

GA-P제어기를 이용한 유도전동기 간접 벡터제어 시스템

이학주^{*} 권성철^{*} 성세진^{**}
^{*} 한전 전력연구원 ^{**} 충남대학교 전기공학과

An Indirect Vector Control System of Induction Motor using Genetic Algorithm based PI Controller

Lee HakJu^{*} Kwon SungChul^{*} Seong SeJin^{**}

^{*} KEPRI (Korea Electric Power Research Institute) ^{**} Dept. of Electrical Engineering Chungnam National University

Abstract - This paper presents the use of a simple genetic algorithm for the tuning of a proportional-integral speed controller for an induction motor drive. The influence of population size, generation number and rate of mutation on the convergence of the genetic algorithm is investigated. On Matlab/Simulink environment, this paper proposes an optimal GA-PI controller of indirect vector control for induction motor drive system. The simulation results verify that the system has a more robust to the parameter variation than classical PI controller.

1. 서 론

산업용 전동기 제어에는 구조가 간단하고 적용하기 쉬운 일정이득의 PI제어기가 널리 사용되지만, 시스템이 동적, 비선형인 경우 일정 이득의 PI제어기로는 제어기의 고성능을 기대하기 어렵다. 이러한 문제를 해결하기 위하여 적응제어, 신경회로망을 이용한 제어, 퍼지제어 및 유전자 알고리즘 등의 연구가 활발히 진행되고 있다. 본 논문에서는 유도전동기 속도제어를 위해 PI제어기와 유전자알고리즘을 합성한 방법을 제안한다. 유전자 알고리즘은 1970년대 초반 John Holland에 의해 최초로 도입되어 최적화 탐색문제에 중요한 알고리즘으로 자리매김을 하고 있다. 1975년 Holland는 자연의 진화과정이 인공시스템에 적용되는가를 설명, 유전자 알고리즘을 체계화 하였다. 이 유전자 알고리즘은 다른 최적화방법과 비교할 경우 다음과 같은 특징을 가진다. 파라미터의 연속성과 미분치의 존재와 같은 조건이 필요하지 않고, 전역 탐색으로 국부수렴에 대한 확률을 줄고, 알고리즘 수행을 위한 보조정보가 필요 없는 특징이 있어 최적화 문제 및 제어시스템의 파라미터 동조에 유용하게 사용되고 있다. 본 논문에서도 유전자 알고리즘을 기반으로 유도전동기 속도제어를 위한 PI제어기를 설계하고, 제어 성능을 확인하기 위하여 유전자 알고리즘의 연산 파라미터 변화에 대한 전동기 스텝속도 응답 변화를 통하여 유전자 알고리즘의 적용에 대한 유효성의 증명과 최적 파라미터를 결정하고 있다. 본 논문의 구성은 다음과 같다. 2.1 및 2.2절에서는 유도전동기 수학적 모델링 및 벡터제어 시스템을 Matlab을 이용하여 구성하고, 2.3절은 유전자 알고리즘의 소개와 제어기 설계 및 제안한 제어기의 유전자 연산 파라미터 변화에 대한 유도전동기의 속도응답에 의해 제어기의 성능을 평가하고 3장에서는 시스템 시뮬레이션 및 일반적인 P제어기와와의 성능비교로 제안한 제어기의 우수성을 확인 후 4장에서 결론을 맺고자 한다.

2. 유도전동기 벡터제어 시스템

2.1 유도전동기 수학적 모델링

고정자속 정직좌표계에서 유도전동기의 고정자와 회전자의 d-q축 전압방정식은 다음의 식(1)~(4)와 같다.

$$v_{ds} = R_s i_{ds}^s + \frac{d\lambda_{ds}^s}{dt} \tag{1}$$

$$v_{qs} = R_s i_{qs}^s + \frac{d\lambda_{qs}^s}{dt} \tag{2}$$

$$0 = R_r i_{dr}^s + \frac{d\lambda_{dr}^s}{dt} + w_r \lambda_{qr}^s \tag{3}$$

$$0 = R_r i_{qr}^s + \frac{d\lambda_{qr}^s}{dt} - w_r \lambda_{dr}^s \tag{4}$$

동기좌표계(d-q축)에서의 전압방정식은 다음과 같다.

$$\begin{bmatrix} v_{ds} \\ v_{qs} \\ v_{dr} \\ v_{qr} \end{bmatrix} = \begin{bmatrix} R_s + pL_s & w_r L_s & pL_m & w_r L_m \\ -w_r L_s & R_s + pL_s & -w_r L_m & pL_m \\ pL_m & (w_r - w_r)L_m & R_r + pL_r & (w_r - w_r)L_r \\ -(w_r - w_r)L_m & pL_m & -(w_r - w_r)L_r & R_r + pL_r \end{bmatrix} \begin{bmatrix} i_{ds}^s \\ i_{qs}^s \\ i_{dr}^s \\ i_{qr}^s \end{bmatrix} \tag{5}$$

p는 미분연산자, $w_{sl} = w_e - w_r$ 은 슬립각속도이며, 쇠교자속은 다음과 같이 표현된다.

$$\begin{aligned} \lambda_{qs} &= L_s i_{qs}^s + L_m (i_{qs}^s + i_{qr}^s) \\ \lambda_{qr} &= L_r i_{qr}^s + L_m (i_{qs}^s + i_{qr}^s) \\ \lambda_{ds} &= L_s i_{ds}^s + L_m (i_{ds}^s + i_{dr}^s) \\ \lambda_{dr} &= L_r i_{dr}^s + L_m (i_{ds}^s + i_{dr}^s) \end{aligned} \tag{6}$$

위 식(6)을 이용하여 회전자 자속과 회전자 전류를 구하면 다음 식(7)~(10)과 같다.

$$\frac{d\lambda_{dr}^s}{dt} = \frac{1}{\tau_r} (-\lambda_{dr}^s - w_r \tau - r \lambda_{qr}^s + L_m i_{ds}^s) \tag{7}$$

$$\frac{d\lambda_{qr}^s}{dt} = \frac{1}{\tau_r} (-\lambda_{qr}^s + w_r \tau - r \lambda_{dr}^s + L_m i_{qs}^s) \tag{8}$$

$$i_{dr}^s = \frac{1}{L_r} (\lambda_{dr}^s - L_m i_{ds}^s) \tag{9}$$

$$i_{qr}^s = \frac{1}{L_r} (\lambda_{qr}^s - L_m i_{qs}^s) \tag{10}$$

단, $\tau_r = \frac{L_r}{R_r}$ 이고

발생 토크를 동기좌표계의 d-q축 전류와 자속으로 나타내면 다음과 같다.

$$T_e = \frac{3}{2} \frac{P}{2} \frac{L_m}{L_r} (i_{qs}^s \lambda_{dr}^s - i_{ds}^s \lambda_{qr}^s) \tag{11}$$

여기서, P는 극수이다.

2.2 간접 벡터제어 시스템

유도전동기의 자속과 토크를 분리하여 제어하기 위하여 아래의 식(12)가 만족하여야 한다. 이는 회전자의 자속을 정확하게 추정하여 동기좌표계의 d축을 회전자 자속에 일치 시켜야 가능하다.

$$\lambda_{qr} = \frac{d\lambda_{dr}}{dt} = 0 \tag{12}$$

$$\lambda_{dr} = \lambda_r = \text{constant} \quad (13)$$

$$\frac{d\lambda_{dr}}{dt} = 0 \quad (14)$$

$$\omega_{sl} = \frac{L_m}{\lambda_{dr}} \left(\frac{R_r}{L_r} \right) i_{qs} \quad (15)$$

$$\omega_{sl} = \frac{R_r}{L_r} \cdot \frac{i_{qs}}{i_{ds}} \quad (16)$$

2.3 제어기 설계

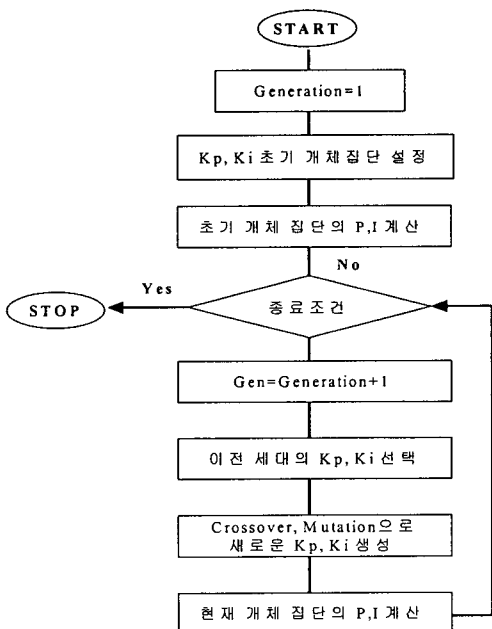
2.3.1 유전자 알고리즘 개요

본 논문에서 사용한 유전자알고리즘은 SGA(Simple Genetic Algorithm)에 각 개체의 선택 확률을 계산하여 룰렛 휠의 슬롯을 할당하는 룰렛 휠을 적용하였으며, 적용과정을 설명하면 다음과 같다.

- [1] 각각의 고정된 길이의 문자열을 초기 해집단을 난수로 생성하여 부호화(Coding) 한다.
- [2] 정지조건이 만족될 때까지 다음 항목을 반복적으로 수행한다.
 - 해집단의 적합도를 평가한다.
 - 다음과 같은 연산으로 새로운 문자열 집단을 생성한다.
 - ① 최대적합도 문자열을 복제(Reproduction)한다.
 - ② 문자열을 복제한다.
 - ③ 두 문자열을 교배(Crossover)하여 새로운 문자열을 생성한다.
 - ④ 한 문자열의 돌연변이를 통하여 새로운 문자열을 생성한다.
- [3] 정지조건을 만족하면 문제의 해에 대한 결과를 출력하고 종료한다.

2.3.2 유전자 알고리즘을 이용한 PI제어기 설계

본 논문의 목적은 유도전동기 간접 벡터제어 시스템에 있어서 속도 PI제어기의 이득 최적화로 유전자 알고리즘을 적용한 PI제어기의 설계과정은 다음과 같다.



【그림 1】 유전자 알고리즘 흐름도

따라서, 제어기를 설계한다 함은 K_p , K_i 를 결정하는 것으로 Ziegler-Nichols방법과 Relay 자동동조기를 사용하는 방법이 있지만 본 논문에서는 최적제어를 이용하여 시스템 오차를 최소화하는 K_p , K_i 를 결정하는 것으로 성능지수는 ITAE를 적용하였다.

본 논문의 최적화 문제는 아래와 같이 표현된다.

$$\text{Minimize } P. I = \int_0^t |e(t)| dt$$

$$\text{bounded to } K_{pl} < K_p < K_{pu}, \quad K_{il} < K_i < K_{iu}$$

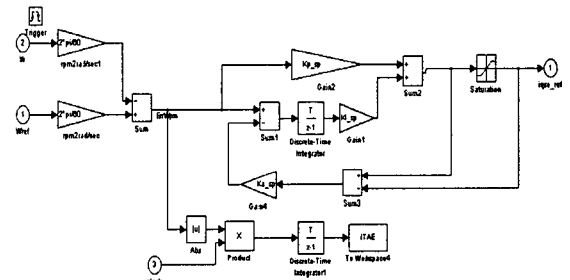
유도전동기 PI 속도제어기는 다음과 같이 표현된다.

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt$$

여기서, $e(t) = y(t+1) - y(t)$ 이다.

유전자 알고리즘에 의해 PI제어기의 최적계수 조정을 위해 $e(t)$ 를 알아야 하고, K_p , K_i 를 이진코딩에 의해 임의체로 표현한다. 또한, 본 논문에서는 $0 < K_p < 100$, $0 < K_i < 10$ 으로 계수 범위를 설정하였다.

다음 그림은 본 논문에서 구성한 PI제어기의 Simulink 블록도이다.



【그림2】 PI제어기의 Simulink Block

K_p 와 K_i 를 전역변수로 설정, 유전자 알고리즘의 연산에 의해 최적의 해를 구한다. 본 제어기는 Anti-windup으로 구성되어 있다.

3. 시스템 시뮬레이션

3.1 시뮬레이션 개요

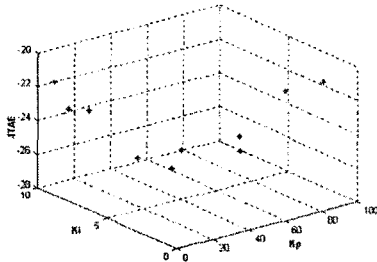
본 논문에서 유전자 알고리즘의 연산 파라미터 변화가 탐색성능에 미치는 영향을 고찰하기 위하여 집단의 크기, 세대수 및 돌연변이율을 변화시켜 제어기 성능지수 ITAE를 구하였다. 우선 탐색체수를 10bit, 20bit로 변화를 주고 세대수를 10과 20세대로 구분한 경우에 대해 돌연변이율을 각각 0%, 1%, 10%로 변화시켜 K_p 와 K_{isp} 의 최적해 및 성능지수를 구하였다. 여기서 교배율은 0.65, 선택방법은 룰렛 휠을 적용하였다. 목적함수에 (-)을 곱하여 최대화를 수행하므로 세대가 증가함에 따라 목적함수가 증가하지만 그 크기는 작아진다. 얻어진 최적해를 유도전동기 벡터제어 시스템에 적용하여, 전동기의 스텝 속도 응답을 통해 제어기의 우수성을 확인한다.

3.2 결과 및 고찰

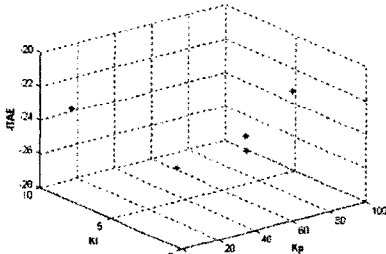
유전자 알고리즘의 파라미터 변화에 따른 PI제어기의 계수 결정의 최적해는 염색체수가 10이고 10세대에서 구할 수 있었다.

다음 그림은 유전자 알고리즘에 의한 초기 및 세대별 해집단이다. 그림에서 알 수 있듯이 10세대 연산 후 $K_p=68.2076$, $K_i=1.0693$ 에 수렴하고 있으며, 이 때 성능지수 $-ITAE$ 는 -20.9065 이다.

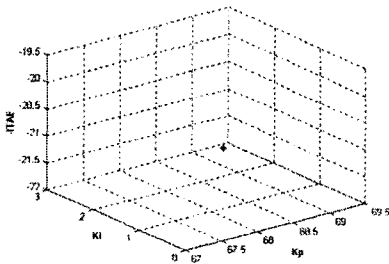
또한, 염색체수를 20, 세대수를 20으로 설정하여 유전자 알고리즘을 수행한 결과 $K_p=4.4226$, $K_i=1.4272$ 이며 성능지수는 -20.8441 이 된다. 다음 [그림3]에서 [그림6]은 최적해 수렴과정을 보이고 있다.



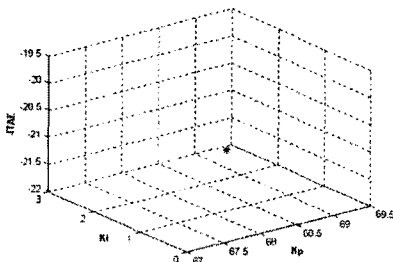
【그림3】 초기 개체



【그림4】 3세대 연산 후 개체

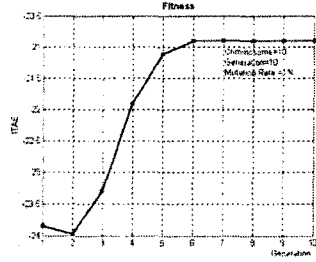


【그림5】 7세대 연산 후 개체



【그림6】 10세대 연산 후 개체

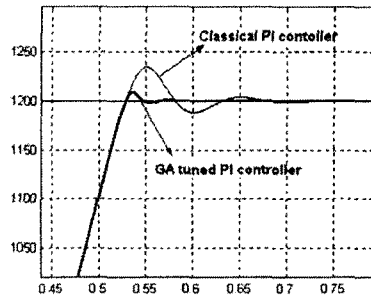
위 그림에서 알 수 있듯이 본 논문에서 설계한 유전자 알고리즘 PI제어기는 개체수를 세대수를 10 결정해도 충분함을 알 수 있다. 【그림7】은 본 논문에서 적용한 유전자 알고리즘의 적합도이다.



【그림7】 적합도

유전자 알고리즘에 의해 얻어진 최적해를 유도전동기 속도 PI제어기에 적용한 경우의 스텝 속도응답을 [그림8]과 [그림9]에 보이고 있다. 【표1】은 본 논문에서 적용한 유도전동기 파라미터이다.

전동기 정격	3.75 [kW]	R_s	0.6992 [Ω]
정격속도	1730 [rpm]	R_r	0.3552 [Ω]
극수	4	L_s	0.0661 [H]
J	0.0918[kg.m ²]	L_m	0.0632 [H]



【그림8】 유도전동기 스텝 속도응답

4. 결 론

본 논문에서는 유도전동기 속도제어를 위하여 PI제어기에 유전자 알고리즘의 합성을 하였다. 유전자 알고리즘을 이용한 오프 라인관점에서 최적 파라미터 동조시 개체수를 10, 세대수를 10, 돌연변이율을 1%한 경우에 최적해를 구할 수 있었다. 시뮬레이션 결과는 유도전동기 구동시스템에 유전자 알고리즘의 적용가능성을 보이고 있으며, 시스템의 속도 응답 향상을 위한 유전자 알고리즘 파라미터를 결정 하여 제어기의 성능이 최적에 가까운 제어특성을 얻을 수 있었다.

【참 고 문 헌】

- [1] WG da Silva et al, "Tuning of a Brushless DC Drive Speed Controller with an On-Line Genetic Algorithm", EPE '99
- [2] 박보선 외, "퍼지제어기와 유전알고리즘에 의한 BLDC 모터의 속도제어", 대한전기학회 논문지 Vol.44, No.11, pp. 1505-15109, 1995. 11
- [3] Chipperfield, A., et al, "Genetic Algorithm Toolbox for Use with MATLAB", Dept. of Automatic Control and Systems Engineering, University of Sheffield, England