

퍼지-PI 제어를 이용한 유도전동기 벡터제어

*박건태, *김재형, *차득근
*동의대학교 전기공학과

Vector Control of Induction Machine with Fuzzy-PI Controller

*Park gun-tae, *Kim jae-hyung, *Cha Duk-keun
*Dong-eui University

Abstract - The Induction motor Vector Control with PI controller has been widely used in industrial application. But PI control difficult in dealing with dynamic speed control, parameter variations, and load disturbances. Therefore, in this paper propose speed control of a induction motor using the PI controller with fuzzy controller. The proposed fuzzy PI controller increases the control performance of the PI controller. Simulation results show that fuzzy PI controller has a good robustness regarding the improper tuned PI controller.

1. 서 론

유도전동기 벡터제어는 고성능 제어분야 에서 많이 이용되어져 왔다. 유도전동기 벡터제어는 자속분 전류와 토크분 전류를 분리 하여 제어 하는 이론으로서, 전동기에 흐르는 고정자 전류를 동기속도로 회전하는 d-q축으로 변환하고, d축을 회전자자속 Phasor와 일치 시키면 d축성분은 자속성분(Flux component), q축 성분은 토크성분(Torque component)으로 분리할수 있다. 따라서 각각을 독립적으로 제어함으로써 타여자 직류전동기와 같은 제어특성을 얻을 수 있어, 유도전동기의 고성능 제어를 위해 많이 사용되는 이론이다. 그리고 속도 제어 및 전류제어를 위해 PI제어가 보편적으로 사용되어져 왔다. 유도전동기를 벡터제어를 채용 하여 운전할 경우 대부분 PI 제어를 이용하여 구성하게 된다. 그러나 유도전동기는 비선형 부하이므로 PI 제어기만으로 구성할 경우 제어 특성이 과도상태와 정상상태에 따라 나뉘지는 단점이 있다. 그리고 전동기 파라미터가 변화하는 환경일 경우 PI제어 게인이 파라미터 변동에 따라 변해야 한다. 이러한 문제점을 해결하기 위해 몇가지 알고리즘이 발표되었다. 퍼지제어는 PI제어의 문제점을 보완할 수 있는 우수한 제어기법의 하나이다. 퍼지제어는 비선형 또는 불확실한 시스템을 수학적 모델링없이 추론에 의해 제어 할수 있는 기법으로 많이 사용되어지고 있다. 따라서 본 논문에서는 이러한 기존의 PI 제어기의 단점을 보완하고자 PI제어기에 퍼지제어기를 추가한 제어를 이용한 유도전동기 벡터제어 기법을 제안하고 이를 MATLAB/SIMULINK를 이용한 시뮬레이션을 통해 알고리즘의 타당성을 제안 하고자 한다.

2. 본 론

2.1 유도전동기 벡터제어 시스템 설계

일반적으로 시스템에 따라 다소 약간의 차이는 있겠지만 기본적인 PI제어를 사용한 유도전동기 벡터제어 시스템은 그림1과 같이 구성되어진다.

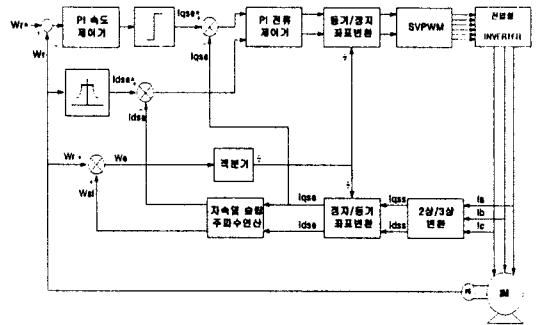


그림1 벡터제어 시스템 블럭도

2.2 퍼지 및 PI의 병렬제어기의 설계

본 논문에서는 PI제어기의 단점을 보완하고자 그림2와 같이 PI제어기에 퍼지제어기를 병렬로 접속한 형태의 속도제어기를 사용하였다. 퍼지제어기를 추가함으로써 전동기 파라미터 변동 및 외란에 강한 제어 성능을 갖도록 하였다. 퍼지제어기는 애매성을 포함하는 제어 알고리즘으로서, 퍼지제어기 특징중 하나는 if-then 형태로 구성되는 제어규칙을 전문가의 경험적 지식을 기반으로 서술하는 것으로 정확하지 않은 비선형 모델이나 유도전동기와 같이 복잡한 다변수 비선형 시스템에 적합한 제어기이다.

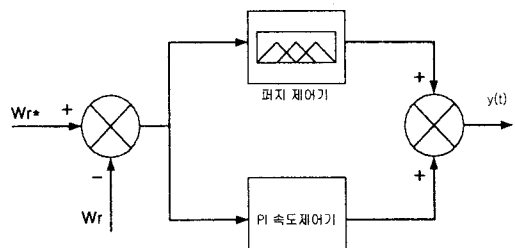


그림2 퍼지 PI 제어기

위 그림1에서 오차 신호 $e(t)$ 는 다음과 같이 주어진다.

$$e(t) = \omega_r * - \omega_r \quad (1)$$

그리고 PI제어기 출력은 식(2)와 같이 주어진다.

$$y_1(t) = K_p e(t) + K_i \int e(t) dt \quad (2)$$

퍼지 제어기는 오차신호($e(t)$) 및 오차 신호의 변화량 ($\Delta e(t)$)의 값을 얻어 이등변 삼각형을 이용하여 퍼지 화(Fuzzification)를 하였고, 퍼지 추론 법으로는 Mamdani의 최대-최소법을 이용하였다. 비퍼지화(Defuzzification)는 가장 많이 사용되어지는 무게 중심법을 사용하여 Fuzzy 제어기의 제어값인 K_{fuzzy} 값을 얻었다. 따라서 속도제어기의 최종 제어값은 식(3)과 같이 주어진다.

$$\begin{aligned} y(t)_{pi+fuzzy} &= y_1(t) + K_{fuzzy} \\ &= K_p e(t) + K_i \int e(t) dt + K_{fuzzy} \end{aligned} \quad (3)$$

2.1 퍼지제어기의 제어 규칙

고속의 퍼지 제어기를 구현하기 위해 멤버십 함수를 간단히 하여 구성하였다. 일반적으로 멤버십 함수를 7개 또는 5개로 구성할 경우, 제어규칙은 49개 및 25개가 존재 하게 된다. 이와 같은 경우 비퍼지화를 할 경우 많은 연산을 필요로하며 제어 규칙 또한 복잡해진다. 본 논문에서 사용한 퍼지, PI 속도 제어기는 PI제어기가 병렬로 연결된 상태 이므로, 멤버십 함수를 줄여 입력단에 2개의 멤버십 함수를 사용하였고, 출력단에는 4개의 멤버십 함수를 사용한 간략화된 퍼지 제어기를 사용하였다. 그림 3은 속도 오차에 대한 멤버십 함수를 보여주고 있고,

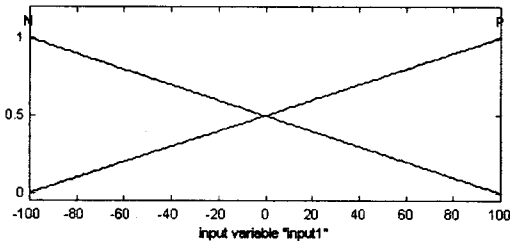


그림3 속도오차 멤버십 함수

그림4는 속도오차 변화량의 멤버십 함수를 보여주고 있다.

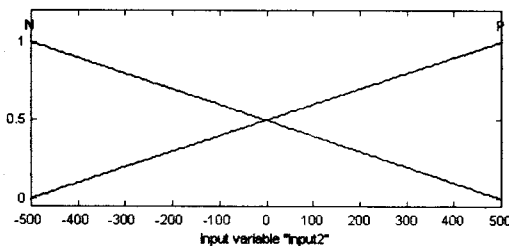


그림4 속도오차 변화량 멤버십 함수

표1은 퍼지제어기의 규칙에 대한 것을 보여주는 것으

로 총 4개의 퍼지 규칙을 이용하여 퍼지 제어를 행하였다.

E \ CE	N	P
N	NL	PS
P	NS	PL

표1 퍼지 제어기 규칙

NS: Negative Small
 NL: Negative Large
 PS: Positive Small
 PL: Positive Large

위 제어 규칙에서 보듯이 입력오차가 Negative 이고 오차 변화량이 Negative 일 경우 제어값은 크게(NL) 나가고, 입력 오차가 Negative이고 입력 오차 량이 Positive 이면 제어값은 작게(NS) 나가는 방향으로 퍼지 규칙을 만들었다.

3. 퍼지 PI 제어기의 시뮬레이션

시뮬레이션에 사용된 속도제어기의 퍼지 PI제어기는 그림5와 같이 구성하였고, 전체 시스템의 시뮬레이션을 위한 블록도는 그림6과 같이 구성하여, 퍼지제어기가 제공되는 MATLAB/SIMULINK 에 의해 시뮬레이션 하였다.

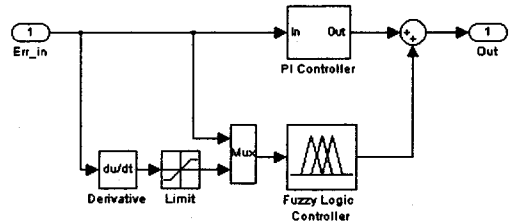


그림5 퍼지 PI 제어기의 블록도

속도 Error는 $Err_in(Wrpm^* - Wrpm)$ 단자에 연결되어 있고, PI제어기 및 퍼지제어기를 거친 값이 Out 단자로 출력된다.

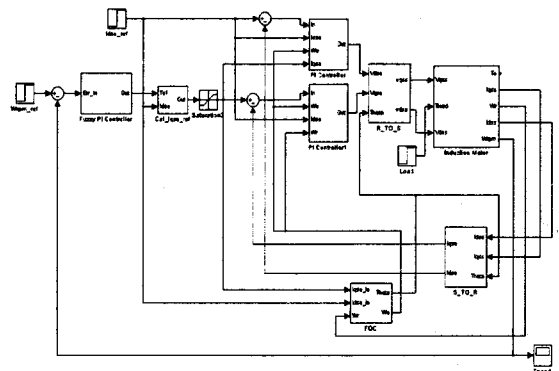


그림6 퍼지-PI제어기를 이용한 시뮬레이션 블록도

4. 시뮬레이션 결과

아래 파형들은 은 PI제어를 사용하였을 경우와 퍼지-PI제어를 사용하였을 경우에 대한 속도 응답 파형을 보여주고 있다. 그림7은 일반적인 PI제어 계인에 대한 속도 응답 파형으로 약간의 오버슈트(Over shoot) 및 언더슈트(Under shoot)가 발생하고 있다.

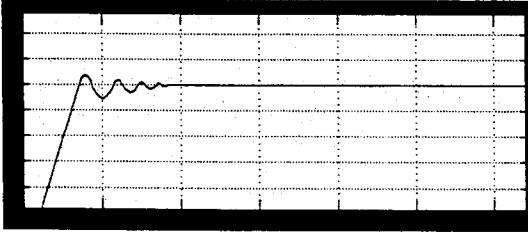


그림7 PI 제어기(1000RPM)

그림8은 퍼지-PI제어를 사용하였을 경우 파형으로 그림7보다는 속도 응답특성이 많이 개선되었음을 알 수 있다.

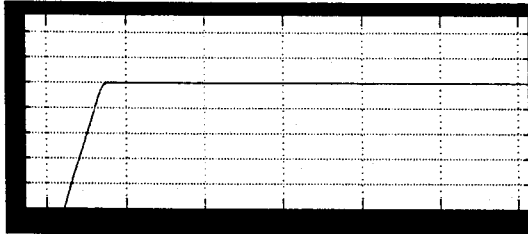


그림8 퍼지-PI 제어기(1000RPM)

그림9는 부적절한 PI제어 이득을 사용하였을 경우 속도 응답 파형으로 오버슈트(Over shoot) 및 언더슈트(Under shoot)가 많이 발생하고 있다.

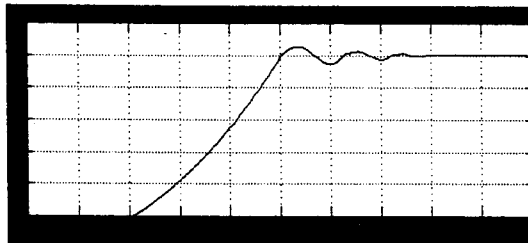


그림9 PI 제어기(1000RPM)

그림10은 퍼지-PI제어를 사용하였을 경우 속도 응답 파형을 보여주는 것으로 속도 응답특성이 PI제어기만 사용하였을 경우보다는 많이 개선되었음을 알 수 있다.

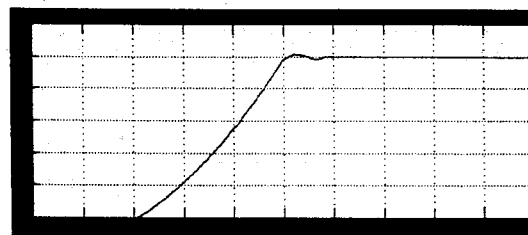


그림10 퍼지-PI 제어기(1000RPM)

시뮬레이션 결과 PI제어기만으로 구성하였을 경우 보다 퍼지-PI제어기를 구성하였을 경우 안정된 속도 응답 특성을 얻을수 있었으며, 부적절한 PI제어 계인 대해서도 속도 특성이 많이 개선이 됨을 알수 있었다.

5. 결 론

본 논문에서는 PI제어기 및 퍼지 제어기를 병렬로 구성하여 속도제어를 하였으며, PI제어기만을 사용 하였을 경우보다 속도제어에서 좋은 성능을 나타냄을 알 수 있었다. 기존의 PI제어기 에서는 부적절한 제어기 계인이나 전동기 파라메터가 변동 할 경우에 동특성이 나빠지는 단점을 가지고 있다. 그러나 퍼지 제어기를 추가할 경우 이러한 PI제어기의 단점을 어느 정도 보완해 줌과 동시에 또한 전동기 파라메터 변동에 강한 특성을 나타냄을 알수있었다.

(참 고 문 헌)

- [1] Brian Heber, Longya Xu, Yifan Tang, "Fuzzy Logic Enhanced Control of an Indirect Field-Oriented Induction Machine Drive", IEEE, 772~778, September, 1997
- [2] Emanuele Cerruto, Alfio Consoli, Angelo Raciti, "Fuzzy Adaptive Vector Control of Induction Motor Drives", IEEE, 1028~1040, November, 1997
- [3] A. Cataliotti, G. Poma, "Fuzzy-logic approach for easy and robust control of an induction motor", TheEPE'97, 2421~2425, 1997
- [4] Mao-Fu Lai, Michio Nakano, Guan-Chyun Hsieh, Vichai Saehcut, Wen-Yuh Chiu, "Design of Fuzzy Logic Controller in PLL-Based Servo Motor Speed Drive", IEEE, 353~358, 1996
- [5] Chee-Mun Ong, "Dynamic Simulation of Electric Machinery"