수 있고, 단위 계를 일치시켜야 한다. 즉, 동체 좌표계를 바로 경·위도 좌표계로 변화하기 어려움으로 동체 좌표계를 절대 좌표계로 바꾼 후 절대 좌표계를 경·위도 좌표계로 변환하게 된다.

$$\begin{bmatrix} X \\ Y \end{bmatrix} = [T_{coord}] \begin{bmatrix} x \\ y \end{bmatrix} \quad (1)$$

여기서, (x, y) 는 비행기의 이동 좌표값이고, T_{coord} 는 좌표 축 변환이다. 그리고, (X, Y) 는 좌표 축 변환에 의한 비행기의 이동 결과 값이다. 단, 단위계는 meter이다.

$$T_{coord} = \begin{bmatrix} \cos(-\theta) & \sin(-\theta) \\ -\sin(-\theta) & \cos(-\theta) \end{bmatrix} \quad (2)$$

식 (1)의 결과 값은 비행기가 이동한 거리이다. 그러므로, 비행기의 현재의 경·위도에 변화량만큼 연산하게 된다. 하지만, 실제 지구의 형태가 평면이 아닌 타원의 형태이므로 직각 좌표계로 사상하여 사용하게 된다

기존에 사용되어 오던 구 삼각공식은 지구가 타원이라는 것을 이용하여 지구 중심에서 비행기의 현재 좌표와 스테이션의 좌표간에 거리와 방향을 구하였다[1]~[4]. 하지만, 이와 같은 방식의 문제점은 실제 지구의 형태는 구가 아닌 타원이며, 국내의 영역은 북반구의 위쪽에 위치하므로 위도와 경도를 미터(meter)단위로 변화하였을 때 정확하게 일치하지 않게 된다. 그러므로, 직각 좌표계를 사용하기 위해

표 1. 경·위도 좌표와 직각좌표의 변환 테이블

Table 1. Conversion table of longitude-latitude coordinates and cartesian coordinates.

지역	위도 1'당	지역	경도 1'당
N30°	1848 m	E120°	1608 m
N40°	1851 m	E130°	1423 m

서는 경·위도를 일정한 비율로 바꾸어서 사용할 수 있다. 따라서, 표 1에 기술된 위도와 경도의 분당 거리를 이용하여 거리를 구하였다.

표 1에서 기술된 값들은 중간의 임의의 값들은 기술하고 있지 않기 때문에 선형 보간법(linear interpolation)을 사용하여 임의의 지역에서의 환산 값을 찾게 된다. 예를 들어 임의의 지점의 위도와 경도가 각각 N°, E° 라면, 1분당 거리 x_m 과 y_m 은 다음 식으로 구할 수 있다.

$$\begin{aligned} x_m &= 1848 + \frac{(N-30) \times (1851 - 1848)}{10} \\ y_m &= 1608 + \frac{(E-120) \times (1423 - 1608)}{10} \end{aligned} \quad (3)$$

x_m 과 y_m 값을 이용하여 수식 (1)에서 얻어진 비행기의 이동 거리 (X, Y) 를 각도(degree)단위로 바꾸게 된다. 즉, 수식 (4)에 따라서 값을 구하게 된다.

$$\begin{aligned} \theta' &= (Y/y_m)/60 \\ \alpha' &= (X/x_m)/60 \end{aligned} \quad (4)$$

여기서, θ' 은 위도의 각 변화량을 의미하게 되고, α' 은 경도의 각 변화량을 의미하게 된다.

식 (4)에서 얻어진 각도를 위도와 경도의 형태로 바꾸어 원 비행기의 좌표에 비행기의 이동 변화량만큼을 변화시키게 된다. 그리고, 원 위치 좌표와 비행기의 이동 좌표를 연산하였을 때, 분과 초에서 오버플로우나 언더플로우가 발생하지 않도록 보정 단계를 거치게 된다.

IV. 거리 및 방위각 연산

4-1 구 삼각공식에 의한 거리와 방위각 연산

두 지점간의 거리와 방위각을 구하는 방법은 평면일 경우에는 아주 간단한 문제일 것이다. 하지만, 실제 비행기는 지구의 상공을 통과하고 있게 된다. 그리고, 각 항법전자장치가 설치된 곳은 지상에 존재하고 있으므로 그림 4와 같이 구삼각형의 형태가 될 것이다.

그림 4에서 지점 D는 현재 지구 상공의 비행기 위치이다. 그리고, A는 비행기의 목적지이고, B는 비행기의 출발지이다. 그리고, 비행 거리(ℓ) nautical mile(NM)로 표시하게 된다. 그리고, 이때 자북(N) 방향과의 차이가 방위각(bearing angle)이 된다. 구 삼각공식에 의한 거리를 계산하는 공식[3]은 식 (5)와 같다.

$$\begin{aligned} \cos c &= \cos a \cdot \cos b + \sin a \cdot \sin b \cdot \cos C \\ c &= \cos^{-1}(\cos c) \end{aligned} \quad (5)$$

여기서, a, b, c 는 각 거리이고, 각 C 는 다음 식으로 나타낼 수 있다.

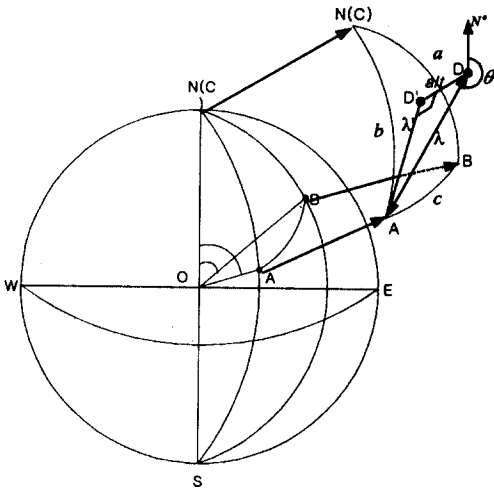


그림 4. 구 삼각형
Fig. 4. The spherical triangle.

$$C = (B \text{ 지점의 경도}) - (A \text{ 지점의 경도}) \quad (6)$$

따라서, A 지점과 B 지점 간의 각 거리 c 는 아크 코사인을 이용하여 구할 수 있다. 즉, 두 지점간 (D', A)의 거리 (ℓ')는 지구 전체 둘레에 대한 각 거리 (2π)와 두 지점간의 각 거리 c 를 지구의 둘레에 곱함으로써 구할 수 있다. 즉, 거리 ℓ' 은 식 (7)과 같다.

$$\ell = \left(\frac{c}{2\pi}\right) \cdot (2 \cdot \pi \cdot R) / 1852 = c \cdot R / 1852 \quad (7)$$

여기서, R 은 지구의 반지름으로 6,371,100 m를 적용하고, 1852는 meter를 nautical mile(NM)로 변환하기 위한 계수이다. 또한, 비행기의 현재 고도(altitude)로부터의 사거리(slant range)는 식 (8)과 같이 구한다(그림 5).

$$\ell = \sqrt{\ell'^2 + alt^2} \quad (8)$$

그리고, 한 지점 D로부터 다른 지점 A로의 상대 방위각은 구 삼각공식을 이용하여 구할 수 있다[3]. 즉, 자북(N) 방향을 기준으로 D와 A에서의 각을 이용하여 현재 방위를 표시할 수 있고, 이는 구 삼각공식을 이용하여 아래의 식으로 나타낼 수 있다.

$$\begin{aligned} \cos a &= \cos b \cdot \cos c + \sin b \cdot \sin c \cdot \cos A \\ \cos A &= \frac{\cos a - \cos b \cdot \cos c}{\sin b \cdot \sin c} \end{aligned} \quad (9)$$

여기서, 방위각 A 는 다음의 수식으로 주어진다.

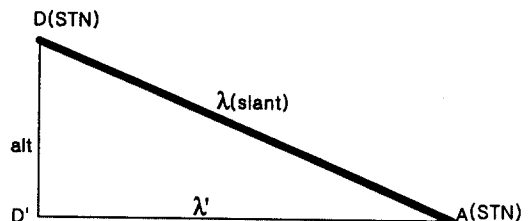


그림 5. 사거리 계산
Fig. 5. Calculation of slant range.

$$A = \cos^{-1}(\cos A) \quad (10)$$

이 때 얻어진 값에서 진북과 자북의 편차를 보정해야 하는데 우리나라 지역에서는 7° W(즉, 진북이 자북에서부터 서쪽으로 7° 기울어져 있음을 의미)이므로 7° 만큼을 보정한다. 단, 다른 지역에서는 편차의 크기와 방향이 위도와 관계없이 다르다.

4-2 평면 근사화 방법에 따른 거리와 방위각 연산

앞에서 살펴보았듯이 구 삼각공식을 이용하여 거리와 방위각을 구하기 위해서는 기본적으로 cos과 sin공식을 이용하게 된다. 이는 많은 연산량을 요하게 되며, 특히 실시간 시뮬레이션이 필요한 CBT의 경우 속도에 많은 영향을 미친다. 따라서, 본 연구에서는 효과적으로 연산량을 줄이기 위해 구 삼각 공식보다는 구를 평면으로 근사화 시키는 방법을 제시한다.

그림 6은 구 형태를 평면으로 근사화시키는 그림이다. 그림에서 A는 도착지 공항의 항법 장치국(STN)이고, D는 비행기의 현 좌표(PSN)이고, D'은 직각좌표에 투영(projective)된 지점이다.

평면에서 두 점(A, D)간의 거리를 구해보면, 각 점들은 경·위도좌표로 표기되어 있으므로 경도와 위도를 따로 나누어서 연산하게 된다. 그러므로 경도를 degree값으로 변환한 후 NM으로 변환하여 두 경도 사이의 거리를 연산하게 된다. 여기서, A점의 경위도

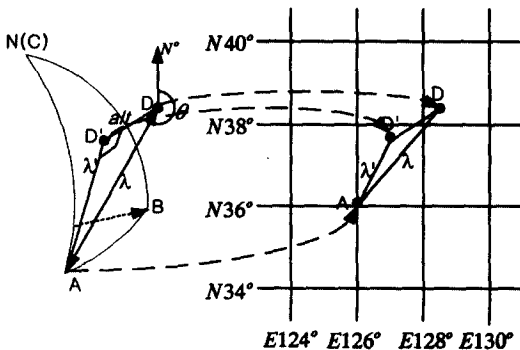


그림 6. 구 좌표계를 직각 좌표계로의 사상
Fig. 6. Mapping from spherical coordinates to cartesian coordinates.

를 $E(a_1^0 b_1^0 c_1^0) \rightarrow E(a_1, b_1, c_1)$ 와 $N(a_2^0 b_2^0 c_2^0) \rightarrow N(a_2, b_2, c_2)$ 로 표시하면, D점의 경위도가 $E(x_1, y_1, z_1)$ 와 $N(x_2, y_2, z_2)$ 일 때 두 점간의 거리를 구해보기로 한다.

우선, 각 경위도의 좌표를 degree값으로 변환하면 다음과 같다.

$$\begin{aligned} Deg(A_E) &= a_1 + b_1/60 + c_1/3600 \\ Deg(D_E) &= x_1 + y_1/60 + z_1/3600 \\ Deg(A_N) &= a_2 + b_2/60 + c_2/3600 \\ Deg(D_N) &= x_2 + y_2/60 + z_2/3600 \end{aligned} \quad (11)$$

그리고, 아래의 식을 이용하여 두 값의 차를 구하게 된다.

$$\begin{aligned} dist'_E &= |Deg(A_E) - Deg(D_E)| \\ dist'_N &= |Deg(A_N) - Deg(D_N)| \end{aligned} \quad (12)$$

이 값은 아래 식을 이용하여 NM값으로 변환하게 된다.

$$\begin{aligned} l_E &= dist'_E \times 60 \times x_m / 1852 \\ l_N &= dist'_N \times 60 \times y_m / 1852 \end{aligned} \quad (13)$$

그리고, 두 좌표점 사이의 거리 l' 를 구하면 식 (14)와 같다.

$$l' = \sqrt{l_N^2 + l_E^2} \quad (14)$$

식 (14)에서 나온 결과 값을 식 (15)에 대입하여 사거리 l 을 구하게 된다.

$$l = \sqrt{l'^2 + alt^2} \quad (15)$$

방위각을 구하는 방법은 그림 7과 같이 A에서 방위각을 구하면 된다.

$$ang = \tan^{-1}(l_N/l_E) \quad (16)$$

위 식으로부터 얻어진 값에서 D 지점을 원점으로 하여 A 지점이 어떤 사사분면에 있는지를 파악하여

진북(N)으로부터의 방위각을 계산한다. 또한, 결과의 값에 진북과 자북과의 편차인 7° 만큼을 보정하게 된다.

V. 실험 및 검토

본 논문에서 시뮬레이션한 환경은 펜티엄-133, 32M의 메모리를 이용하였고, 개발 툴은 비주얼 C++ 5.0을 이용하였다. 본 실험 및 검토에서는 기존에 제시되었던 구 삼각 공식을 이용하여 거리와 방위각을 구하는 대신 평면 근사화 방법을 이용하였을 때 정확도와 실시간 처리가 가능한지를 규명해보기로 한다.

먼저, 표 2와 같이 비행기의 현재 위치는 부산 공항이고 목적지는 대구 공항으로 선택하였을 때 거리와 방위각을 구해본다(그림 8).

비행 경로 차트 [9]에 의하면 두 지점간의 방위각은 167도이고 거리는 50 NM이다. 따라서, 구 삼각 공식에 따른 결과와 평면 근사화 방법에 따른 결과

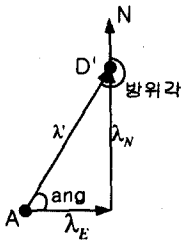


그림 7. 방위각 계산
Fig. 7. Calculation of bearing.

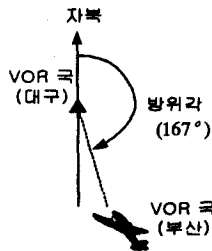


그림 8. 실험을 위한 예
Fig. 8. An example for experiment.

표 2. 두 지점의 좌표와 비행기 고도

Table 2. Coordinates of two points and airplane's altitude.

	비행기의 현재위치(PSN) 부산 공항	목적지 공항(STN) 대구 공항
위도 (N)	35 07' 12"	35 53' 54"
경도 (E)	129 00' 06"	128 38' 36"
고도	3000ft	

를 비교 분석하기로 한다.

먼저 구 삼각공식에 의해서 구해보면, 방위각은 166.5도이고, 거리는 49.9 NM로 산출된다. 그리고, 본 논문에서 제시하고 있는 알고리즘을 이용하여 구하면 다음과 같다.

① PSN과 STN:

$$Deg(D_E) = 129 + 0/60 + 06/3600 = 129.0017$$

$$Deg(A_E) = 128 + 38/60 + 36/3600 = 128.641667$$

$$Deg(D_N) = 35 + 7/60 + 12/3600 = 35.12$$

$$Deg(A_N) = 35 + 53/60 + 54/3600 = 35.898333$$

② $dist'_E | Deg(A_E) - Deg(D_E) |$

$$= | 128.6417 - 129.0017 | = 0.36$$

$$dist'_N | Deg(A_N) - Deg(D_N) |$$

$$= | 35.8983 - 35.12 | = 0.7783$$

③ 비행기의 현재 위치와 공항의 위치사이의 분당 거리를 구해 보면,

$$x_m = 1849.7695$$

$$y_m = 1448.129167$$

과 같이 나타나고, 이것을 NM으로 바꾸어 보면 다음과 같다.

$$l_E = dist'_E \times 60 \times y_m / 1852 = 0.4 \times 60$$

$$\times 1448.9 / 1852 = 16.889627$$

$$l_N = dist'_N \times 60 \times x_m / 1852 = 0.778 \times 60$$

$$\times 1850 / 1852 = 46.643756$$

④ l' 을 구해 보면,

$$l' = \sqrt{l_N^2 + l_E^2} = 49.607454$$

⑤ l 을 고도를 이용하여 구해보면, 우선 고도를 NM단위로 변환하여 거리를 구할 수 있다. 피트를 NM으로 바꾸는 공식은 (feet * 0.3048) / 1852 이다.

$$l = \sqrt{0.4937^2 + 49.607454^2} = 49.61$$

㉔ 식 (16)을 이용하여 방위각을 구하면,

$$ang = \tan^{-1}(l_E/l_N) = \tan^{-1}(46.643756 / 16.889627) = 70.095^\circ$$

목적지 공항을 원점으로 하여 비행기의 현재 위치가 4사분면에 위치하므로 위에서 얻어진 값에 90°를 더하면 160.095°가 되고, 여기에 편차 7°를 더하면 최종 방위각은 167.095°도가 된다. 따라서, 거리는 약 0.4 NM, 방위각은 약 0.1°의 차이가 발생하였다.

또 다른 실험으로, 한국항공대학교에서 운영중인 FRASCA 142 시뮬레이터의 결과와 두 방식의 결과를 표 3에 나타내었다.

표 3을 검토하여 보면, 출발지와 도착지의 거리가 40NM 이하에서 구 삼각공식과 평면 근사화 방식 사이의 거리와 방위각의 차이는 대략 0.1~0.2 NM 이고, 방위각은 약 0.1~0.3°의 차이가 있음을 볼 수 있다.

연산 속도를 측정하기 위해 비행 운동 좌표를 1000개 입력하였다. 구 삼각 공식을 사용할 때는 약 0.494505초가 소요되었고, 평면 근사화 방식을 사용할 때는 0.164835초가 소요되었다. 따라서, 실시간 연산이 필요로 하는 CBT에서 평면 근사화 방식이

보다 적합한 방식임을 확인할 수 있었다.

VI. 결 론

컴퓨터 기반 시뮬레이터의 제작 시 발생하는 비행기의 운동을 표현하는 좌표계와 실제 비행기의 이동 경로 등을 표현하는 좌표계가 일치하지 않는다는 문제점을 가지고 있다. 따라서, 두 좌표계간의 일치 문제를 해결하기 위한 방법과 비행기의 이동 경로와 방위각을 연산하는 방법을 제시하였다.

두 좌표계간의 일치 방법은 동체 좌표계(body axes system)와 경·위도 좌표계를 축 회전 공식을 이용하여 해결하였다. 그리고, 경·위도 좌표계는 구 공간상에서 거리 및 방위각을 해석하기 위해 평면 근사화 방식을 이용하였다. 기존의 구 삼각 공식의 문제점은 cos, sin 및 실수 연산이 많음으로 평면 근사화 방법을 이용하여 연산량을 줄였다. 그리고, 40 NM이하에서는 실제 비행 결과 값을 구 삼각 공식의 결과 값과 평면 근사화 방법의 결과 값을 비교했을 때, 오차는 거리 상에서는 0.2NM 이하이고 방위각은 0.1~0.3도의 차이가 발생함을 확인하였다.

연산 속도 면에서는 구 삼각 공식을 사용 시 4.94505×10^{-4} 초가 소요되었고, 평면 근사화 방식을 사용 시에는 1.64835×10^{-4} 초가 소요되었다. 따라서,

표 3. 두 방식의 결과 비교

Table 3. Comparisons of the results between two methods.

비행기 현재 위치	도착지	FRASCA 142		구 삼각 공식		평면 근사화 방법	
		사거리 (NM)	방위각 (도)	사거리 (NM)	방위각 (도)	사거리 (NM)	방위각 (도)
N35 53.9 E128 38.6 (대구)	N35 58.5 E129 28.6 (포항)	40	90	39.97	89.54	38.19	89.46
N37 05.5 E127 01.8 (오산)	N37 33.2 E126 47.8 (김포)	30.2	344	30.26	345.2	30.24	345.0
N37 05.5 E127 01.8 (오산)	N37 24.7 E126 55.9 (안양)	19.9	354	20.2	353.2	20.17	353.13
N36 58.2 E127 01.2 (평택)	N37 05.5 E127 01.8 (오산)	7.2	11	7.32	10.75	7.32	10.76
N37 33.2 E126 47.8 (김포)	N37 24.7 E126 55.9 (안양)	10.8	149	10.63	151.04	10.64	150.87

속도 면에서 평면 근사화 방식이 더 우수함을 확인하였다.

참 고 문 헌

[1] 황명신 외 3명, "계기비행 훈련용 Mooney 시뮬레이터 개발," KACC, Oct. 1998.

[2] 백중환 외 3명, "비행기 시뮬레이터용 항공 전기/전자 모의실험 소프트웨어 개발(인터페이스 설계서 및 소스 프로그램 리스트)," 1994.

[3] 백중환, "항법 지리 데이터베이스를 이용한 비행 시뮬레이터용 항공전자 장치 소프트웨어 개발," 한국항공학회, 1997.

[4] 황명신의 3명, "계기비행 훈련용 Mooney 시뮬

레이터 개발", 한국항공대학교 항공산업기술 연구소, 1998.

[5] 김민경, "훈련용 계기비행 시뮬레이터의 개발", 학사졸업논문, 1999.

[6] Jan Roskam, "Airplane Flight Dynamics and Automatic Flight Controls," *DARP Corporation*, 1995.

[7] R. M. Howe, "Dynamics of Real-time Digital Simulation," Ann Arbor, Michigan, Sep., 1993.

[8] Robert C. Nelson, "Flight Stability and Automatic Control," McGraw-Hill, New York, 1989.

[9] "HIGH/LOW ALTITUDE ENROUTE CHARTS", Jeppesen sanderson INC. 1987.

백 중 환(白重煥)



1981년 2월 : 한국항공대학교 항공
공통신공학과(공학사)

1987년 7월 : 미국 오클라호마주
립대 전자공학과(공학석사)

1991년 7월 : 미국 오클라호마주
립대 전자공학과(공학박사)

1981년 6월~1984년 10월 : (주)

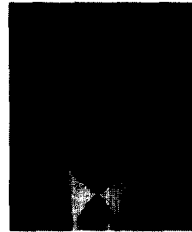
동아엔지니어링 근무

1991년 9월~1992년 2월 : 한국전자통신연구소 선임연구원

1992년 3월~현재 : 한국항공대학교 항공통신정보공학과 부교수

관심분야 : 컴퓨터비전, 영상처리, 영상압축 및 항공기 시뮬레이터 개발 등

황 수 찬(黃壽贊)



1984년 2월 : 서울대학교 전자계
산기공학과(공학사)

1986년 2월 : 서울대학교 컴퓨터
공학과(공학석사)

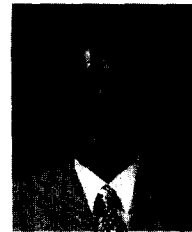
1991년 2월 : 서울대학교 컴퓨터
공학과(공학박사)

1995년~1996년 : 미국 캘리포니

아주립대 방문교수

1991년 3월~현재 : 한국항공대학교 컴퓨터공학과 부교수
관심분야 : 멀티미디어 데이터베이스, 객체지향 데이터베이스, 지리정보시스템 등임.

김 칠 영(金七永)



1978년 2월 : 한국항공대학교 항
공운항학과(이학사)

1991년 8월 : 인하대학교 경영대
학원 교통학과(경영학 석사)

1997년 8월 : 인하대학교 대학원
경영학과(경영학 박사)

1989년 3월~현재 : 한국항공대학

교 항공운항학과 부교수

관심분야 : 항공운항, 휴먼팩터 등임