

論文

DOI:10.5139/JKSAS.2010.38.4.327

항공기의 수평속도에 대한 비행영역 보호 시스템 설계

신호현*, 이상현*, 김유단**, 김응태***, 성기정***, 최형식***

Design of Flight Envelope Protection System on Velocity of Aircraft

Hohyun Shin*, Sanghyun Lee*, Youdan Kim**,
Eung Tae Kim***, Ki Jung Seong*** and Hyoung Sik Choi***

ABSTRACT

Recently developed aircrafts use Fly-By-Wire(FBW) or Fly-By-Light(FBL) system. These systems have some merits; they can perform very complicated missions, they can expand the flight region and improve the reliability of the aircrafts. With the development of flight control systems that use FBW technique, flight envelope protection concept is introduced to guarantee reliability of the aircraft and improve the efficiency of mission achievement. In this study, flight envelope protection system is designed using a dynamic trim algorithm, a peak response estimation, and a gain scheduling technique. The performance of these methods are compared by performing numerical simulation.

초 록

최근 개발된 항공기는 대부분 디지털 Fly-By-Wire(FBW) 혹은 광신호를 사용하는 Fly-By-Light(FBL) 시스템을 사용하고 있다. 이러한 시스템들은 조종사의 직접 조종으로는 불가능하거나 어려운 복잡한 임무의 수행, 비행범위 확장, 신뢰성 향상 등의 장점을 가지고 있다. FBW 기술을 적용한 비행제어기술이 발전됨에 따라 운항 시 안정성 확보 및 보다 효율적인 임무수행을 위해 비행영역(Flight Envelope) 보호의 개념이 항공기 설계에 있어 중요한 문제로 부각되고 있다. 본 논문에서는 dynamic trim 알고리즘, peak response estimation, 제어이득 스케줄링 등의 방법을 이용해서 항공기의 비행영역 보호를 수행하고, 각 방법의 성능을 비교하여 최적의 제어기를 설계하는 연구를 수행하였다.

Key Words : Flight Envelope Protection(비행영역 보호), Dynamic Trim Algorithm (Dynamic Trim 알고리즘), Peak Response Estimation(Peak Response Estimation), Gain Scheduling(제어이득 스케줄링)

1. 서 론

1980년대까지 개발된 항공기는 대부분 조종간과 조종면이 유압장치를 통해 연결되어 있는 방식이었다. 이러한 시스템에서는 항공기의 운용 폭에 비례해서 조종사가 스틱에 가하는 힘 또한 더 크게 필요하였으며, 따라서 고성능 전투기와 같이 고기동을 위해 안정성을 완화시켜 기체를 설계한 항공기의 경우, 심한 기동을 하게 되면

† 2009년 10월 19일 접수 ~ 2010년 3월 17일 심사완료

* 정회원, 서울대학교 기계항공공학부 대학원

** 정회원, 서울대학교 기계항공공학부

교신저자, E-mail: ydkim@snu.ac.kr

서울특별시 관악구 신림동 산56-1

*** 정회원, 한국항공우주연구원 첨단 비행제어팀

조종사가 제어하기 어려운 상황에 이를 수 있었다. 이러한 문제를 해결하기 위해 전기적 신호로 조종계통을 제어하는 Fly-By-Wire(FBW) 방식이 개발되었다. 최근에 개발되는 대부분의 항공기는 FBW 방식 또는 광신호에 의해 작동되는 Fly-By-Light(FBL) 방식을 사용하고 있다. 이러한 방식들은 안정성 및 조종성이 기존의 항공기에 비해서 높으며, 다중화 구조를 취함으로 인해서 비행 신뢰성이 증대된다는 장점이 있다. 또한, 비행제어와 관련된 모든 사항이 비행제어 컴퓨터의 소프트웨어로 통제되므로, 항공기에 요구되는 변경사항을 소프트웨어의 수정만으로 유연하게 대처할 수 있다.

Fly-By-Wire 기술이 발전됨에 따라서 '비행영역 보호'(Flight Envelope Protection) 개념이 항공기 설계에 있어 중요하게 부각되고 있다.[1] 비행영역 보호란 조종사가 구조적 한계, 공기역학적 한계, 또는 제어입력 한계를 초과하지 않고 항공기 고유의 비행영역 내에서 자유롭게 비행을 할 수 있도록 보호하는 것을 의미한다. 비행영역 보호 기술이 적용된 시스템은 군용 항공기뿐만 아니라 대형 민간 여객기에도 사용되고 있다. 이러한 기술을 적용하면 항공기 운영 도중 비행사고 감소, 임무시간 단축, 조종사의 부하 감소 등의 장점이 있다.

본 논문에서는 소형 항공기 모델에 대해서 dynamic trim 알고리즘, peak response estimation, 그리고 제어 이득 스케줄링 등의 방법을 이용해서 비행영역 보호 시스템을 설계하고, 이를 Eclipse 500 항공기의 비선형 시뮬레이션을 통해 성능을 비교/검증하였다.

II. 본 론

2.1 비행영역 보호

비행영역은 항공기가 안전하게 운영할 수 있는 영역을 의미한다. 비행영역은 마하수(속도), 고도, 하중배수 한계 등에 의해서 정의되며, 영역의 경계는 항공기의 실속한계, 추력한계, 구조적 한계 등과 밀접하게 연관되어 있다. 항공기가 정해진 비행영역을 벗어나면 조종성 상실(Loss of Control) 상태에 빠질 수 있다. 조종성 상실이란 항공기가 조종사의 조종입력대로 기동하지 않는 상태로 주로 실속(stall)에 의해 발생하며, 옆미끄럼각, PIO(Pilot induced oscillation), 또는 기타 원인으로 인한 롤 모멘트 발생 등이 원인이 된다 [1].

이러한 문제들을 방지하기 위해서는 항공기가 항상 안전한 비행영역 내에서 비행하도록 하는 것이 필요한데, 이를 위해서 사용되는 것이 비행영역 보호 알고리즘이다. 비행영역 보호 알고리즘은 크게 한계예측(limit prediction)과 신호생성(signal generation)의 두 부분으로 구성된다.

한계예측 부분에서는 현재의 비행상태와 제어입력 등의 요인을 고려해서 항공기가 안전한 영역을 벗어나지 않도록 하는 입력의 한계점을 예측하게 된다. 이를 위해서 사용되는 방법은 dynamic trim 알고리즘[2], peak response estimation, adaptive neural network based[3,4] 알고리즘 등이 있다.

신호생성 부분에서는 한계예측을 통해서 계산된 한계 입력값을 이용해서 항공기 혹은 조종사에게 신호를 보내게 된다. 그림 1-a에서 보여주고 있는 것과 같이 Hard envelope protection의 경우에는 한계 입력값 u_{ref} 을 바로 비행제어 컴퓨터를 거쳐 조종면으로 보내게 되며, 그림 1-b의 soft envelope protection의 경우에는 스틱을 통한 force cue나 시각, 청각적인 신호 δ 를 통해서 조종사에게 경고를 하게 된다. Soft envelope protection의 경우, 조종사에게 신호를 주어 비행상황을 판단하게 하기 때문에 구성된 알고리즘의 성능을 객관적으로 판단하기가 힘들다. 본 연구에서는 성능을 정량적으로 평가할 수 있는 hard envelope protection 형태의 다양한 비행영역 보호 시스템을 설계하고, 성능을 비교하였다.

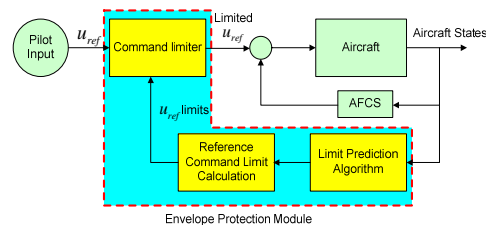


그림 1-a. hard envelope protection 시스템의 구조

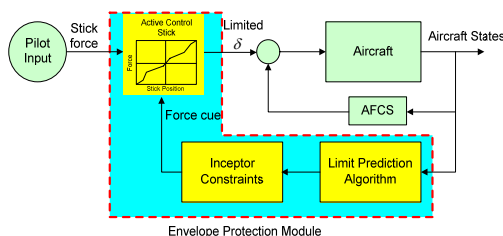


그림 1-b. soft envelope protection 시스템의 구조

2.2 Dynamic Trim 알고리즘

Dynamic trim 알고리즘은 준정상(quasi-steady) 비행 기동상황에 대한 개념을 이용하여 비행영역 한계점을 초과하게 되는 제어입력을 추정하는 방식이다. 항공기의 상태변수들은 빠른 동역학(fast dynamics)과 느린 동역학(slow dynamics)을 갖는 변수들로 구분할 수 있다. 느린 동역학을 갖는 변수는 빠른 동역학을 갖는 변수와 비교해서 변화폭이 상당히 작기 때문에 짧은 시간 변화에 대해서는 거의 변화수는 0 값을 갖는다고 가정할 수 있다. Dynamic trim 알고리즘은 느린 동역학을 갖는 상태변수들에 적합한 알고리즘으로, 상태변수들의 시간에 대한 미분값을 모두 0이라고 가정하며, 이를 식으로 표현하면 다음과 같다.

$$y' = y^{(2)} = \dots = y^{(n)} = 0 \tag{1}$$

그러나 느린 동역학을 갖는 상태변수도 시간에 대한 미분값이 정확히 0이 아니므로, 위의 가정에 대한 오차는 어느 정도 존재한다. 이 오차의 크기는 계단입력에 대한 상태변수 응답이 빠른 시간 내에 평형상태에 도달한 후 지속되는 형태일수록 작아진다. 상태변수 y 의 운동방정식을 시간에 따른 미분항들에 대해 쓰면 다음과 같이 표현할 수 있다.

$$f(x, u) = a_n y^{(n)} + \dots + a_1 y' + ay + bu + v(y, u) = 0 \tag{2}$$

여기서 $v(y, u)$ 는 실제 시스템과의 오차를 보정해주는 부분으로, 이 값은 신경회로망을 통해서 계산한다.[2,5] 본 연구에서는 1개의 hidden layer 와 4개의 뉴런을 갖는 신경회로망을 이용하였으며, 현재와 지난 시간의 상태변수 값과 현재의 제어입력 값을 입력으로 하고, 선형화된 시스템과 실제 시스템의 출력값 차이를 신경회로망의 출력으로 하였다. Training 방법은 back propagation을 이용하였다. 식 (2)에 dynamic trim 가정을 적용하여 식을 다시 쓰면 다음과 같은 형태가 된다.

$$ay_{DT} + bu_{DT} + v(y_{DT}, u_{DT}) = 0 \tag{3}$$

여기서 아래첨자 DT 는 dynamic trim 가정을 적용한 변수임을 의미하고, $y_{lim}^{upper}, y_{lim}^{lower}$ 은 각각 비행영역의 상, 하한값이다. 식 (3)을 이용해서 상한 및 하한에 대한 한계입력을 구하면 다음과 같다.

$$u_{lim}^{upper} = -\frac{ay_{lim}^{upper} + v(y_{lim}^{upper}, u_{lim}^{upper})}{b} \tag{4}$$

$$u_{lim}^{lower} = -\frac{ay_{lim}^{lower} + v(y_{lim}^{lower}, u_{lim}^{lower})}{b}$$

위 식의 형태를 살펴보면 양변에 구해야 하는 값이 들어가 있기 때문에 반복계산(iteration)으로 입력의 한계값을 구하게 된다.

Dynamic trim 알고리즘은 간단한 형태의 식으로 표현되어 있기 때문에 계산시간이 상대적으로 짧다는 장점을 가지고 있다. 그러나 실제로는 상태변수들의 시간에 대한 미분값이 0이 아니기 때문에 한계영역을 유지함에 있어 어느 정도의 오차가 생기게 된다. 이러한 오차를 보정하기 위해서 본 논문에서는 dynamic trim에 대한 가정을 어느 정도 완화시켜 모든 고차 미분항을 0으로 하는 대신, 1차 미분항은 0이 아닌 값을 가진다고 가정한 후에 이를 차분식의 형태로 표현하도록 수정 보완하였다. 이러한 가정을 적용해서 식 (3)을 다시 쓰면 다음과 같이 된다.

$$\frac{y^+ - y}{\Delta t} = ay + bu + v(y, u) \tag{5}$$

여기서 y^+ 는 Δt 이후의 상태변수 값이다. 이 식을 통해서 한계입력을 계산하면 다음과 같은 형태가 된다.

$$\frac{y_{lim}^{upper} - y}{\Delta t} - ay_{lim}^{upper} - bu_{lim}^{upper} = v(y_{hist}, u_{lim}^{upper}) \tag{6}$$

$$\frac{y_{lim}^{lower} - y}{\Delta t} - ay_{lim}^{lower} - bu_{lim}^{lower} = v(y_{hist}, u_{lim}^{lower})$$

여기서 y_{hist} 는 상태변수의 시간에 대한 값으로, 본 연구에서는 3개의 시간($t, t - \Delta t, t - 2\Delta t$)에 대한 값을 사용하였다. 식 (6) 역시 양변에 한계입력 값이 포함되어 있으므로, 반복계산을 통해서 한계입력을 구한다. 이렇게 구한 한계입력은 기존의 방식에 의한 식 (4)를 이용해서 구한 입력보다 정확한 값을 가지게 된다.

2.3 Peak Response Estimation

Peak response algorithm은 제한하고자 하는 상태변수의 Peak Response를 예측하는 기법이다. 시스템에 어떠한 외부 입력이 가해지면 그 시스템의 반응을 시간에 따라서 나타낼 수 있다. Peak response algorithm은 반응이 가해진 직후에서의 시스템의 상태를 분석하는데 유용한 기법이다. 이러한 Peak response algorithm의 특성 때문에 시스템이 선형 모델에 가까운 특성을 가

진 경우에 큰 효과를 발휘한다[1,6].

시스템을 단일 입력, 단일 출력 모델로 가정하고, 선형화된 식으로 나타내면 다음과 같이 표현된다.

$$\Delta \dot{x} = A\Delta x + B\Delta u \quad (7)$$

$$\Delta y = C\Delta x \quad (8)$$

여기서 $\Delta x, \Delta y, \Delta u$ 는 각각 다음과 같이 상태변수, 출력값, 입력값의 평형상태 e 에 대한 변화량을 의미한다.

$$\begin{aligned} \Delta x &= x - x_e \\ \Delta y &= y - y_e \\ \Delta u &= u - u_e \end{aligned} \quad (9)$$

임의의 초기값에 대해서 위 시스템의 계단응답을 식으로 나타내면 다음과 같다.

$$y(t) = \Delta y(t) + y_e = y_e + Ce^{At}\Delta x_0 + CA^{-1}(e^{At} - I)B\Delta u \quad (10)$$

출력상태의 평형값을 이용해 출력벡터의 초기조건을 나타내면 다음과 같이 표현된다.

$$y(t) = y_0 + E_1(t)\Delta x_0 + E_2(t)\Delta u \quad (11)$$

여기서 $\Delta x_0 = \Delta x(0)$, $y_0 = y(0) = y_e + C\Delta x_0$,

$E_1(t) = C(e^{At} - I)$, $E_2(t) = CA^{-1}(e^{At} - I)B$ 이다.

Peak response algorithm의 장점은 계산양이 적어 실시간으로 적용할 수 있다는 것인데, 그 이유 중 하나는 E_1, E_2 가 시간에 따라서 변화하는 값이 아니기 때문에 이 값을 미리 구해 놓은 후에 사용할 수 있기 때문이다.

제한하고자 하는 임계변수를 $y(t)$ 로 선택한 후, 이를 제공하여 정리하면 다음과 같은 식을 얻을 수 있다.

$$\|y_{\max}\|^2 = \Delta u^T F \Delta u + G \Delta u + H \quad (12)$$

여기서

$$F = E_2^T E_2 \quad (13)$$

$$G = 2(\Delta x_0^T E_1^T + y_0^T) E_2 \quad (14)$$

$$H = y_0^T y_0 + 2y_0^T E_1 \Delta x_0 + \Delta x_0^T + x_0^T E_1^T \Delta x_0 \quad (15)$$

위 식에 현재의 상태변수와 출력의 초기값인 x_0, y_0 를 대입하면 $y(t)$ 의 값을 구할 수 있는데, 이는 곧 미래의 짧은 시간 동안의 예측값을 의미한다. 여기서 gradient vector n 을 다음과 같이 정의한다.

$$n = 2F\Delta u_0 + G^T \quad (16)$$

식 (12)는 n 을 이용하여 다음과 같이 정리할 수 있다.

$$\begin{aligned} [(n^T F n) / \|n\|^2] d^2 + (G n / \|n\|) d \\ + H - \|y_{\max}\|^2 = 0 \end{aligned} \quad (17)$$

여기서 d 는 일차원 변수로 제어입력과 입력의 한계치 사이의 거리를 의미한다.

또한 d 를 이용해 제어입력과 그 한계치까지의 임계제어 여유벡터 Δu^* 를 나타내면 아래와 같다.

$$\Delta u^* = d^*(n / \|n\|) \quad (18)$$

여기서

$$d^* = \min d(t) \quad (19)$$

따라서, d^* 값을 추정함으로써 제어입력의 한계값을 예측할 수 있다.[7] 제어입력에 대한 제한이 가해지기 시작하는 지점과 제어입력에 대한 제한이 최대한으로 가해져야 하는 지점을 각각 d_1, d_2 라 하면, 그림 2에 정리된 방식의 알고리즘을 이용해서 제어입력을 단계적으로 조절할 수 있다.

$\begin{aligned} &\text{if } d^* > d_1 \\ &\text{then } U = U_{\max} \\ \\ &\text{else if } d^* > d_2 \\ &\text{then } U = U_{\max} (1 - [(d^* - d_1) / (d_2 - d_1)])(n / \ n\) \\ \\ &\text{else } U = 0 \end{aligned}$

그림 2. Peak Response Estimation으로 한계입력을 구하는 과정

2.4 제어이득 스케줄링 기법

제어이득 스케줄링(Gain Scheduling) 기법은 기존의 항공기에서 많이 이용되어 온 방식으로 비행영역 보호 시스템 설계에 적용하면, 한계예측 부분이 필요없다는 부분에서 장점을 가지고 있다. 반면, 제어이득을 튜닝(tuning)하는 과정이 필요하다는 단점이 있다.

FBW 시스템에서 조종사가 스틱을 움직여서 입력을 생성하면 비행제어 컴퓨터를 통해서 조종면을 움직이게 된다. 따라서 본 연구에서는 비행제어 컴퓨터의 제어이득을 비행상황에 따라 변경시킴으로써 항공기가 비행영역을 벗어나지 않게 하여 비행영역 보호 시스템을 설계하였다.

제어이득을 설정하는 방식에는 여러 가지가 있으나, 상태변수가 제한된 비행영역에 근접함에 따라서 제어이득 값을 낮추어서 항공기에 실제로 가해지는 입력이 줄어드는 효과를 내도록 할 수 있다. 본 연구에서는 항공기가 안전한 영역에서 일상적 운항을 하고 있을 때는 일정한 값의 제어이득을 유지하다가, 특정 값 이상으로 한계값에 가까워지게 되면 제어이득 값을 상태변수에 대해 선형적으로 감소시키는 방식을 적용하였다.

III. 수치 시뮬레이션

본 연구에서는 Eclipse 500 항공기의 비선형 모델을 이용하여 설계한 시스템의 성능을 비교하였다. 그림 3은 본 연구에서 사용된 비선형 모델의 시뮬링크 블록선도를 보이고 있다.

앞에서 언급했던 dynamic trim 알고리즘, peak response estimation, 제어이득 스케줄링 기법을 각각 적용하여 추력을 이용해 수평속도를 제한하는 비행영역 보호 시스템을 설계하였다. 실제 항공기의 한계영역은 트림점에서 상당히 멀리 위치하여 있기 때문에 수치 시뮬레이션에서 속도값이 한계영역에 도달하기 위해서는 상당히 긴 시간이 필요하다. 그래서 본 논문에서는 그림 4와 같이 실제 항공기의 한계영역과 비슷한 양상을 가지되 크기가 작은 값을 이용하였다. 추력은 크기 0.5인 계단입력을 사용하였다. 그리고 추력을 제외한 입력값(엘리베이터, 러더, 플랩)들은 모두 0으로 설정하였다. 마지막으로, 결과를 나타내는 그래프에서 제어입력들은 구동기 모델을 거친 후의 값을 표시하였다. 즉, 어떠한 입력이 의

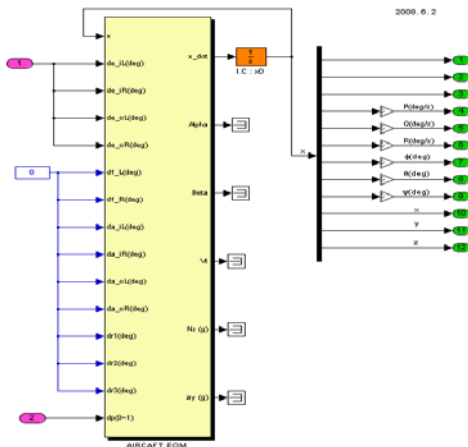


그림 3. Eclipse 500 항공기의 비선형 모델

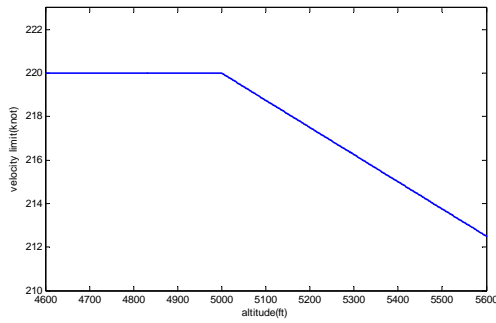


그림 4. 고도에 따른 한계속도 값

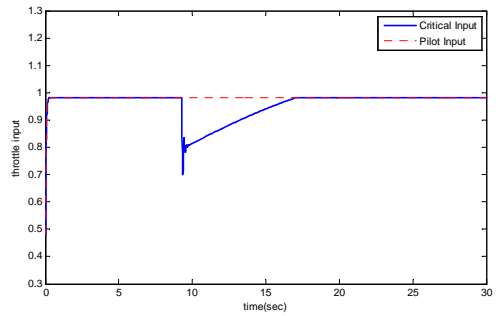


그림 5. Dynamic Trim 알고리즘 적용 시 제어입력 값

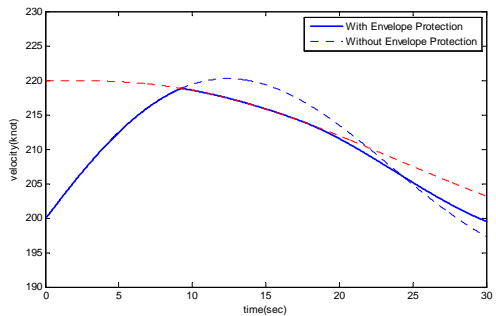


그림 6. Dynamic Trim 알고리즘 적용 시 수평속도 값

부에서 가해지면 트림값에 해당하는 입력이 추가된 후에 엔진의 동특성을 반영하여 실제 항공기에 적용되는 입력값을 표시하게 되는 것이다.

Dynamic trim 알고리즘을 적용한 결과를 그림 5~6에 도시하였다. 그림 5는 제어입력을 나타내는데, 처음에는 거의 1에 가까운 입력을 유지하다가 속도가 한계값에 다다르면 계산된 한계입력값을 주게 됨을 알 수 있다. 그림 6은 시간에 따른 수평속도 값을 나타낸다. 고도에 따라서 변화

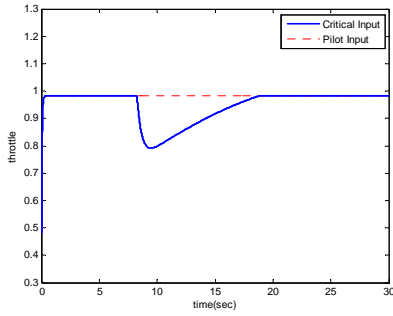


그림 7. Peak Response Estimation 적용 시 제어입력 값

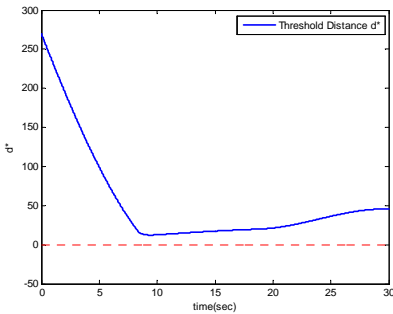


그림 8. 시간에 따른 d^* 의 값

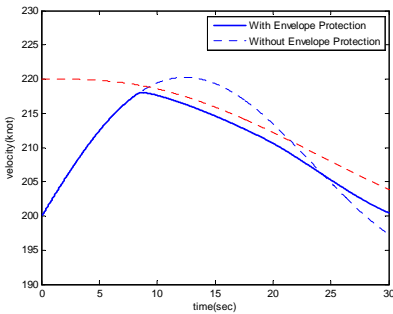


그림 9. Peak Response Estimation 적용 시 수평속도 값

하는 한계값을 넘지 않으면서 운항하고 있음을 볼 수 있다. 또한 한계영역에 아주 가깝게 접근하지 않는 한, 외부의 간섭을 최소화하며 제어가 이루어지고 있음을 확인할 수 있다.

그림 7~9는 Peak response algorithm을 이용하여 수평속도 u 에 대한 비행영역 제한을 가한 경우의 결과를 나타내는 그래프이다. 그림 7은 시간에 따른 제어입력의 크기를 나타낸다. 초기에는 1에 가까운 입력을 유지하다가 한계영역에

근접할 때 입력의 크기를 줄이고 있음을 알 수 있다. 그림 8은 시간이 변함에 따라서 제어입력과 제어입력의 한계치 사이의 여유를 나타내는 d^* 의 변화를 나타내는 그래프이다. 시뮬레이션이 시작된 시점에는 수평속도 u 와 최대 수평속도 사이의 차이가 크기 때문에 d^* 또한 큰 값을 가진다. 그러나 시스템에 일정한 추력을 가하게 되면 시스템의 수평속도가 점차 증가하게 되고 결국 최대수평속도에 도달하게 되는데, 이 시점에 이르면 제어입력의 여유가 거의 없어지기 때문에 d^* 의 값이 거의 0에 도달하게 된다. 그림 8은 시간에 따른 수평속도의 변화를 나타낸다. 비행영역 보호 알고리즘을 적용했을 때 u 값이 비행영역 안에서 잘 머무르고 있는 것을 알 수 있다. Dynamic trim 알고리즘의 경우와는 다르게 한계점과 어느 정도의 간격을 유지하며 운항함을 확인할 수 있다. 이는 peak response estimation의 방법이 선형으로 근사화한 모델을 사용하는데, 이 모델과 실제 비선형 시스템과의 차이에서 기인한 것으로 보인다.

제어이득 스케줄링 기법을 적용한 결과를 그림 10~12에 도시하였다. 그림 10은 속도의 한계값이 220 knot일 때, 수평속도 크기에 따른 제어이득 값의 변화를 나타낸다. 실제 시뮬레이션에서는 고도에 따라서 한계속도가 변화하므로 제어이득 값의 변화폭 또한 변하게 된다. 그림 11은 실제 제어 입력값의 변화를 나타내고 있다. 그림에서 보듯이 처음에는 거의 1에 가까운 입력을 유지하다가 속도가 한계값에 근접함에 따라서 입력값을 줄이고 난 후에, 항공기가 안정적인 영역에 도달하면 다시 입력값을 올리는 형태를 가짐을 알 수 있다. 그림 12는 제어이득 스케줄링 기법을 적용 시 수평속도 값의 변화를 나타낸다.

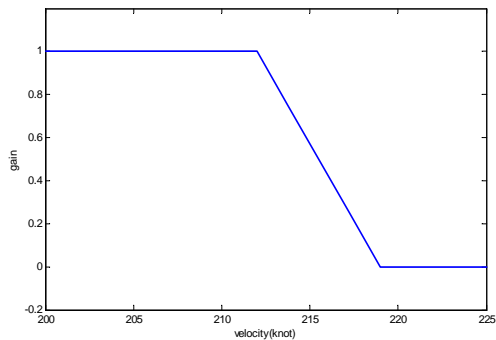


그림 10. 수평속도 값에 따른 제어이득 값

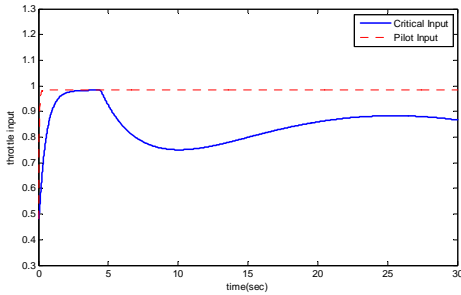


그림 11. 제어이득 스케줄링 기법 적용 시 제어입력 값

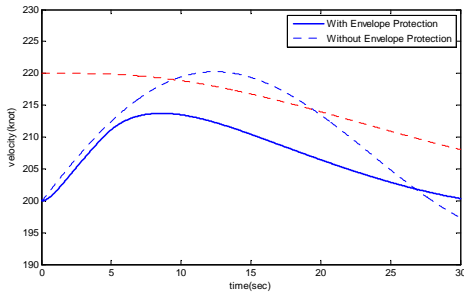


그림 12. 제어이득 스케줄링 기법 적용 시 수평속도 값

앞의 결과들과 마찬가지로 한계값을 넘지 않으면서 운항하고 있음을 볼 수 있지만 어느 정도 여유분을 가지고 있다. 이 여유분은 튜닝을 통해서 줄일 수 있지만 그에 따른 안정성 감소를 가져올 수 있다.

비행영역 보호 알고리즘의 성능은 크게 두 가지로 나눌 수 있는데, 첫 번째는 한계영역을 초과하지 않는 범위 내에서 최대한 조종사가 의도한 기동을 하는 것이며, 두 번째는 조종사의 입력에서 변화량을 최소화하는 것이다.

한계영역을 초과하지 않는 범위 내에서 조종사가 의도한 기동에 가까운 정도를 표시하는 척도를 다음과 같이 생각할 수 있다.

$$J_1 = \min \{y_{crit}(t) - y_{ep}(t)\} \quad (20)$$

여기서 $y_{ep}(t)$ 는 각각 비행영역 보호 알고리즘 적용시의 상태변수 값이며, $y_{crit}(t)$ 는 한계영역 값이다. 이 값이 작을수록 비행영역 한계에 근접할 때까지는 외부의 간섭 없이 조종사의 의도대로 기동이 이루어졌음을 의미한다. 반대로 값이 클수록 한계에 도달하기 전에 비행영역 보호 알

고리즘이 작동하여 보수적인 영역 보호를 수행하였음을 의미한다.

한편, 조종사의 입력에 대한 변화량을 평가할 수 있는 척도는 다음과 같이 설정한다.

$$J_2 = \int |u_{\pi lot} - u_{ep}| dt \quad (21)$$

여기서 $u_{\pi lot}, u_{ep}$ 는 각각 조종사의 입력과 비행영역 보호 알고리즘 적용 시의 제어입력을 의미한다. 위 값이 클수록 비행영역 보호를 위하여 조종사의 입력과 큰 차이가 나는 한계입력을 가하였음을 의미한다.

또한, 조종입력의 최대 변화율을 다음과 같이 정의할 수 있다.

$$J_3 = \max \{ \dot{u}_{ep} \} \quad (22)$$

위 값이 클수록 제어입력 변화율이 급격한 구간이 존재하고 있음을 의미한다. 실제 항공기 구동 시에는 엔진의 동특성에 의해서 최대 제어입력 변화율이 제한된다. 그러나 그 한계 내에서도 급격한 변화를 가지는 구간이 적을수록 안정적인 비행이 가능하기 때문에 제어입력의 최대 변화율은 비행 안정성에 대한 성능계수로 사용될 수 있다.

Dynamic trim 알고리즘, peak response estimation, 그리고 제어이득 스케줄링 알고리즘에 대해서 각각 J_1, J_2 값을 구해 보면 표 1과 같다.

표 1을 살펴보면, 전체적으로 dynamic trim 알고리즘이 가장 좋은 성능을 보임을 확인할 수 있다. 그러나 J_3 에서 보듯이 dynamic trim 알고리즘의 경우 항공기가 한계영역에 다다랐을 때 너무 급격한 제어입력의 변화를 일으킨다는 단점이 있다. 반면, peak response estimation은 한계영역과 근접도에 있어서는 좋지 않은 성능을 가지지만, 제어입력의 변화가 부드럽게 이루어진다는 장점이 있다.

표 1. 비행영역 보호기법 성능 비교

	한계영역 근접도 (J_1)	제어입력 변화량 (J_2)	최대 제어입력 변화율 (J_3)
Dynamic trim 알고리즘	0.00005	0.6902	9.06
Peak response estimation	0.9195	1.0188	5.4055
제어이득 스케줄링	5.3547	4.0698	0.77

제어이득 스케줄링의 경우는 한계영역 근접도와 제어입력 변화율 모두 성능이 낮지만, 설계 과정에 있어서는 가장 간단하다는 장점을 가진다.

IV. 결 론

본 논문에서는 dynamic trim 알고리즘, peak response estimation, 제어이득 스케줄링 기법을 이용해서 항공기의 수평속도에 대한 비행영역 보호 시스템을 설계하였다. 설계된 시스템의 성능을 비교 검증하기 위해서 Eclipse 500 항공기의 비선형 시스템에 대해서 수치 시뮬레이션을 수행하였다. 그 결과, 한계영역을 벗어나지 않으면서 최대한의 기동을 하고자 하는 경우에는 dynamic trim 알고리즘이, 급격한 기동을 최대한 피하며 비행영역 보호를 수행하고자 할 때는 peak response estimation이 적합함을 알 수 있었다. 제어이득 스케줄링의 경우에는 시스템이 복잡하여 dynamic trim 알고리즘이나 peak response estimation을 사용하기 힘든 경우에 활용할 수 있을 것으로 보인다.

후 기

본 연구는 항공우주연구소 지원으로 수행하는 다중화 FBW 비행제어시스템 핵심기술 연구사업의 일환으로 수행되었음.

참고문헌

1) N. Sahani, "Envelope Protection System for Piloted and Unmanned Rotorcraft", Ph.D. Dissertation, Department of Aerospace

Engineering, Pennsylvania State University, PA, 2005.

2) S. Unnikrishnan, J.V.R. Prasad, and A. J. Calise, "Adaptive Dynamic Trim Prediction for Limit Detection and Avoidance", AIAA Atmospheric Flight Mechanics Conference, Austin, TX, Aug. 2003.

3) A. Bateman, D. Ward, R. Barron, and M. Whalley, "A Piloted Simulation Evaluation of a Neural Network Limit Avoidance System for Rotorcraft", AIAA Atmospheric Flight Mechanics Conference, Portland, OR, Aug. 1999.

4) J. F. Horn, A. J. Calise, and J. V. R. Prasad, "Flight Envelope Cueing on a Tilt-Rotor Aircraft Using Neural Network Limit Prediction", Journal of America Helicopter Society, Vol. 46, No. 1, 2001, pp. 23-31.

5) S. Unnikrishnan, and J.V.R. Prasad, "Carefree Handling using Reactionary Envelope Protection Method", AIAA Guidance, Navigation, and Control Conference and Exhibit, Colorado, Aug. 2006.

6) V. Sharma, P. G. Voulgaris, and F. Emilio Frazzoli, "Aircraft Autopilot Analysis and Envelope Protection for Operation Under Icing Conditions", Journal of Guidance, Control, and Dynamics, AIAA, Vol. 27, No. 3, 2004, pp. 454-465.

7) J. F. Horn, and N. Sahani, "Detaction and Avoidance of Main Rotor Hub Moment Limits on Rotorcraft", Journal of Aircraft, Vol. 41, No. 2, 2004, pp. 372-379.