

論文

소형항공기 자동착륙시스템의 퍼지제어기 적용에 관한 기초 연구

김근택*, 김응태*, 성기정*, 안석민*

A Preliminary Study on the Application of a Fuzzy Controller for the Automatic Landing System of Small Aircraft

Keun-Taek Kim*, Eung-Tai Kim*, Kiejeong Seong*, Seok-min Ahn*

ABSTRACT

Fuzzy control has emerged as a practical alternative to classical control schemes in controlling certain time-varying, nonlinear, and ill-defined processes. As the current of this kind of a research paradigm, we concluded that there is a need for application study of a fuzzy control theory to the flight control systems of small aircraft being to be developed at KARI. And then, this preliminary study was carried out to the automatic landing system of the canard aircraft (Firefly) for the purpose of the preparation of extension of research contents and various application areas, in which FMRLC was chosen as the fuzzy controller of the system.

Key Words : Automatic Landing System(자동착륙시스템), Fuzzy Control(퍼지제어), FMRLC(Fuzzy Model Reference Learning Controller), Small Aircraft(소형항공기), Flight Control System(비행조종장치)

1. 서 론

1.1. 배경 및 연구목적

제어분야에서 퍼지제어는 고전적인 제어이론에 대한 새로운 패러다임으로서, 학계의 많은 관심과 더불어 실질적인 산업분야에 적용이 다수 이루어지고 있다. 특히, 시변의 매개변수를 가지거나 비선형성이 존재하는 시스템과 혹은 간단히 정의할 수 없는 플랜트를 가지는 시스템의 경우에는 퍼지제어의 적용을 통해 상당한 제어성능의 향상을 얻을 수가 있음이 보고되고 있다. 특히 최근에는 항공분야에서의 한 적용 예로서, 항공기의 자동착륙시스템과 관련한 퍼지제어에 관한 연구가 발표되었다[1,2,3].

본 연구에서는 이러한 연구의 흐름에 따라, 본 항공우주연구원에서 개발하였던 소형항공기의 비행조종장치 개발 분야에서 퍼지제어 이론의 적용을 위한 기초적인 연구를 수행할 필요가 있을 것으로 판단하고 우선, 자동착륙시스템에 대하여 퍼지제어의 적용을 위한 기초적인 연구를 수행하였다.

1.2. 퍼지제어기의 선정

고전적인 제어이론에 대응하여 제안된 많은 퍼지제어기(Fuzzy Controller)들은 성공적인 적용 사례에도 불구하고, 다수의 학자들로부터 수많은 의문과 회의를 받고 있는 것이 사실이다. 또한, 제어기 각각의 특수한 적용 한계와 제어기의 다양한 매개변수를 어떻게 정의하여야 하는지 명확하지 않을 뿐만 아니라, 제어시스템의 매개변수가 변하였거나 구조적 변화가 있을 경우 예측할 수 없는 결과를 초래할 수 있는 문제점이 있다. 이 문제점을 보완하기 위해, Layne과 Passino[1]

2007년 월 일 접수 ~ 2007년 월 일 심사완료

* 한국항공우주연구원 항공혁신기술연구소

연락처자, E-mail : ktkim@kari.re.kr

대전시 유성구 과학로 169-84

는 고전적인 적응제어의 일반적인 개념을 적용한 Procyk과 Mamdani(1979)의 언어적인 자기구성 제어기(Linguistic Self-Organizing Controller, SOC)를 근간으로 Fuzzy Model Reference Learning Controller (FMRLC)를 제안하였다.

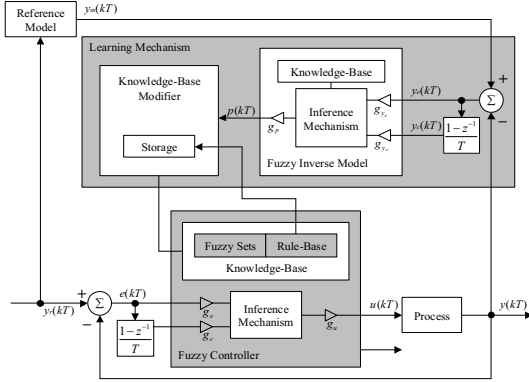


Fig. 1 FMRLC의 개략적인 구조

FMRLC의 프로세스는 Fig. 1에서와 같이, 퍼지 제어기(Fuzzy Controller)와 기준모델(Reference Model), 학습장치(Learning Mechanism) 등으로 이루어진다. 이것은 Model Reference Adaptive Controller와 마찬가지로, 기준모델을 통해 제어기의 성능 목표를 설정하고 원하는 성능을 얻을 수 있도록 학습장치를 통해 퍼지제어기의 퍼지관계들을 자동으로 조정하는 제어기이다.

FMRLC의 학습제어기법은 퍼지제어시스템으로부터 데이터를 검출하는 학습장치를 사용하여, 현재 성능의 특성을 파악하고 사전에 규정하였던 성능목표를 만족시킬 수 있도록 자동적으로 퍼지제어기를 분석, 조정하는 것이다. 여기서 이들 성능목표들은 Fig. 1에서와 같이 기준모델을 통해 규정되는데, 이것은 종래의 MRAC와 유사한 방법으로 학습장치가 폐회로 시스템 $[y_r(kT) \Rightarrow y(kT)]$ 을 마치 기준모델 $[y_r(kT) \Rightarrow y_m(kT)]$ 처럼 작동하도록 퍼지제어기를 조정한다.

2. FMRLC의 개요 및 주요 이론

2.1. 퍼지제어기

Fig. 1에서 퍼지제어기의 프로세스는 r 개의 입력과 s 개의 출력을 가진다고 가정하고, 벡터 형태로 표시하면 다음과 같이 쓸 수 있다.

$$\text{입력: } \underline{u}(kT) = [u_1(kT), \dots, u_r(kT)]^T \quad (1)$$

$$\text{출력: } \underline{y}(kT) = [y_1(kT), \dots, y_s(kT)]^T \quad (2)$$

단, T : 표본주기(Sampling Period)

퍼지제어기는 플랜트의 출력, $\underline{y}(kT)$ 와 기준 입력(Reference Input), $\underline{y}_r(kT)$ 의 선형함수를 통해 생성한다. Fig. 1의 경우에는 그러한 선형사상(Linear Map)의 특수한 경우이다.

퍼지제어기의 입력은 오차(Error)와 오차변화율(Change in Error)로서 다음과 같이 정의한다.

$$\underline{e}(kT) = [e_1(kT), \dots, e_s(kT)]^T \quad (3)$$

$$= \underline{y}_r(kT) - \underline{y}(kT)$$

$$\underline{c}(kT) = [c_1(kT), \dots, c_s(kT)]^T \quad (4)$$

$$= \frac{\underline{e}(kT) - \underline{e}(kT - T)}{T}$$

$$\text{단, } \underline{y}_r(kT) = [y_{r_1}(kT), \dots, y_{r_s}(kT)]^T$$

: 원하는 프로세스 출력

일반적으로, 퍼지제어기의 실행에서 보다 큰 유연성을 위해, 각 프로세스의 입력에 대한 정의역은 일정한 스케일링계수(Scaling Factor, 일종의 이득(Gain))에 의해 구간 $[-1, +1]$ 로 정규화한다. 따라서 퍼지제어기의 설계에서, 이득, \underline{g}_e , \underline{g}_c , \underline{g}_u 는 각각 오차 $\underline{e}(kT)$ 와 오차변화율, $\underline{c}(kT)$ 및 제어기 출력, $\underline{u}(kT)$ 에 대한 정의역을 정규화하는데 사용한다.

또한, n -번째 프로세스 입력에 대한 퍼지제어기의 지식기반은 다음과 같은 IF-THEN 제어규칙으로부터 생성된다.

"If \tilde{e}_1 is \tilde{E}_1^k and ... and \tilde{e}_s is \tilde{E}_s^k and \tilde{c}_1 is \tilde{C}_1^l and ... and \tilde{c}_s is \tilde{C}_s^m , Then \tilde{u}_n is $\tilde{U}_n^{j, \dots, k, l, \dots, m}$ "

단, \tilde{e}_i , \tilde{c}_i : 제어기 입력의 언어적인 변수

\tilde{u}_n : 제어기 출력의 언어적인 변수

\tilde{E}_i^q , \tilde{C}_i^q : \tilde{e}_i 와 \tilde{c}_i 에 대한 q -번째 언어적인 변수

$\tilde{U}_n^{j, \dots, k, l, \dots, m}$: 일련의 \tilde{u}_n 에 대한 언어적인 변수

이러한 제어규칙은 다음과 같은 형태의 퍼지 함의(Fuzzy Implication)를 얻기 위하여 퍼지집합(Fuzzy Set) 이론을 사용하여 양자화 할 수 있다.

"If E_1^j and ... and E_s^k and C_1^l and ... and C_s^m , Then $U_n^{j, \dots, k, l, \dots, m}$ "

단, E_i^q , C_i^q , $U_n^{j, \dots, k, l, \dots, m}$: \tilde{e}_i , \tilde{c}_i , \tilde{u}_n 을 각각 양자화 한 퍼지집합

따라서 퍼지함의는 다음과 같이 하나의 퍼지 관계로 나타낼 수 있다.

$$R_n^{j,\dots,k,l,\dots,m} = (E_1^j \times \dots \times E_s^k) \times (C_1^l \times \dots \times C_s^m) \times U_n^{j,\dots,k,l,\dots,m} \quad (3)$$

그리고 제어규칙에 대한 퍼지제어기의 의사결정장치(Decision Mechanism)는 다음과 같이 표시할 수 있다.

$$\hat{U}_n^{j,\dots,k,l,\dots,m}(kT) = \left[\begin{array}{l} \{\hat{E}_1^j(kT) \times \dots \times \hat{E}_s^k(kT)\} \\ \times \{\hat{C}_1^l(kT) \times \dots \times \hat{C}_s^m(kT)\} \end{array} \right] \circ R_n^{j,\dots,k,l,\dots,m} \quad (4)$$

단, $\hat{E}_j(kT)$, $\hat{C}_j(kT)$: $\mathcal{L}(kT)$ 와 $\mathcal{C}(kT)$ 각각에 대한 퍼지화 오차 및 오차율
 $\hat{U}_n^{j,\dots,k,l,\dots,m}(kT)$: 함의 퍼지집합
 \circ : Zadeh의 Min-Max 연산 부호

$$\mu_{R_1 \circ R_2}(x, z) = \bigvee_u [\mu_{R_1}(x, y) \wedge \mu_{R_2}(y, z)]$$

일반적으로 퍼지시스템의 설계에서 퍼지함의는 퍼지시스템의 입력을 기술하는 가능한 모든 조합의 퍼지집합에 대해 존재한다. 따라서 퍼지 제어기는 많은 퍼지함의로 구성되어 있으며, 퍼지함의의 전반적인 제어기능은 다음과 같은 중력 중심(Center Of Gravity, COG)법에 의해 연산할 수 있다.

$$u_n(kT) = \frac{\sum_{j,\dots,k,l,\dots,m} \hat{A}_n^{j,\dots,k,l,\dots,m}(kT) \hat{c}_n^{j,\dots,k,l,\dots,m}(kT)}{\sum_{j,\dots,k,l,\dots,m} \hat{A}_n^{j,\dots,k,l,\dots,m}(kT)} \quad (5)$$

단, $\hat{A}_n^{j,\dots,k,l,\dots,m}(kT)$, $\hat{c}_n^{j,\dots,k,l,\dots,m}(kT)$: 면적 및 면적의 중심

2.2. 기준모델

기준모델은 프로세스의 원하는 성능을 양자화 하는데 필요한 능력을 제공한다. 일반적으로, 기준모델은 동력학적인 시스템(선형 혹은 비선형, 시불변 혹은 시변, 이산시간 혹은 연속시간, 등등) 가운데 어떤 하나의 형태일 것이다. 전반적인 시스템의 성능은 오차에 대한 신호 즉,

$$y_e(kT) = [y_{e_1}(kT), \dots, y_{e_s}(kT)]^T \quad (6)$$

단, $y_e(kT) = y_m(kT) - y(kT)$

를 생성함으로써 기준모델과 연관하여 연산한다.

주어진 기준모델은 안정성(Stability), 지나침(Overshoot), 안정시간(Settling Time) 등과 같은 설계기준(Design Criteria)의 특성을 결정한다. 또한, 기준모델의 입력은 기준입력, $y_r(kT)$ 로서,

만일 학습장치에 의해 오차인 $y_e(kT)$ 가 모든 시간에 대해 매우 작은 값으로 유지된다면, 제어할 프로세스에서 원하는 성능을 만족하는 것이다. 따라서 $y_e(kT)$ 는 시간, $t = kT$ 에서 원하는 성능을 만족하는 범위까지 그 특성을 제공한다. 만일 성능을 만족하면 즉, $y_e(kT) \approx 0$ 인 경우에는, 학습장치가 퍼지제어기에 중대한 변경을 일으키지 않지만, 반대로 $y_e(kT)$ 가 매우 크다면 원하는 성능을 얻지 못한 것이므로, 학습장치는 퍼지제어기를 조정하여야 한다.

2.3. 학습장치

앞서 언급한 바와 같이, 학습장치는 직접적응 퍼지제어기의 지식기반을 수정하는 기능을 수행하여, 폐회로시스템이 마치 기준모델과 같이 움직이도록 한다. 이러한 지식기반의 수정은 제어 프로세스와 기준모델 그리고, 퍼지제어기로부터 얻어진 데이터를 기초로 하여 이루어진다. 학습장치는 크게 퍼지역모델(Inverse Model)과 지식기반 수정장치(Knowledge-Base Modifier)의 두 부분으로 나누어진다. 퍼지역모델은 프로세스 출력 변화를 얻기 위해 필요한 프로세스 입력, $p = [p_1 \dots p_r]^T$ 의 상대적인 변화에 대응하여 프로세스 출력, $y_e(kT)$ 에서 필요한 변화를 사상(Mapping)하는 기능을 수행한다. 지식기반 수정장치는 프로세스 입력에서 요구되는 변화를 일으키기 위해서 퍼지제어기의 지식기반을 수정하는 기능을 수행한다.

2.4. FMRLC 설계 절차

FMRLC를 설계하는 절차에는 다음과 같은 내용을 포함한다.

- 조정될 수 있는 일련의 소속함수를 가지는 직접 퍼지제어기의 규격 \rightarrow 이 퍼지제어기는 정규의 플랜트에 대한 종래의 휴리스틱(Heuristic) 퍼지제어기 설계기법을 통해 선택할 수 있다.
- 원하는 시스템의 성능을 특성을 규정하는 기준모델을 결정
- 원하는 성능이 얻어 질 수 있도록 그 플랜트에 대한 입력을 어떻게 변경하여야 하는지 규정하는 퍼지역모델을 결정
- 퍼지제어기와 퍼지역모델의 정규화 이득에 대한 선택

따라서 퍼지제어기와 퍼지역모델의 정규화 이

득에 대한 선택이 시스템 전체에 영향을 미치기 때문에 이들 매개변수들을 선택하는 방법에 대해 주의를 기울여야 한다. 이러한 선택이 비록 집중적인 주목을 받지는 못한다고 할지라도, 대부분의 학습 또는 적응제어 기법은 제어기의 구조 및 매개변수의 초기 선택을 위한 어떤 전형적인 수단이 필요하다. 이러한 FMRLC의 초기 매개변수를 선택하기 위한 체계적인 방법은 다음과 같다.

주어진 시스템의 물리적인 제한조건으로 인해, 프로세스 입력 및 출력 값에 대한 범위는 일반적으로 그 프로세스의 정량적인 분석을 통해 알 수 있다. 결과적으로, $e(kT)$, $u(kT)$, $y_e(kT)$, $p(kT)$ 등에 대한 값의 범위 혹은 정의역을 결정할 수 있다. 계속해서 g_e , g_u , g_{y_e} , g_p 등의 스케일링계수도 정의역이 구간 $[-1, +1]$ 로 적절히 사상되도록 선택된다. 또한, 스케일링계수 g_c 를 결정하기 위해서는 적응장치(Adaptive Mechanism)를 시스템에서 분리한 다음, 표준의 퍼지제어 설계기법을 사용하거나 또는, $e(kT)$ 를 관찰하면서 반복적으로 입력을 y_e 에 대입하여 구간 $[-1, +1]$ 로 정의역을 사상하기 위한 스케일링계수를 구하여 그 값을 결정한다.

원하는 출력의 변경과 관련이 있는 스케일링계수, g_{y_e} 는 FMRLC에 대한 매개변수를 조정하는 벡터로서, 제어기 수정에 감쇄(Damping)를 주는 효과가 있다. 또한, 스케일링계수, g_{y_e} 의 요소가 증가함에 따라 그러한 감쇄효과는 더욱 커진다. 기준모델의 응답에 대한 전체 프로세스의 응답을 감시함으로써 스케일링계수, g_{y_e} 의 적절한 선택이 이루어 질 수 있다. 주어진 프로세스와 관련된 기준모델의 출력 응답 사이에 바람직하지 않은 진동이 존재하는 경우에는 출력과 관련한 g_{y_e} 가 매우 작아서 발생하는 문제이므로 g_{y_e} 를 증가시켜야 한다. 반면에, g_{y_e} 가 너무 크면, 그 프로세스는 합성되는 감쇄효과 때문에 기준모델을 유지할 수가 없다.

다음은 스케일링계수들을 선택하기 위한 절차를 간략히 기술한다.

- ① 제어기 이득 g_e , g_u , g_{y_e} 의 정의역이 각각 구간 $[-1, +1]$ 로 사상될 수 있도록 선택한다.
- ② 제어기의 이득 g_{p_i} 가 퍼지제어기의 출력 이득, g_{u_i} 와 동일하도록 선택한다. 이것은 $p_i(kT)$ 가 최대 가능한 입력 $u_i(kT)$ 만큼 큰

값을 취할 수 있도록 한다.

③ 표준의 퍼지제어 설계기법을 사용하거나 간단한 실험을 통해 $e(kT)$ 의 정의역을 구간 $[-1, +1]$ 로 사상할 수 있도록 g_c 를 선택한다.

④ 원하는 출력의 변경에 관련된 스케일링계수들을 "0"의 값을 갖도록 배정한다. 즉, g_{y_e} 의 모든 요소들을 "0"의 값으로 배정한다.

⑤ 정상상태의 프로세스에 대해 적절한 크기의 계단함수(Step Function)를 적용하여, 프로세스의 응답과 기준모델의 응답을 조사한다.

⑥ 이상의 절차가 이루어지면 다음과 같은 3가지의 경우가 존재한다.

㉔ 기준모델의 응답에 비해 주어진 프로세스의 출력 응답에서 받아들일 수 없는 진동이 존재하는 경우에는, g_{y_e} 와 관련된 요소를 증가시킨 다음 ⑤단계를 다시 수행한다.

㉕ 만일 주어진 프로세스의 출력 응답이 기준모델의 응답을 유지할 수 없는 경우에는 g_{y_e} 와 관련된 요소를 증가시킨 다음 ⑤단계를 다시 수행한다.

㉖ 만일 프로세스 응답이 기준모델의 응답과 비교하여 만족스러운 결과가 얻어지면 제어기의 설계를 완료한 것이다.

3. 문제의 정의

시뮬레이션 대상 항공기 모델은 본 연구소에서 개발한 선미익기로서, 이에 대한 기본적인 형상 및 제원은 Fig. 2 및 표 1과 같다.

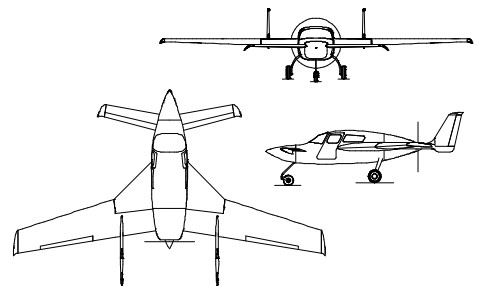


Fig. 2 선미익 항공기(반디호) 형상

표 1. 선미의 항공기 제원

전 장	21.7 ft	이륙거리	1,400 ft
전 폭	34.0 ft	착륙거리	1,500 ft
전 고	8.0 ft	최대수평속도	190 kt
최대탑승인원	4 명	순 항 속도	180 kt
추진기관	250 hp	실 속 속도	60 kt
최대이륙중량	2,600 lb	순 항 고도	8,000 ft
공허중량	1,500 lb	항 속 거리	1,000 nm
최대 상승률	1,200 fpm		

3.1 자동착륙시스템

항공기의 착륙궤도에 대한 자동제어 문제는 자동비행시스템의 구축에 가장 필수적인 업무의 하나로서, 속도와 비행경로의 각도 및 고도 등과 같은 여러 가지 기동조건을 동시에 제어하여야 한다. 일반적으로 이에 대한 문제는 임무에 따라 부과되는 제한조건들을 만족하도록 퍼지제어시스템을 설계하는 것이다. Fig. 3은 일반적인 활공경로(Glide Path)의 궤적을 보여준다.

자동착륙시스템의 성능은 대부분 활공경로의 포착을 위해 얼마나 정확하게 제어할 수 있는가에 달려있다. 그것은 항공기의 안정성뿐만 아니라 Flare 기동의 출발점을 결정하는데 중요하기 때문이다.

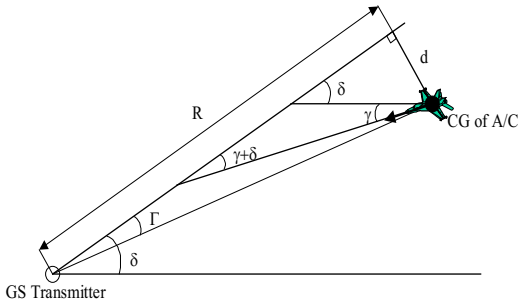


Fig. 3 일반적인 항공기의 활공경로 궤적

본 연구 대상의 선미익 항공기 모델에서는 Pitch Attitude Command Control System과 함께 Speed Control System이 갖추어져 있다고 가정한다. 또한, 항공기 글라이더슬로프(Glideslope) 안테나는 항공기의 무게중심의 위치에 장착되어 있으며 항공기가 글라이더슬로프를 따라 유도된다고 가정한다. 따라서 글라이더슬로프 오차각 (Γ)은 항공기에 장착된 글라이더슬로프 수신기로 감지되어 Pitch Attitude Command Control System에 의해 항공기가 글라이더슬로프를 유지하게 된다.

항공기가 Pitch Attitude Command Control

System의 제어로 글라이더슬로프에 접근하는 속도는

$$\frac{d}{dt}(d) = U \sin(\gamma + \delta) \approx U(\gamma + \delta) \quad (1)$$

와 같이 쓸 수 있다. 항공기 무게중심(C.G.)으로부터 글라이더슬로프까지의 거리는

$$d = \int U(\gamma + \delta) dt = \frac{U}{s}(\gamma + \delta) \quad (\text{in } s\text{-domain}) \quad (2)$$

와 같으므로, 글라이더슬로프 오차각은 다음 식과 같이 쓸 수 있다.

$$\tan \Gamma = \frac{d}{R}, \quad (R = R_0 - U \cdot t) \Rightarrow \therefore$$

$$\Gamma \approx \frac{d}{R} \quad (3)$$

한편, Glideslope Hold System과 Pitch Attitude Command Control System에 대한 블록 선도는 Fig. 4 및 Fig. 5와 같이 구성한다. Fig. 4에서 Glideslope Coupler는 자동비행조종장치에 대해 기준 입력 신호와 글라이더슬로프 수신기 신호의 차이 값인 오차 신호를 결합(Coupling)하는 기능을 갖는다.

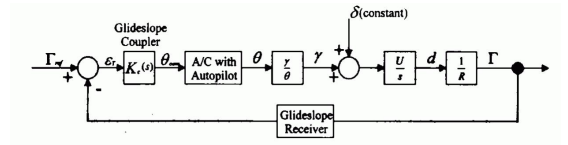


Fig. 4 Glideslope Hold System

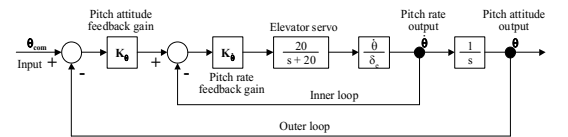


Fig. 5 Pitch Attitude Command System

따라서 항공기 비행경로각, γ 는 다음의 식으로 표시할 수 있다.

$$\gamma = \theta - \alpha \quad \Rightarrow \quad \therefore$$

$$\frac{\gamma}{\theta} = 1 - \frac{\alpha}{\theta} = 1 - \frac{\alpha/\delta_e}{\theta/\delta_e} \quad (4)$$

결과적으로, 주어진 문제의 경우에는 이상과 같은 가정 하에서 원하는 설계 요구조건들을 만족하도록 Glideslope Coupler Gain, $K_c(s)$ 를 설계하는 것이다.

3.2 수학적 모델링 및 설계 요구조건

일반적으로 항공기의 비행역학을 다루는 기본적인 세로운동(Longitudinal Motion)에 대한 운동방정식은 다음과 같이 표시된다.

일반적으로 항공기의 비행역학을 다루는 기본적인 세로운동(Longitudinal Motion)에 대한 운동방정식은 다음과 같이 표시된다.

1) State Vector, \mathbf{x}

$$\mathbf{x} = [u w q \theta]^T$$

2) 엘리베이터의 변위(δ_E)에 의해서만 제어되는 Control Vector, \mathbf{u}

$$\mathbf{u} = \delta_E$$

3) System 행렬 A 와 B

$$A = \begin{bmatrix} X_u & X_w & 0 & -g \cos \gamma_o \\ Z_u & Z_w & U_0 & -g \sin \gamma_o \\ \tilde{M}_u & \tilde{M}_w & \tilde{M}_q & \tilde{M}_\theta \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$B = [X_{\delta_E} \ Z_{\delta_E} \ \tilde{M}_{\delta_E} \ 0]^T$$

따라서, 본 연구에서 가정된 자동비행조종장치가 장착된 항공기 모델의 세로운동방정식은 다음과 같이 유도할 수 있다.

1) State Vector, \mathbf{x}

$$\mathbf{x} = [u w q \theta \delta_e q_c \theta_c \Gamma \Gamma_{ref}]^T \quad (5)$$

2) Control Vector, \mathbf{u}

$$\mathbf{u} = \Gamma \quad (6)$$

3) System 행렬 A 와 B

$$A = \begin{bmatrix} X_u & X_w & 0 & -g \cos \gamma_o & X_{\delta_E} \\ Z_u & Z_w & U_0 & -g \sin \gamma_o & Z_{\delta_E} \\ \tilde{M}_u & \tilde{M}_w & \tilde{M}_q & \tilde{M}_\theta & \tilde{M}_{\delta_E} \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & a_1 K_q & 0 & -a_1 \\ 0 & K_t K_\gamma / R - K_t & -K_t (a_1 K_q U / R) & 0 & 0 \\ 0 & K_\gamma / R & 0 & -K_\gamma U / R & 0 \\ 0 & -1/R & 0 & U/R & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -a_1 & 0 & 0 & 0 & 0 \\ 0 & a_1 K_t & -a_1 K_t K_\gamma a_\gamma K_t K_\gamma & & \\ 0 & 0 & -a_\gamma K_\gamma & a_\gamma K_\gamma & \\ 0 & 0 & 0 & 0 & \\ 0 & 0 & 0 & 0 & \\ 0 & 0 & 0 & 0 & \end{bmatrix} \quad (7)$$

$$B = [00000 K_\gamma K_t K_\gamma 0]^T \quad (8)$$

한편, 상기 운동방정식으로부터 Glideslope

Coupler Gain, $K_c(s)$ 를 설계하기 위하여 적용할 설계 요구조건은 다음과 같이 설정하였다.

- 1) Rising Time < 10sec
- 2) Settling Time < 20sec
- 3) Overshoot < 20%

3.3 시뮬레이션

시뮬레이션 대상 항공기 모델의 경우에는 2개의 입력과 1개의 출력을 가지는 시스템으로서, 오차(e)와 오차의 변화량(c)은 다음과 같이 표시된다.

$$e(kT) = y_r(kT) - y(kT) \quad (9)$$

$$c(kT) = \frac{e(kT) - e(kT-T)}{T} \quad (10)$$

단, $y_r(kT)$: 프로세스의 요구 출력

또한, e 와 c 는 Fig. 6과 같이 각각 11개의 균일분포형 삼각소속함수로 정의한다. 여기서, 각각의 프로세스 입력에 대한 정의역(Universe of Discourse)은 어떤 일정한 스케일링계수 (이득) 값에 의해 구간 $[-1, +1]$ 로 정규화 된다.

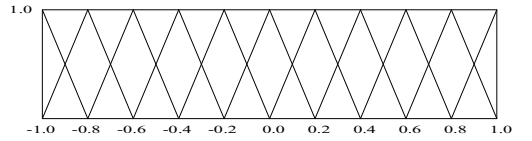


Fig. 6 e 와 c 에 대한 Membership Function

그리고 사용할 퍼지규칙은 구간 $[-1, +1]$ 로 정규화된 소속함수의 중앙값(COG)으로 정의하고, 그 결과를 표 2에 나타내었다.

표 2. Fuzzy Rule Base Array Table

$P_j^{i,k}$	Y_c^k										
	-5	-4	-3	-2	-1	+0	+1	+2	+3	+4	+5
-5	-1.0	-1.0	-1.0	-1.0	-1.0	-1.0	-0.8	-0.6	-0.4	-0.2	0.0
-4	-1.0	-1.0	-1.0	-1.0	-1.0	-0.8	-0.6	-0.4	-0.2	0.0	0.2
-3	-1.0	-1.0	-1.0	-1.0	-0.8	-0.6	-0.4	-0.2	0.0	0.2	0.4
-2	-1.0	-1.0	-0.8	-0.6	-0.4	-0.2	0.0	0.2	0.4	0.6	0.8
-1	-1.0	-0.8	-0.6	-0.4	-0.2	0.0	0.2	0.4	0.6	0.8	1.0
+0	-0.8	-0.6	-0.4	-0.2	0.0	0.2	0.4	0.6	0.8	1.0	1.0
+1	-0.6	-0.4	-0.2	0.0	0.2	0.4	0.6	0.8	1.0	1.0	1.0
+2	-0.4	-0.2	0.0	0.2	0.4	0.6	0.8	1.0	1.0	1.0	1.0
+3	-0.2	0.0	0.2	0.4	0.6	0.8	1.0	1.0	1.0	1.0	1.0
+4	0.0	0.2	0.4	0.6	0.8	1.0	1.0	1.0	1.0	1.0	1.0
+5	0.2	0.4	0.6	0.8	1.0	1.0	1.0	1.0	1.0	1.0	1.0

시뮬레이션은 Glideslope Hold Mode에 대해 다음 2 경우로 나누어 수행하였다.

1) Case 1:

$$R = \text{Constant, Unit Step Response}$$

2) Case 2:

$$R = R_0 - Ut, \text{ Time Varying Response}$$

4. 결과 및 고찰

본 연구에서 시뮬레이션은 MatLab을 사용하여 프로그래밍 하였으며, Simulink 및 기타 다른 프로그램과의 인터페이스는 사용하지 않았다. 왜냐하면, 본 연구에서는 현재 개발 중인 프로젝트에서 퍼지제어의 적용에 관한 타당성을 파악하기 위한 기초연구를 수행하는데 그 목적을 두었기 때문이다. 또한, 고전적인 제어에 의한 시뮬레이션을 수행하고, 그 결과와 비교하여 퍼지제어의 성능을 비교하였다.

본 연구에서의 시뮬레이션 결과는 다음 Fig. 7에서 Fig. 11까지, 그리고 표 3에 간략히 나타내었다. 이 결과는 주어진 설계 요구조건을 충족시킬 뿐만 아니라, 고전적인 제어기에 비해 매우 좋은 성능을 얻을 수 있음을 보여준다.

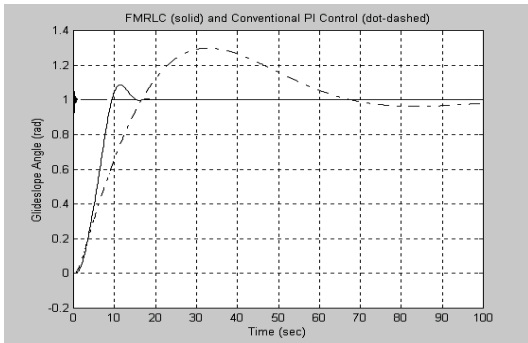


Fig. 7 Case 1의 Unit Step Response

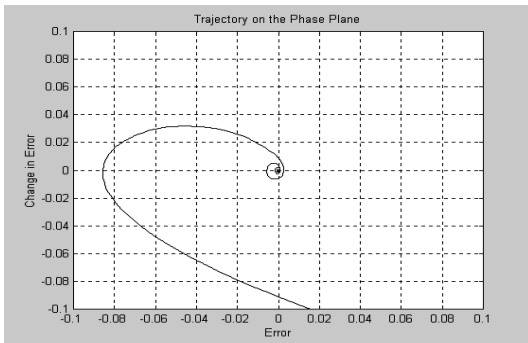


Fig. 8 Case 1의 오차 및 오차의 변화량

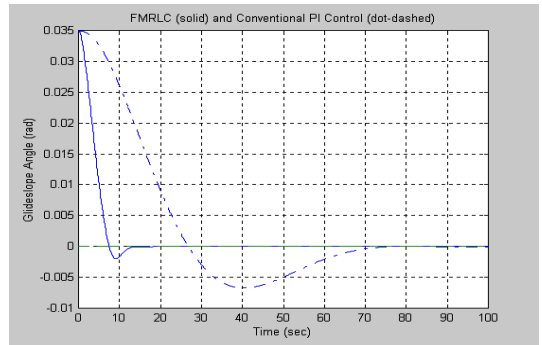


Fig. 9 Case 2의 Time Response

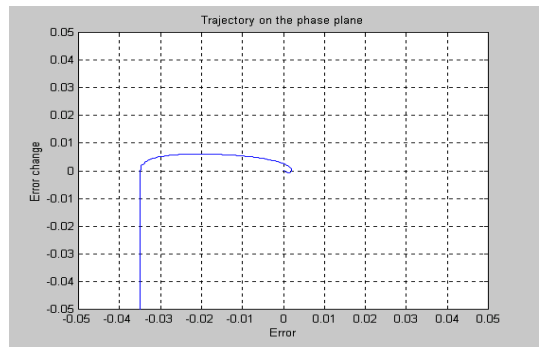


Fig. 10 Case 2의 오차 및 오차의 변화량

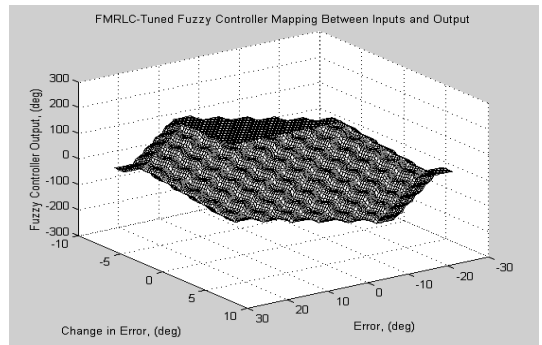


Fig. 11 Case 2의 Control Surface

표 3. Case 2의 조정된 Fuzzy Rule Base

-5.0000	-5.0000	-5.0000	-5.0000	-5.0000	-5.0000	-4.0000	-3.0000	-2.0000	-1.0000	0.0000
-5.0000	-5.0000	-5.0000	-5.0000	-5.0000	-4.0000	-3.0000	-2.0000	-1.0000	0.0000	1.0000
-5.0000	-5.0000	-5.0000	-5.0000	-4.0000	-3.0000	-2.0000	-1.0000	0.0000	1.0000	2.0000
-5.0000	-5.0000	-5.0000	-4.0000	-3.0000	-2.0000	-1.0000	0.0000	1.0000	2.0000	3.0000
-5.0100	-5.0000	-4.0000	-3.0000	-2.0000	-1.0018	-1.0029	-1.0043	1.0000	2.0000	3.0000
-5.0100	-4.0000	-3.0000	-2.0000	-1.0018	0.0024	1.0043	2.0000	3.0000	4.0000	5.0000
-4.0000	-3.0000	-2.0000	-1.0000	0.0000	1.0053	2.0053	3.0000	4.0000	5.0000	5.0000
-3.0000	-2.0000	-1.0000	0.0000	1.0000	2.0000	3.0000	4.0000	5.0000	5.0000	5.0000
-2.0000	-1.0000	0.0000	1.0000	2.0000	3.0000	4.0000	5.0000	5.0000	5.0000	5.0000
-1.0000	0.0000	1.0000	2.0000	3.0000	4.0000	5.0000	5.0000	5.0000	5.0000	5.0000
0.0000	1.0000	2.0000	3.0000	4.0000	5.0000	5.0000	5.0000	5.0000	5.0000	5.0000

또한, 각 경우에 대해 시뮬레이션 결과로서 설

계 요구조건을 만족하는 e 와 c 에 대한 각 경우에서 스케일링계수 값은 다음과 같이 구해진다.

1) Case 1:

$$g_e = 2.5; g_c = 8.0; g_u = 5.0;$$
$$g_{y_e} = 2.5; g_{y_c} = 7.5; g_p = 1.0$$

2) Case 2:

$$g_e = 2.5; g_c = 10.0; g_u = 5.0;$$
$$g_{y_e} = 2.5; g_{y_c} = 7.5; g_p = 0.01$$

5. 결론 및 향후 과제

단순한 비교에는 무리가 있으나, 기존의 고전적인 제어기의 경우보다 FMRLC가 주어진 문제에 대해 설계 요구조건을 잘 충족시키고 있음을 알 수 있다. 스케일링계수들의 변화에 매우 민감한 응답을 나타내는데 특히, g_c 와 g_{y_e} 는 임의의 임계값을 경계로 시스템을 불안정하게 한다.

또한, 본 연구에서는 기준모델과 대상 항공기의 시스템을 선형으로 가정하였으며, 항공기의 속도를 일정하게 가정하고 R 이 일정한 경우와

시간에 따라 선형적으로 변화하는 경우에 대해서만 시뮬레이션을 수행하였다.

향후, 외란 등에 의한 제어기 시스템의 안정성을 확인하기 위한 연구의 보완과 비선형 시스템인 경우 및 항공기 속도가 시간에 따라 변화하는 경우에 대한 시뮬레이션을 수행할 필요가 있다.

참고문헌

1) Layne J. R. and Passino K. M., "Fuzzy Model Reference Learning Control," Journal of Intelligent and Fuzzy Systems, Vol. 4,, 1996 pp. 34-47.

2) Nho, Kyungmoon and Agarwal, R. K., "Glideslope Capture in Wind Gust Via Fuzzy Logic Controller", AIAA-99-0635, 1999.

3) Nho, Kyungmoon and Agarwal, R. K., "Automatic Landing System Design Using Fuzzy Logic", AIAA-98-4484, 1998.

4) McLean, D., Automatic Flight Control Systems, Prentice Hall, 1990.

5) Roskam, J., Airplane Flight Dynamics and Automatic Flight Controls, Part II, DARcorporation, 1995