

論文

J. of The Korean Society for Aeronautical and Space Sciences 45(4), 276-283(2017)

DOI:https://doi.org/10.5139/JKSAS.2017.45.4.276

ISSN 1225-1348(print), 2287-6871(online)

특이섭동 모델역변환을 이용한 멀티콥터 위치제어 연구

최형식*, 정연득*, 이장호*, 유혁*, 이상중**

Multicopter Position Control using Singular Perturbation based Dynamic Model Inversion

Hyoung Sik Choi*, Yeondeuk Jung*, Jangho Lee*, Hyeok Ryu* and Sangjong Lee**

Flight Control Research Team, Korea Aerospace Research Institute*

Department of Aircraft Mechanical Engineering, Inha Technical College**

ABSTRACT

This paper presents position control of multicopter using nonlinear dynamic model inversion in singular perturbation. Multicopter dynamics are developed and separated into the fast time-scale variables, related with the inner-loop design, and the slow time-scale variables, related with the outer-loop design. The final design is evaluated in 6-DOF simulation. The results show accurate position tracking performance.

초 록

본 논문은 4개의 로터를 가지는 쿼드콥터에 대한 비선형 운동모델을 구성하고, 특이섭동 기법을 이용한 모델 역변환 위치 제어기 설계 결과에 대해 나타낸다. 특이섭동 모델 역변환 방식은 느린 동역학과 빠른 동역학을 시분할 기법(time scale separation)기법을 이용하여 각각 역변환 시키는 방법이다. 수립한 6자유도 비선형 운동모델 기반으로, 모델 역변환 제어를 설계하고 시뮬레이션을 수행한 결과 정확한 위치 추종을 수행함을 확인하였다.

Key Words : Drone(드론), Multicopter(멀티콥터), Singular Perturbation(특이섭동), DMI (모델 역변환), Two-time scale(시분할)

1. 서 론

최근 드론이 부각되면서 여러 개의 로터를 가지는 멀티콥터가 널리 사용되고 있다. 이러한 연구는 2003년을 기점으로[1] 주로 자율 시스템을 다루는 연구실에서 시작되었으며, 관련 선행 연구는 참고문헌[2]에 잘 정리되어 있다. 한편, 최근 대두되고 있는 충돌회피 기능의 경우 긴급한 회피기동을 적절히 수행하기 위해서는 고기동,

고성능의 제어방식이 요구 되고 있다. 이러한 고성능, 고기동 제어법칙 연구에는 비선형 모델 역변환(Nonlinear Dynamic Model Inversion) 제어법칙이 주로 연구되었고[3~7], 본 논문은 비선형 모델역변환 기법 중의 하나인 특이섭동 기법(Singular perturbation)을 쿼드콥터에 적용해 보 고자 한다. 특이섭동 모델 역변환은 시분할(time scale separation) 기법을 이용하여 느린 상태와 빠른 상태를 분리하여 각각 역변환 시켜 통합하

† Received : May 11, 2016 Revised : January 26, 2017 Accepted : March 8, 2017

** Corresponding author, E-mail : leesj@inhac.ac.kr

는 방식으로[8~10], 항공분야의 경우 먼저 고정익에 대한 연구가 발표 되었고[10~12], 최근 멀티콥터에 유사한 개념으로 적용된 바 있다[13,14].

본 논문은 항공 제어분야에 익숙한 좌표계를 기준으로 비선형 6자유도 운동방정식을 구성하여 쿼드콥터에 대한 운동모델을 수립하였으며, 수식의 복잡도를 낮추기 위해 벡터, 행렬식으로 표현하고 모델 역변환기법의 흐름을 따라가기 쉽도록 순차적으로 정리하였다. 마지막으로 특이섭동 모델역변환 제어기의 위치제어 시뮬레이션을 수행하여 제어특성을 살펴보았다.

II. 운동 모델

좌표계는 Fig. 1과 같이 전통적으로 비행제어 분야에서 사용하는 기준 좌표계를 사용하였다. 여기서, l 은 무게중심 기준 축으로부터 로터의 거리를 나타내며 로터 모두 동일한 거리에 배치된다고 가정한다. 로터의 배치를 Fig. 1과 같이 구성할 때 조종입력에 따른 각각의 모터 제어를 위한 조종할당 관계는 식 (1)과 같이 정의 할 수 있다. \bar{M} 은 믹싱 테이블이다.

$$\delta \rightarrow \bar{M}u, \quad \bar{M} = \begin{bmatrix} +1 & +1 & +1 & +1 \\ +1 & -1 & -1 & +1 \\ -1 & -1 & +1 & +1 \\ +1 & -1 & +1 & -1 \end{bmatrix} \quad (1)$$

여기서, δ 는 조종입력, u 는 모터입력이다.

$$\delta = [\delta_{col}, \delta_{pit}, \delta_{rol}, \delta_{yaw}]^T, \\ u = [u_1, u_2, u_3, u_4]^T$$

식 (1)의 테이블을 이용하여 조종입력 δ 에 따른 모터 배분을 식 (2)와 같이 표현할 수 있다.

$$u = \bar{M}^T \delta \quad (2)$$

δ_{col} 은 네 개의 로터를 모두(collective) 조종하

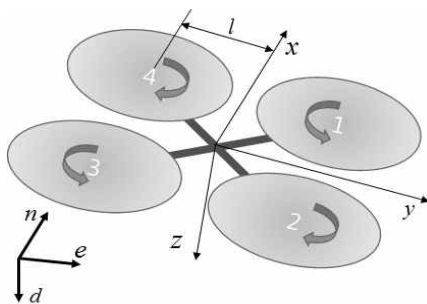


Fig. 1. Coordinates system

여 고도 상승, 하강을 제어 할 수 있고 상승을 (+) 부호로 정의한다. δ_{pit} 는 피치(θ)제어로 기수 들림 방향을 (+) 명령으로 정의하며, δ_{rol} 은 롤(ϕ) 제어로 우측이 내려가는 방향을 (+)로 정의한다. δ_{yaw} 는 요(ψ) 제어 명령으로 기수가 시계방향으로 회전하는 방향을 (+)로 정의한다.

네 개의 프로펠러는 동일한 추력, 토크 특성을 가지는 것으로 가정하고, 식 (3)과 같이 비례식으로 간략화 하였다. 이때 추력은 윗방향이 (+), 토크는 로터 각각의 회전방향의 반대 방향이 (+)이다. 여기서, 4개의 모터 입력 $u_i(i=1,2,3,4)$ 는 식 (2)로부터 계산되며 κ 와 η 는 각각 추력, 토크 비례 상수이다.

$$T_i = \kappa u_i, \quad Q_i = \eta u_i \quad (3)$$

2.1 힘과 모멘트

식 (3)을 적용하여 모터 제어에 따른 힘과 모멘트는 다음과 같이 나타낼 수 있다.

$$\begin{bmatrix} F_{col} \\ M_{pit} \\ M_{rol} \\ M_{yaw} \end{bmatrix} = \begin{bmatrix} +\kappa & +\kappa & +\kappa & +\kappa \\ +\kappa l & -\kappa l & -\kappa l & +\kappa l \\ -\kappa l & -\kappa l & +\kappa l & +\kappa l \\ +\eta & -\eta & +\eta & -\eta \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}$$

또는,

$$FM = \bar{K} \bar{M} u \quad (4)$$

$$FM = [F_{col}, M_{pit}, M_{rol}, M_{yaw}]^T, \\ \bar{K} = \text{diag}([\kappa, \kappa l, \kappa l, \eta])$$

따라서 동체 좌표계 기준 6자유도 힘과 모멘트는 식 (5)와 같이 나타낼 수 있다. 여기서, 저속비행의 경우 동체 항력은 무시할 만하므로 F_x, F_y 는 0으로 둔 것이다.

$$F_b = [0, 0, -T]^T, \quad T = F_{col} \\ M_b = [M_{rol}, M_{pit}, M_{yaw}]^T \quad (5)$$

한편, 식 (4)를 조종입력 δ 에 대한 식으로 나타내면 식 (6)과 같다.

$$FM = \bar{K} \bar{M} \bar{M}^T \delta \quad (6)$$

결국, 동체에 전달되는 힘과 모멘트는 식 (6)으로부터 식 (7)과 같이 나타낼 수 있으며, 조종입력 δ 에 따른 힘과 모멘트는 κ, η, l 를 계수로 사용하는 비례식으로 간략화 할 수 있다[15].

$$\begin{aligned}
 F_{col} &= 4 \kappa \delta_{col} \\
 M_{pit} &= 4 \kappa l \delta_{pit} \\
 M_{rol} &= 4 \kappa l \delta_{rol} \\
 M_{yaw} &= 4 \eta \delta_{yaw}
 \end{aligned}
 \tag{7}$$

$$[T_e^b] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

식 (11)과 같은 특성을 가진다.

$$[T_b^e] = [T_e^b]^T, \quad [\dot{T}_b^e] = [T_b^e] \Omega \tag{11}$$

2.2 6자유도 운동방정식

비선형 6자유도 강체 운동방정식은 다음과 같다[16]. 하첨자 b 는 동체 좌표계 기준을 의미하고, F_b, M_b 는 앞의 식 (5)에서 계산한 로터 힘과 모멘트이다.

- **Force Equation**

$$\dot{v}_b = -\Omega v_b + [T_e^b] \bar{g} + F_b/m \tag{8}$$

여기서, $v_b = [u, v, w]^T$, $\Omega = (\omega_b \times)$,

중력가속도 : $\bar{g} = [0, 0, g]^T$

- **Moment Equation**

$$I_b \dot{\omega}_b = -\Omega I_b \omega_b + M_b \tag{9}$$

여기서, $\omega_b = [p, q, r]^T$, I_b : 관성모멘트

- **Navigation Equation**

$$\dot{P} = [T_e^b]^T v_b \tag{10}$$

여기서, $P = [P_n, P_e, P_d]^T$

지면좌표를 동체좌표로 변환하는 행렬 $[T_e^b]$ 은 다음과 같고,

$$\Omega = (\omega_b \times) = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix}$$

III. 모델 역변환 제어기

특이섭동(singular perturbation) 모델역변환 기법은 시분할 기법(time scale separation)을 이용하여 느린 동역학과 빠른 동역학을 분리하여 각각 역변환 시키는 방법으로[10], 항법 방정식(Navigation Equation)은 느린 동역학, 모멘트 방정식(Moment Equation)은 빠른 동역학으로 볼 수 있다. 다음 3.1절 부터 상세한 특이섭동 모델 역변환 기법의 수학적 흐름을 서술하고, 전체적인 개념은 한눈에 보기 쉽도록 Fig. 2에 블록선도로 나타내었다.

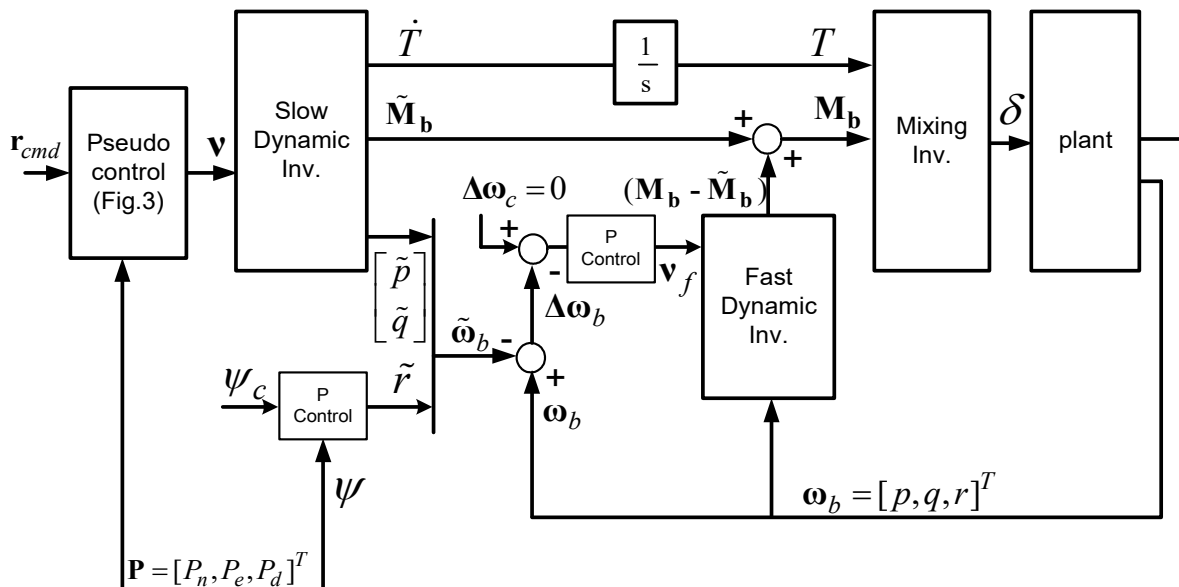


Fig. 2. Multicopter model inversion control diagram

3.1 Time scale separation

모델역변환은 필요한 제어명령을 생성하기 위해 운동방정식을 미분하여 사용하는 경우가 많은데, 본 논문에서도 마찬가지로 미분을 취하여 모델 역변환을 위한 운동모델을 구성한다. 식 (10)의 항법 방정식을 미분하면 다음과 같고,

$$\begin{aligned} \ddot{P} &= [\dot{T}_b^e]v_b + [T_b^e]\dot{v}_b \\ &= [T_b^e]\Omega v_b + [T_b^e](-\Omega v_b + [T_e^b]\bar{g} + F_b/m) \end{aligned}$$

정리하면, 식 (12)와 같다. P 는 지면좌표 기준 비행체 위치벡터이다.

$$\ddot{P} = [T_b^e]F_b/m + \bar{g} \quad (12)$$

$U = \ddot{P}$ 로 두고 식 (12)를 한 번 더 미분하면 다음과 같다.

$$\begin{aligned} \dot{U} &= [\dot{T}_b^e]F_b/m + [T_b^e]\dot{F}_b/m \\ &= [T_b^e]\Omega F_b/m + [T_b^e]\dot{F}_b/m \end{aligned}$$

여기서, $F_b = [0, 0, -T]^T$ 및 cross product matrix Ω 를 대입하면 다음과 같다.

$$\begin{aligned} \dot{U} &= [T_b^e]/m \left(\begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ -T \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -\dot{T} \end{bmatrix} \right) \\ \text{또는, } \dot{U} &= [T_b^e]/m \begin{bmatrix} -Tq \\ Tp \\ -T \end{bmatrix} \end{aligned}$$

위 식을 다시 쓰면 식 (13)과 같다[13].

$$\begin{aligned} \dot{U} &= [T_b^e]Az \quad (13) \\ A &= \begin{bmatrix} -T/m & 0 & 0 \\ 0 & T/m & 0 \\ 0 & 0 & -1/m \end{bmatrix}, \quad z = [q, p, \dot{T}]^T \end{aligned}$$

여기서, U 는 모델역변환을 통해 달성하고자 하는 최종 제어 목표인 상태변수가 된다. 한편, 식 (13)에서 $T \approx mg$ 이므로 A 행렬은 다음과 같이 들 수 있다.

$$A = \begin{bmatrix} -g & 0 & 0 \\ 0 & g & 0 \\ 0 & 0 & -1/m \end{bmatrix}$$

결국, 식 (13)과 식 (9)로부터 특이섭동 모델역변환을 위한 운동모델은 식 (14)와 같이 구성할 수 있다.

$$\begin{cases} \nu = [T_b^e]Az & : \text{slow} \\ \epsilon I_b \dot{\omega}_b = -\Omega I_b \omega_b + M_b & : \text{fast} \end{cases} \quad (14)$$

여기서, ν 는 의사입력(pseudo control)으로 $\nu = \dot{U}$ 로 둔 것이고, 3.3절에서 자세히 나타내었다.

그리고 ϵ 은 특이섭동(singular perturbation)을 위한 시분할(time-scale separation) 파라미터로, 0일 때는 느린 시간스케일(slow time scale)에서의 빠른 동역학(fast dynamics) 특성을 의미하고, 1일 때는 빠른 시간스케일(fast time scale)내에서의 동역학 특성을 의미한다[10,11].

3.2 Slow time scale

식 (14)에서 나타낸 두 개의 시분할(slow and fast) 모델을 이용하여 느린 시간스케일(slow time scale)에 해당하는 식으로 나타내면 식 (15)와 같다. 느린 시간스케일(slow time scale)내에서 빠른 상태변수(fast state) p, q, r 은 정상상태(steady state)로 가정한 것으로, (\sim)로 표현하였으며 그 미분은 0으로 취한다. 즉, $\epsilon = 0$.

$$\begin{cases} \nu = [T_b^e]A\tilde{z}, \quad \tilde{z} = [\tilde{q}, \tilde{p}, \dot{T}]^T \\ \epsilon I_b \dot{\tilde{\omega}}_b = -\tilde{\Omega} I_b \tilde{\omega}_b + \tilde{M}_b, \quad \epsilon = 0 \end{cases} \quad (15)$$

식 (15)의 역변환은 다음과 같고,

$$\begin{cases} \tilde{z} = A^{-1}[T_b^e]\nu & : \text{slow} \\ \tilde{M}_b = \tilde{\Omega} I_b \tilde{\omega}_b & : \text{fast} \end{cases}$$

풀어서 나타내면, 식 (16), (17)과 같다.

$$\begin{bmatrix} \tilde{q} \\ \tilde{p} \\ \dot{T} \end{bmatrix} = \begin{bmatrix} -g & 0 & 0 \\ 0 & g & 0 \\ 0 & 0 & -1/m \end{bmatrix}^{-1} [T_b^e] \begin{bmatrix} \nu_x \\ \nu_y \\ \nu_z \end{bmatrix} \quad (16)$$

$$\begin{bmatrix} \tilde{M}_x \\ \tilde{M}_y \\ \tilde{M}_z \end{bmatrix} = \begin{bmatrix} (-I_y + I_z)\tilde{r}\tilde{q} \\ (I_x - I_z)\tilde{p}\tilde{r} \\ (-I_x + I_y)\tilde{p}\tilde{q} \end{bmatrix} \quad (17)$$

식 (16)의 slow 방정식은 \tilde{p}, \tilde{q} 및 \dot{T} 를 계산하는데, 위치 및 고도제어에 필요한 피치 각속도, 롤 각속도 및 추력항이다. 이때, 추력항 \dot{T} 은 적분을 통해 $T (= F_{col})$ 를 직접 계산하고, 식 (7)로부터 필요 조종입력 δ_{col} 을 구할 수 있다.

식 (17)의 fast 방정식은 $\tilde{p}, \tilde{q}, \tilde{r}$ 값을 이용하여 조종모멘트 $\tilde{M}_x, \tilde{M}_y, \tilde{M}_z$ 를 계산한다. \tilde{p}, \tilde{q} 는 식 (16)으로부터 계산되며, \tilde{r} 은 요 제어에 필요한 값으로, $\tilde{r} = K_\psi(\psi_c - \psi)$ 와 같이 간단한 비례제어

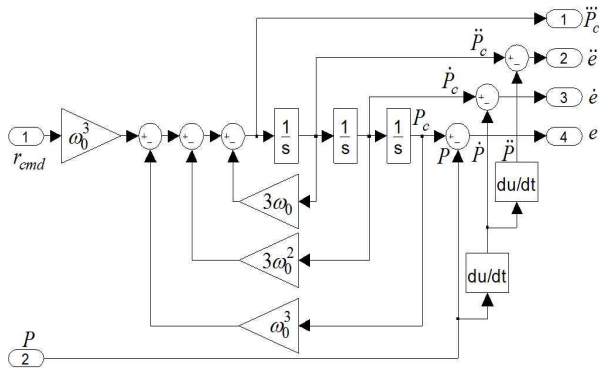


Fig. 3. Command shaping (reference model)

(P control) 방식으로 계산 할 수 있다. 참고로, 이는 1차 기준모델(reference model)을 사용한 것과 동일한 효과를 나타낸다[14].

한편, 식 (16)의 의사입력(pseudo control) ν 는 다음과 같이 설계할 수 있다[13,14].

3.3 Command Shaping

오차 동역학 모델(desired error dynamics)을 식 (18)과 같이 안정한 3차 미분 방정식으로 정의 할 수 있다. 모델 역변환 제어기법의 경우 역변환이 정상적으로 동작할 경우 기준모델을 그대로 추종하기 때문에 오차모델 역시 안정한 모델로 선정하면 전체 시스템은 안정하게 된다.

$$\ddot{e} + 3\omega_0\dot{e} + 3\omega_0^2e + \omega_0^3e = 0, \quad e = P_c - P \quad (18)$$

식 (18)로부터 $\nu = \dot{U}$ 로 두면, 식 (19)와 같이 나타낼 수 있다.

$$\nu = \ddot{P}_c + 3\omega_0\dot{P}_c + 3\omega_0^2P_c + \omega_0^3P_c \quad (19)$$

식 (19)의 P_c 는 식 (20)과 같이 오차모델과 동일한 특성을 가지는 3차 기준모델을 사용하여 만들어낼 수 있다. 여기서, r_{cmd} 은 말단 위치 명령, P_c 는 가공된 위치명령이다. 결국, 의사입력(pseudo control) ν 는 Fig. 3의 출력을 이용하여 계산한다.

$$\ddot{P}_c + 3\omega_0\dot{P}_c + 3\omega_0^2P_c + \omega_0^3P_c = \omega_0^3r_{cmd} \quad (20)$$

3.4 Fast time scale

이번에는 느린 시간 스케일(slow time scale) 내에서 정상상태($\tilde{\omega}_b$)로 가정하였던 식 (15)의 빠른 동역학과 원래의 빠른 동역학인 식 (9)의 차이를 계산하고, 그 차이($\Delta\omega = \omega_b - \tilde{\omega}_b$)가 0이 되도록 하는 제어기를 설계한다.

두 식을 묶어서 나타내면 식 (21)과 같다. 이때

$\epsilon = 1$ 이다.

$$\begin{cases} I_b\dot{\omega}_b = -\Omega I_b\omega_b + M_b \\ \epsilon I_b\dot{\tilde{\omega}}_b = -\tilde{\Omega} I_b\tilde{\omega}_b + \tilde{M}_b, \quad \epsilon = 1 \end{cases} \quad (21)$$

식 (21)의 차이를 나타내면, 식 (22)와 같다.

$$I_b\Delta\dot{\omega}_b = -\Delta\Omega I_b\Delta\omega_b + \Delta M_b \quad (22)$$

여기서, $\Delta\omega_b = (\omega_b - \tilde{\omega}_b)$, $\Delta M_b = (M_b - \tilde{M}_b)$, $\Delta\Omega = (\Omega - \tilde{\Omega})$.

또한, 식 (22)의 미분항 $\Delta\dot{\omega}_b$ 를 의사입력(pseudo control) ν_f 로 두고 역변환 하면 식 (23)과 같다.

$$\Delta M_b = \nu_f I_b + \Delta\Omega I_b\Delta\omega_b \quad (23)$$

여기서, $\nu_f = K_p(\Delta\omega_c - \Delta\omega_b)$ 로 설계 할 수 있는데 이는 앞의 \tilde{r} 계산과 동일한 방식이며, $\Delta\omega_c$ 는 $\Delta p, \Delta q, \Delta r$ 의 명령으로 모두 0 값이 입력된다. 이는 $\Delta p, \Delta q, \Delta r$ 이 0 으로 수렴 하도록 하는 의도이다.

결국, $M_b = \Delta M_b + \tilde{M}_b$ 이므로 식 (23)과 (17)로부터 필요 조종모멘트 M_b 를 구하고, 이를 식 (7)에 대입하여 실제 필요한 조종입력 $\delta_{pit}, \delta_{rol}, \delta_{yaw}$ 를 구할 수 있게 된다.

IV. 시뮬레이션

시뮬레이션은 MATLAB/Simulink로 구성하였고 time step은 0.01초, solver는 euler로 설정하였다. 방위각 명령을 0으로 유지하고 목표점을 이동시킨 시뮬레이션 결과로, Table 1은 시뮬레이션에서 사용한 파라미터를 나타낸다.

Table 1. Simulation parameters

Par	Value	Description	Unit
m	1.0	mass	kg
g	9.8	gravity const.	m/s^2
I_x, I_y	0.01	moment of inertia	kgm^2
I_z	0.02	moment of inertia	kgm^2
l	0.2	rotor distance	m
k	4.9	thrust coeff.	[-]
η	0.196	torque coeff.	[-]
ω_0	1.5	desired frequency	rad/s
K_p, K_ψ	10.0	P control gain	[-]

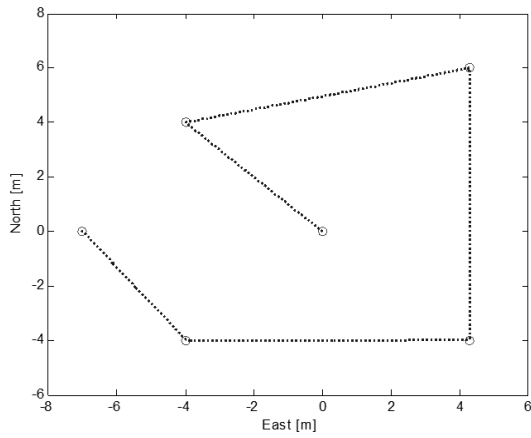


Fig. 4. Position control trajectory

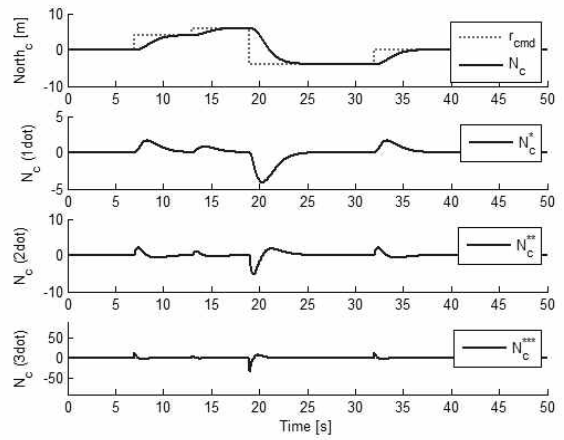


Fig. 7. Command shaping (north)

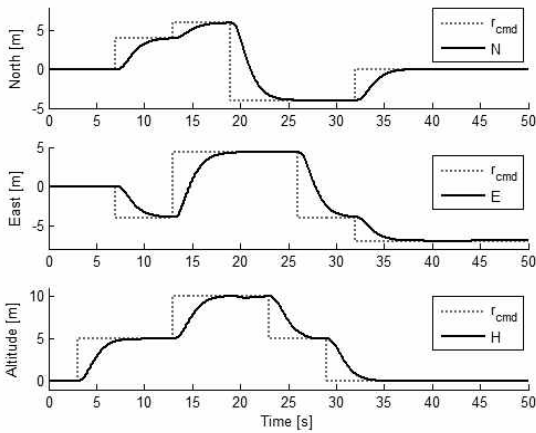


Fig. 5. Position command and response

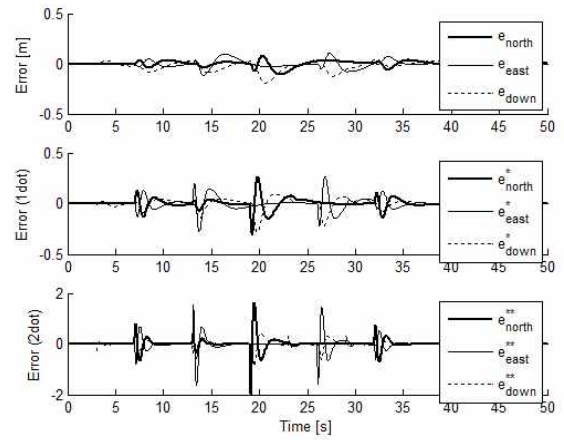


Fig. 8. Error dynamic response(n, e, d)

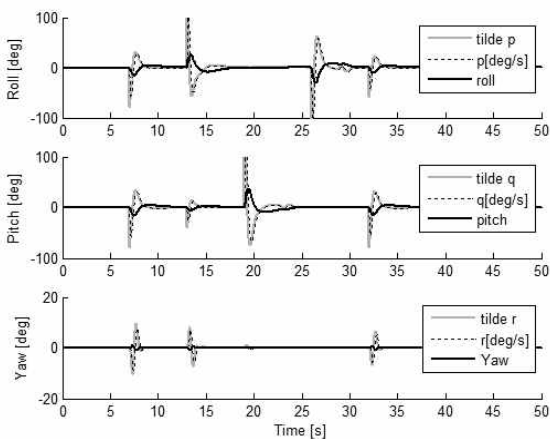


Fig. 6. Attitude and rate response

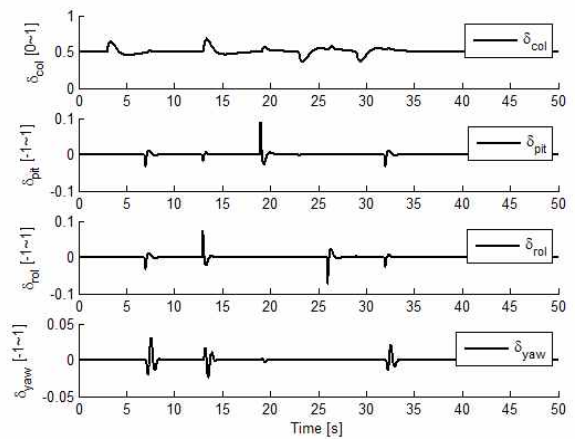


Fig. 9. Control inputs

Figure 4는 위치제어 목표점 및 수평이동 경로를 나타내고, Fig. 5는 위치 및 고도 명령에 따른 응답을 나타낸다. 위치 이동 및 목표점 도달

과도 응답 없이 잘 추종함을 보이고 있다.

Figure 6은 자세 및 변화율을 나타내고, $\tilde{p}, \tilde{q}, \tilde{r}$ 을 이내 추종하며 안정적인 응답을 보이고 있다.

Fig. 7은 식 (20)의 3차 기준모델의 명령(P_c)을 나타내며 북쪽 명령을 대표로 나타내었다. 말단 명령(r_{cmd})을 부드럽게 가공하는 역할을 수행함을 보이고 있다. Fig. 8은 식 (18)의 오차(e)를 나타내며, Fig. 9는 사용된 조종입력을 나타내고 있다. 제한 범위 내에서 적절히 동작함을 보이고 있다.

V. 결 론

본 논문은 쿼드콥터 비행체에 대한 비선형 6자유도 모델을 간결한 형태로 수립하고, 로터의 배치와 믹싱에 따른 힘과 모멘트를 간단한 비례관계식으로 도출하였다. 이를 기반으로, 시분할 기법을 사용하는 비선형 특이섭동 모델역변환 제어기의 설계 개념을 적용하여 위치제어 시뮬레이션을 수행해 보았으며 그 결과에 대해 다음과 같이 분석하였다.

Figure 4, 5에서 나타난 바와 같이 위치 제어 시뮬레이션 결과 과도 응답 없이 목표점을 정확히 추종함을 확인할 수 있으며, 이는 충돌회피를 위한 위치이동 명령을 신속하고 정확히 추종할 수 있을 것으로 기대할 수 있다.

논문에서 나타난 제어법칙의 특징으로는 항법 방정식의 두 번 미분으로부터 식(13)과 같이 q, p, \dot{T} 등 3개의 식이 하나의 행렬식으로 조합되어 모델 역변환이 용이한 형태로 나타나는데, 이는 수학적으로 검토해 볼 때 쿼드콥터뿐만 아니라 유사한 형태의 여러 개의 로터를 가지는 멀티콥터라면 모두 적용 가능함을 알 수 있다.

또 다른 특징으로는 자세제어기(SCAS)를 별도로 두지 않는데, 그렇다고 해서 자세 측정치를 제어에 사용하지 않는 것이 아님에 유의할 필요가 있다. 즉, 블록선도에서 볼 때 자세값은 나타나 있지 않지만 식 (13)에서처럼 지면좌표계와 동체좌표계간 좌표변환을 위해서는 자세 측정값이 필요하다.

또한, Fig. 3에서 볼 때 위치를 두 번 미분한 값인 지면좌표계 기준 가속도가 피드백에 쓰이게 되어 있는데, 실제 비행의 경우 위치 측정값의 정확도, 연속성 등을 고려하여 적절한 필터링이 필요할 것으로 예상된다.

마지막으로, 본 논문에서 나타난 비선형 운동모델 기반의 수학적 역변환 설계 방법은 기존 제어법칙의 개인 설계 방법을 개선하거나, 이미 사용 중인 제어법칙과 결합하는 방식을 사용하여 보다 향상된 제어 성능을 확보하는 방안으로 응용될 수도 있을 것으로 기대한다.

후 기

본 논문은 한국항공우주연구원 주요사업 "비행상황 인식 및 자율대응 기술연구"의 지원을 받아 수행한 연구입니다. 지원에 감사드립니다.

References

- 1) S. Bouabdallah, R. Siegwart, "Full control of Quadrotor," International Conference on Intelligent Robots and Systems, 2007.
- 2) S. Bouabdallah, "Design and control of quadrotors with application to autonomous flying," Ph. D. dissertation, EPFL, 2006.
- 3) Meyer, G. and Cicolani, L., "Application of Nonlinear Systems Inverses to Automatic Flight Control Design," Theory and Application of Optimal Control in Aerospace Systems, AG251, 1981, pp.1-29
- 4) Bugajski, D. J. and Enns, D. F., "Nonlinear Control Law with Application to High Angle-of-Attack Flight," Journal of Guidance, Control, and Dynamics, vol.15, no.3, 1992, pp.761-767
- 5) Kim, B. S. and Calise, A.J., "Nonlinear Flight Control Using Neural Networks," Proc. of the AIAA Guidance, Navigation, and Control Conference, Scottsdale, AZ, 1994, pp.930-940
- 6) Reiner, J., Balas, G.J., and Garrard, W.L., "Robust Dynamic Inversion for Control of Highly Maneuverable Aircraft," Journal of Guidance, Control, and Dynamics, vol.18, no.1, 1995, pp.18-24
- 7) S. A. Snell, D. F. Enns, W. L. Garrard, "Nonlinear Inversion Flight Control for a Super maneuverable Aircraft", J. of. Guidance and Dynamics, Vol. 15, n0. 4, 1992.
- 8) Hassan K. Khalil "Nonlinear Systems" 3rd edition, Prentice Hall, pp.424.
- 9) J. H. Chow, and P. V. Kokotovic, "Two-time scale feedback design of a class of nonlinear systems." IEEE Transactions on Automatic Control, vol. AC-23, no. 3, pp.438-443, June 1978.
- 10) P. K. A Menon, M. E. Badgett, and R. A. Walker, "Nonlinear Flight Test Trajectory Controllers for Aircraft", J. Guidance, vol.10,

no.1 , Jan-Feb. 1987, pp. 67-72

11) Ha, C. K., Choi, H. S., "Design of Autolanding Guidance and Control Algorithm using Singular Perturbation" Journal of Institute of Control, 2005.

12) Choi, H. S. "Design of Auto-landing Guidance and Control System using Model inversion" M. S. thesis, 2001.

13) Jian Wang, et al, "Attitude Free Position Control of a Quadcopter using Dynamic

Inversion", AIAA 2011-1583. 2011.

14) Jian Wang, "Novel Control Approaches to Quadrotors Inspired by Dynamic Inversion and Backstepping", ph. D thesis. 2014.

15) Choi, H. S., Jung, Y. D., Lee, J., "Multi-copter Modeling and Simulation", Korea Society for Simulation Conference, Apr, 2016.

16) Brian L. Stevens and Frank L. Lewis "Aircraft control and Simulation," JOHN WILEY & SONS, INC , 1992. pp.7~81.