

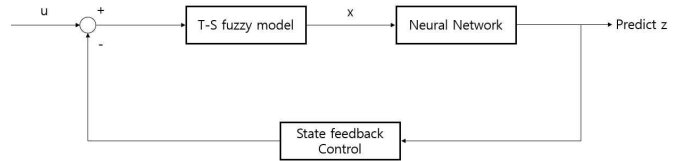
딥러닝을 이용한 T-S Fuzzy 선형화에 관한 연구

윤현섭, 엔크문크 수미야바야르, 박승규
 창원대학교 전기공학과

A Study on T-S Fuzzy Linearization using Deep Learning

Hyeonsub Yun, Enkhmunkh Sumiyabayar, Seungkyu Park
 Department of Electrical Engineering, Changwon National University

Abstract - T-S Fuzzy 시스템은 비선형 시스템을 여러 개의 선형화된 근사 모델로 나누어 퍼지적으로 결합함으로써 표현한다. 본 논문에서는 T-S Fuzzy 모델을 기반으로 비선형 시스템을 상태변환을 사용하여 선형화하는 문제를 다룬다. 기존의 T-S Fuzzy 기반 선형화에 있어 비선형 상태변환의 문제를 딥러닝 알고리즘을 도입하여 해결한 새로운 T-S Fuzzy 선형화 제어방법을 제안하고, MATLAB 시뮬레이션을 통해 더 우수한 제어성능을 확인한다.



〈그림 2〉 딥러닝을 이용한 제어시스템 블록선도

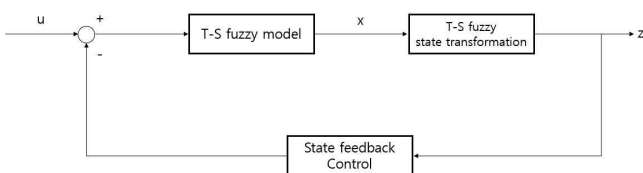
1. 서 론

T-S Fuzzy는 비선형 시스템을 여러 개의 근사 모델로 나누어 퍼지적으로 결합하여 입력을 만들으로써, 근사화된 선형 모델의 결합으로 만드는 방법으로 비선형 시스템의 제어에 효과적으로 사용되고 있다[1]. 하지만, 각각의 선형제어기는 시스템에서 안정적인 상태로 설계되어야하고[2], 시스템에 불확실성과 시간지연성이 존재하는 경우 T-S fuzzy 시스템에 선형 제어이론 적용이 어렵다는 문제가 있다[3]. 이에 T-S Fuzzy 모델을 기반으로 상태변환을 적용하여 제어 가능한 Canonical form으로 바꾸는 과정을 통해 선형화하는 방법이 제안되었다[4]. 그러나 비선형 상태변환은 수학적으로 표현되기 어렵다는 문제가 있다. 따라서 본 논문에서는 딥러닝을 통해 인공신경망을 구축하여 비선형 상태변환관계를 확립하고 상태피드백제어를 통해 z의 고유치를 조정함으로써, 비선형 시스템을 제어하는 방법을 제안한다. 그리고 MATLAB 시뮬레이션을 통해 기존의 상태변환을 이용한 T-S fuzzy 선형화 방법과 비교하여 개선된 제어성능을 확인한다.

2. 본 론

2.1 제어시스템 설계

본 논문에서는 비선형 시스템에 T-S Fuzzy를 적용하여 T-S Fuzzy 모델을 구성하였으며, 공칭제어입력은 상태피드백제어를 통해 구성하였다. 시스템의 선형화를 위해 T-S fuzzy 상태변환을 적용한 기존의 방법과 달리, 딥러닝을 사용하여 상태간의 관계를 확립함으로써, 시스템의 전체적인 선형화를 구현하였다. 그림1은 기존의 상태변환을 이용한 제어시스템 블록선도이며, 그림2는 딥러닝을 이용한 제어시스템 블록선도이다.



〈그림 1〉 기존의 제어시스템 블록선도

2.1.1 상태변환을 이용한 T-S Fuzzy 선형화

다음의 조건을 만족할 때, 모든 선형 모델들이 제어가 가능하다고 가정한다.

$$\text{rank}[B_i \ A_i B_i \ \dots \ A_i^{n-1} B_i] = n \quad (1)$$

상태변환을 통한 선형화기법을 다루기 위해 식 (2)와 같이 If-Then Rule i 를 정한다.

Plant Rule i :

If $\lambda_1(t)$ is F_{i1} , ..., and $\lambda_y(t)$ is F_{iy}

Then $\dot{x}(t) = A_i z(t) + B_i u(t)$

$$z(t) = T_i x(t) \quad (2)$$

$$\dot{z}(t) = A_{ci} z(t) + B_{ci} u(t) \quad (3)$$

$$A_{ci} = T_i A_i T_i^{-1}, \ B_{ci} = T_i B_i, \ i = 1, 2, \dots, L, \quad (4)$$

여기서 T_i 는 변환행렬이며, L 은 If-Then Rule의 개수이다.

식 (3)의 선형시스템들을 퍼지결합함으로써 전체계통을 식 (5)와 같이 나타낼 수 있다.

$$\dot{z}(t) = \sum_{i=1}^L h_i(\lambda(t)) \{A_{ci} z(t) + B_{ci} u(t)\} \quad (5)$$

변환행렬 T_i 는 i 번째 선형 시스템을 제어 가능한 Canonical form으로 만든다.

식 (1)의 가정에 의해, 변환행렬 T_i 은 식 (6)과 같이 얻어진다.

$$T_i = [B_{ci} \ A_{ci} B_{ci} \ \dots \ A_{ci}^{n-1} B_{ci}] \times [B_i \ A_i B_i \ \dots \ A_i^{n-1} B_i] \quad (6)$$

파라미터 A_{ci} 는 i 번째 선형시스템의 특성식(7)을 통해 구해진다.

$$|sI - A_i| = s^n - a_{in}s^{n-1} - \dots - a_{i2}s - a_{i1} = 0 \quad (7)$$

식 (5)는 식 (8)과 같이 선형화 된다.

$$\dot{z}(t) = \begin{bmatrix} \sum_{i=1}^r h_i z_2 \\ \sum_{i=1}^r h_i z_3 \\ \vdots \\ \sum_{i=1}^r h_i a_i z(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \sum_{i=1}^r h_i \end{bmatrix} u(t) = \begin{bmatrix} z_2 \\ z_3 \\ \vdots \\ \sum_{i=1}^r h_i a_i z(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t) \quad (8)$$

식 (9)와 같은 입력을 통해 T-S fuzzy 시스템을 제어 가능한 Canonical form으로 만들 수 있다.

$$u(t) = -\sum_{i=1}^L h_i(\lambda(t)) a_i z(t) + u_0(t) \quad (9)$$

제어입력을 구성하기 위해서는 $z(t)$ 와 $x(t)$ 간의 관계가 필요하며, 기존의 논문에서는 비선형 좌표변환을 식 (10)과 같이 결정하였다.

$$z(t) = \sum_{i=1}^L h_i(\lambda(t)) T_i x(t) \quad (10)$$

위의 식은 미분하였을 때 변환된 상태에 대한 식 (8)과의 관계가 확보될 수 없으므로 수학적 오류를 가지고 있다. 이를 해결하기 위해, 비선형 상태변환을 딥러닝으로 대체하고자 한다.

2.1.2 딥러닝을 이용한 T-S Fuzzy 선형화 시스템 제어

상태변환을 통해 선형화된 시스템의 상태 z 를 알고 있으므로 기존 T-S Fuzzy 모델의 상태 $x(t)$ 를 input, 선형화된 시스템의 상태 $z(t)$ 를 output으로 하는 신경망을 학습시켜 상태간의 관계를 확인한다. 학습에는 LSTM(Long Short Term Memory)신경망을 사용하였다.

신경망을 통해 예측된 z 는 식 (11)과 같이 나타낼 수 있다.

$$\dot{z}(t) = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & 0 & \cdots & 1 \\ \alpha_{i1} & \alpha_{i2} & \alpha_{i3} & \cdots & \alpha_{in} \end{bmatrix} z(t) + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t) \quad (11)$$

z 를 원하는 고유치값으로 보내기 위한 공칭제어입력 $u(t)$ 를 식 (12)와 같이 구성한다.

$$u(t) = -\sum_{j=1}^n \alpha_{ij} z_j + \sum_{j=1}^n \alpha'_j z_j \quad (12)$$

z 를 원하는 고유치 $\alpha'_1, \alpha'_2, \alpha'_3, \dots$ 으로 만들어 시스템의 개선된 과도응답과 정상상태응답을 확인한다.

2.2 시뮬레이션 및 결과

본 논문에서는 식 (13)과 같이 T-S Fuzzy model을 선정한다.

$$A_1 = \begin{bmatrix} 0 & 1 & 0 \\ 9.8 & 0 & 1 \\ 0 & -10 & -10 \end{bmatrix}, A_2 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -10 & -10 \end{bmatrix}, B_1 = B_2 = \begin{bmatrix} 0 \\ 0 \\ 10 \end{bmatrix} \quad (13)$$

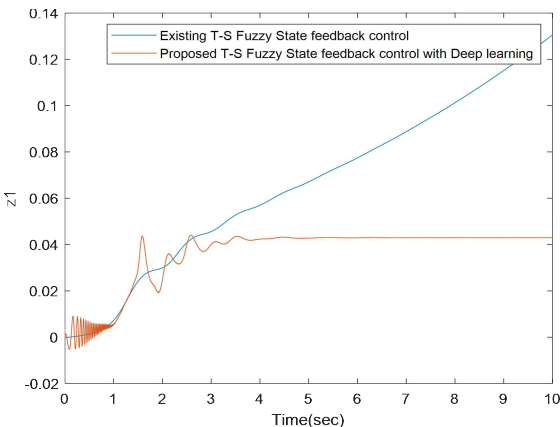
membership 함수 $h_1(t)$ 와 $h_2(t)$ 는 식 (14)와 같이 정의한다.

$$\begin{cases} h_1(x_1(t)) = \begin{cases} \sin x_1(t)/x_1(t), & x_1(t) \neq 0 \\ 1, & x_1(t) = 0 \end{cases} \\ h_2(x_1(t)) = 1 - h_1(x_1(t)) \end{cases} \quad (14)$$

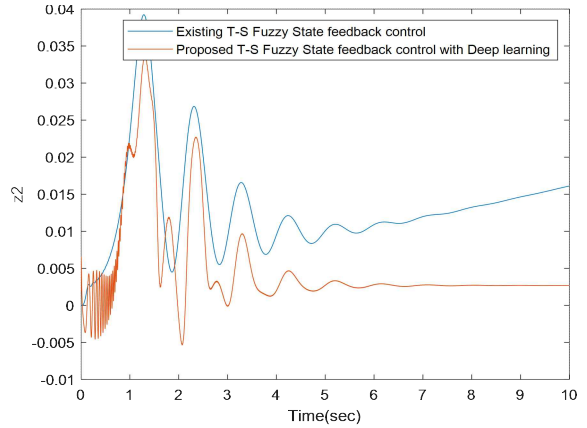
Canonical form은 식 (15)와 같다.

$$A_{c1} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 98 - 0.2 & -10 & \end{bmatrix}, A_{c2} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -10 & -10 \end{bmatrix}, B_{c1} = B_{c2} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (15)$$

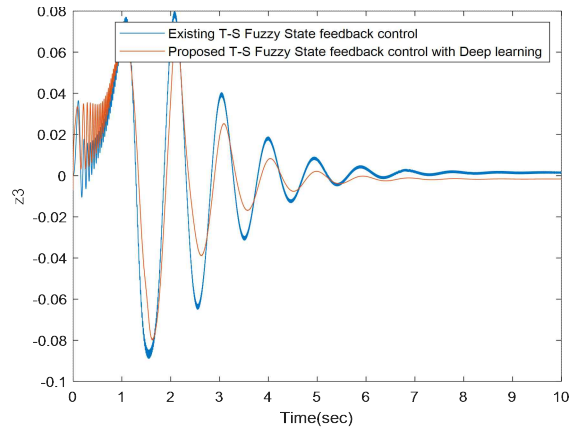
그림 3은 z_1 , 그림 4는 z_2 , 그림 5는 z_3 에 대한 제어성능을 나타낸다. 기존의 상태변환을 이용한 선형화시스템의 제어결과 z_1 과 z_2 의 경우, 정상상태에 도달하지 못하고 발산하는 경향을 보이는 반면, 딥러닝을 이용한 선형화시스템 제어의 경우 정상상태에 도달하는 모습을 확인할 수 있다. z_3 의 경우, 기존과 비슷한 경향을 보이지만 Overshoot와 정상상태응답에 도달하는 시간이 개선된 결과를 확인할 수 있다.



<그림 3> 기존 상태변환과 딥러닝 상태변환 z_1 제어성능비교



<그림 4> 기존 상태변환과 딥러닝 상태변환 z_2 제어성능비교



<그림 5> 기존 상태변환과 딥러닝 상태변환 z_3 제어성능비교

3. 결론

본 논문에서는 T-S Fuzzy 모델에 상태변환을 이용하여 근사화된 모델을 선형화하여 비선형 시스템을 제어하는 기존방법의 문제점을 다루고 있다. T-S Fuzzy 모델의 상태값을 입력, 선형화된 시스템의 상태값을 출력으로 하는 신경망 학습을 통해 상태간의 관계를 알아내고, 상태변환식을 딥러닝 블록으로 대체함으로써 T-S Fuzzy 선형화를 확립하였으며, 이를 MATLAB 시뮬레이션을 통해 확인하였다.

[참고 문헌]

- [1] Xin Tang, Donghong Ning, Haiping Du., "A Takagi-Sugeno Fuzzy Model-Based Control Strategy for Variable Stiffness and Variable Damping Suspension", IEEE Access, 8, 71628-71641, 2020
- [2] TAKAGI T, SUGENO M., "Fuzzy identification of systems and its applications to modeling and control", IEEE Trans Man and Cybernetics Systems, Part A: Systems and Humans 15, 116-132, 1985
- [3] Gang Wang, Liang Jia, "Stability and Stabilization of T-S Fuzzy Time-Delay System via Relaxed Integral Inequality and Dynamic Delay Partition", IEEE Trans Fuzzy Systems, 29(10), 2829-2843, 2021
- [4] YOON Tae-Sung, "Linearization of T-S fuzzy systems and robust H_∞ control", J. Cent. South Univ. Technol., 18, 140-145, 2011