

## 지능형 AC 서보 제어드라이브의 개발에 관한 연구

김동완<sup>o</sup> · 황기현

<sup>o</sup> 동명대학 전기전자계열 · 부산대학교 컴퓨터 및 정보통신 연구소

### A Study on Development of Intelligent AC Servo Control Drive

Dong-Wan Kim<sup>o</sup> · Gi-Hyun Hwang

<sup>o</sup> Department of Electricity, Tongmyong College

Research Institute of Computer, Information and Communication at Pusan National Univ

**Abstract** - We propose an Tabu search changing neighborhood solution's range to be searched each iteration according to an objective function. It is applied for designing the scaling factors of Fuzzy Logic Controller (FLC) using the proposed Tabu search. We apply it to the speed control of AC Servomotor to evaluate the usefulness of the proposed method. As a result of the computer simulation, the FLC shows the better performance than PI controller in terms of overshoot and settling time.

## 2. Tabu 탐색법

### 2.1 Tabu 탐색법의 개요

Tabu Search는 경험적인 최적화 알고리즘으로써 다른 최적화 탐색 알고리즘에 비해 최적해 탐색속도가 비교적 빠르고 알고리즘이 단순하여 쉽게 프로그래밍 할 수 있는 장점이 있다. 그러나 Tabu Search의 최적해 탐색성능은 초기해, 이웃해 선정, Tabu List의 크기 등의 영향을 받는다. 초기해를 잘못 선정하면 국부 최소값에 수렴할 가능성이 있기 때문에 일반적으로 전문가의 경험, 또는 수치적 방법으로 초기해를 선정한다. 이웃해의 정의 및 크기는 대상 문제나 탐색 전략에 따라 달라지는데, 이웃해를 잘못 선정할 경우 불필요한 탐색으로 인해 탐색영역이 증가하고 수행시간이 많이 소요된다. 새로운 이웃해를 탐색하는 동안에 이전에 탐색한 영역을 다시 탐색하는 것을 방지하기 위해 Tabu List를 사용한다. Tabu List의 크기는 탐색의 진행 방향과 밀접한 관계가 있는데, Tabu List의 크기가 클수록 이전에 탐색되었던 해가 오랫동안 재탐색되지 않는데, 이는 탐색공간을 다양하게 탐색할 수 있도록 하는 역할을 한다. 반면에 Tabu List의 크기가 작으면 이런 제약이 약해지므로 한번 탐색했던 곳이라도 재탐색될 확률이 높아 같은 탐색공간을 반복적, 세부적으로 탐색할 수 있도록 한다. 즉 Tabu List의 사용을 통해 한번 방문되었던 국부 최적해를 재방문하는 것을 방지할 수 있으며 탐색공간을 다양하게 또는 한곳을 집중적으로 탐색할 수 있다. 그러나 Tabu List에 있는 해라 할지라도 평가 결과가 정해진 조건을 만족할 경우에는 이를 새로운 이웃해로 선정할 수 있는데 이를 열망 조건이라고 한다. 일반적으로 현재까지의 최적해보다 Tabu List에 속한 해의 평가 결과가 더 좋은 경우에는 열망 조건이 만족된다.

## 1. 서론

1980년대 중반에 Glover에 의해서 연구되기 시작한 Tabu Search는 경험적인 최적화 알고리즘으로 일반적인 수치적 방법에 비해 국부 최소값에 빠질 위험이 적으면서 유전 알고리즘이나 시뮬레이트 어닐링(Simulated Annealing)과 같은 전역 최적해 탐색방법에 비해 대상 문제에 관한 지식을 활용하기가 유리한 탐색기법이다. 기존의 최적화 알고리즘은 대상 문제의 복잡도가 높고 탐색공간이 클 경우 최적해에 수렴하기 위해 많은 계산시간이 소요된다는 단점을 가진다. 이에 비해 Tabu Search는 후보해의 생성에 직접 관여함으로써 복잡한 문제에 대해 효율적인 탐색을 가능하게 함으로써 빠른 수렴속도를 가진다. 그리고 다른 최적화 알고리즘과 결합하기 쉬운 특성 때문에 근래에는 여러 분야에서 활발히 연구되고 있다[1-3].

Tabu Search의 최적해 탐색성능은 초기해, 이웃해 선정, Tabu List의 크기 등의 영향을 받는다. 초기해를 잘못 선정하면 국부 최소값에 수렴할 가능성이 있기 때문에 일반적으로 전문가의 경험 또는 수치적 방법으로 초기해를 선정한다. 이웃해의 정의 및 크기는 대상 문제나 탐색 전략에 따라 달라지는데, 이웃해를 잘못 선정할 경우 불필요한 탐색으로 인해 탐색영역이 증가하고 수행시간이 많이 소요된다.

본 논문에서는 탐색영역내에서 대상 문제나 탐색 전략에 따라 현재해에 대한 이웃해를 생성한 후 이를 목적함수에 따라 평가한 후, 목적함수에 따라 매 반복횟수마다 탐색할 이웃해의 범위를 적응적으로 변경하는 Tabu Search를 제안하였다. 제안한 Tabu Search를 이용하여 퍼지제어기의 입·출력 이득을 튜닝하였다. 제안한 방법의 유용성을 평가하기 위해 AC 서보 모터의 속도제어에 적용하였다. 컴퓨터 시뮬레이션 결과, 제안한 퍼지제어기가 오버슈트 및 정정시간면에서 PI 제어기보다 좋은 제어성능을 나타내었다.

### 2.2 Tabu 탐색법의 구성

일반적인 Tabu Search의 주요 과정은 초기해 생성, 이웃해 선정 및 평가, 모의해 생성, Tabu Test, 열망조건 및 다양성으로 구성되어 있으며, 이를 이용하여 해를 탐색하는 과정은 다음과 같으며 전체적인 흐름도는 그림 1과 같다.

**단계 1) 초기해 선정:** 주어진 문제의 제약조건을 고려하여 경험적인 방법으로 초기해를 생성한 후 이를 현재해로 설정한다.

**단계 2) 이웃해 생성 및 평가:** 탐색영역내에서 대상 문제나 탐색 전략에 따라 현재해에 대한 이웃해를 생성한 후 이를 목적함수에 따라 평가한다.

**단계 3) 모의해 생성:** 단계 2)에서 생성된 이웃해 중에서 평가값이 가장 우수한 해를 모의해로 설정한다.

**단계 4) Tabu Test:** 모의해가 Tabu List에 포함되는지를 검토한다. 만일 모의해가 Tabu List에 포함될 경우 단계 5)를 실행하고 그렇지 않을 경우 단계 6)를 실행한다.

**단계 5) 열망조건:** 모의해가 Tabu List에 포함되더라도 열망조건을 만족할 경우에는 Tabu List에서 제외되어 다음 탐색시의 현재해로 설정된다.

**단계 6) 현재해 갱신:** 모의해가 Tabu List에 포함되지 않으면 다음 탐색의 현재해로 설정된다.

**단계 7) 종료조건:** 종료조건이 만족할 때까지 단계 2) ~ 단계 6) 과정을 반복한다.

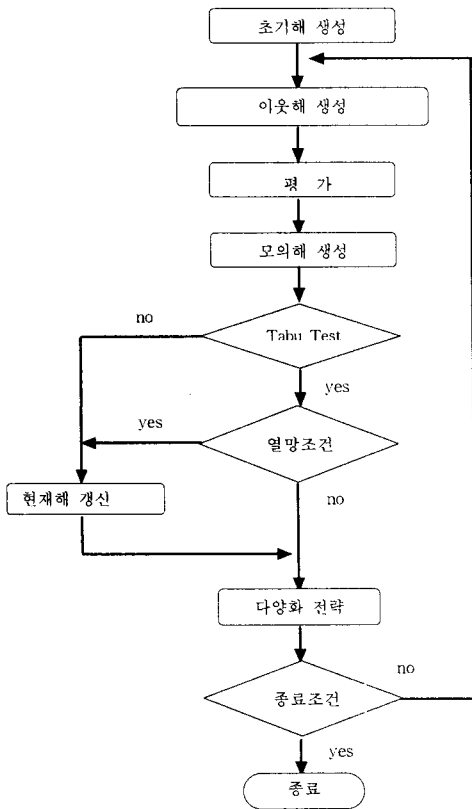


그림 1 Tabu 탐색법의 흐름도

### 3. Tabu 탐색법을 이용한 퍼지제어기의 설계

일반적인 Tabu Search의 성능은 초기해 생성, 이웃해 선정, Tabu List 등에 크게 영향을 받는다. 따라서 본 논문에서는 이웃해 선정시 목적함수에 따라 이웃해의 범위가 적응적으로 변경되는 Tabu Search를 제안하였고, 그림 2는 본 논문에서 제안한 Tabu Search를 이용하여 퍼지제어기의 입·출력 이득을 튜닝하기 위한 구성도를 나타낸다. 그림 2에서 보는 것처럼 퍼지제어기의 입력으로 식 (1)과 같이 오차와 오차의 변화분을 사용하였고, 퍼제어기의 입·출력 이득을 설계하는 과정은 다음과 같다.

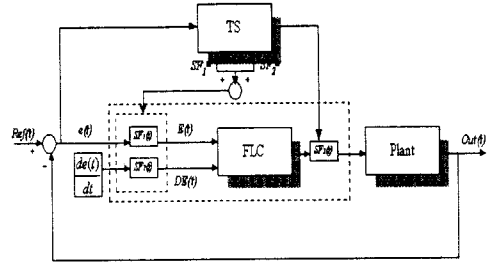


그림 2 Tabu 탐색법을 이용한 FLC 튜닝을 위한 구성도

$$E(t) = SF_1(t) e(t) \quad (1)$$

$$DE(t) = SF_2(t) \frac{de(t)}{dt}$$

여기서,  $e(t) = Ref(t) - Out(t)$

$SF_1(t), SF_2(t)$ : 퍼지제어기의 오차 및 오차변화분의 이득

$Ref(t)$ : 기준입력

$Out(t)$ : 실제출력

**단계 1) 초기해 선정:** 주어진 문제의 제약조건을 만족하도록 식 (2)와 같이 초기해를 생성한 후, 이를 현재해로 설정하였다.

$$S_0 = \text{Random} [ SF_{\max}, SF_{\min} ], i = 1, 2, 3 \quad (2)$$

여기서,  $SF_{\max}$ :  $i$  번째 퍼지제어기의 이득의 상한치

$SF_{\min}$ :  $i$  번째 퍼지제어기의 이득의 하한치

$i$ : 퍼지제어기의 입·출력 이득의 개수

**단계 2) 이웃해 생성 및 평가:** 탐색영역내에서 대상 문제나 탐색 전략에 따라 현재해에 대한 이웃해를 생성한 후, 이를 식 (3)과 같이 설정한 목적함수에 따라 평가한다.

$$J = \text{Minimize} \sum_{k=1}^N |E(k)| \quad (3)$$

여기서,  $E(k)$ :  $k$ 번째 기준입력과 실제출력과의 오차

본 논문에서 제안한 이웃해 생성 방법은 그림 3에 나타내었고, 그 과정은 다음과 같다.

- ① 그림 3에서 보는 것처럼 제약조건을 만족하는 초기해  $S_0$ 에 대해서 탐색할 이웃해의 범위( $R^i$ )를 결정한다.
- ② 식 (4)에 의해서 매 반복횟수마다 탐색할 이웃해의 범위( $R^i$ )가 적응적으로 개선하였다.

$$R^{i+1} = \begin{cases} c_d \times R^i & \text{if } \alpha(t) < \delta \\ c_i \times R^i & \text{if } \alpha(t) > \delta \\ R^i & \text{if } \alpha(t) = \delta \end{cases} \quad (4)$$

여기서,  $\alpha(t)$ :  $t$  반복횟수에서 이웃해 중에서 상위 10%에 대해 표준편차

$R^i$ : 탐색할 이웃해의 범위

$c_d, c_i$ :  $R^i$ 의 감소율과 증가율

$\delta$ : 상수

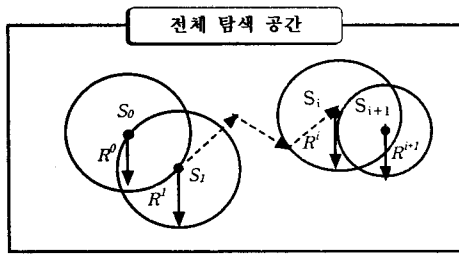


그림 3 이웃해 생성 방법

**단계 3) 모의해 생성:** 단계 2)에서 생성된 이웃해 중에서 평가값이 가장 우수한 해를 모의해로 설정한다.

**단계 4) Tabu Test:** 모의해가 Tabu List에 포함되는지를 검토한다. 본 논문에서의 Tabu List는 반복횟수마다 탐색된 최적해로 구성되며 FIFO(first in first out)방법을 사용하였다. 현재까지 탐색한 최적해와 이웃해중에서 목적함수 값이 가장 우수한 모의해와의 유클리언 거리가 일정값 이하가 되는 경우에는 Tabu로 설정하였다. 만일 모의해가 Tabu List에 포함될 경우 단계 5)를 실행하고 그렇지 않을 경우 단계 6)를 실행한다.

**단계 5) 열망조건:** 모의해가 Tabu List에 포함되더라도 열망조건을 만족할 경우에는 Tabu List에서 제외되어 다음 탐색시의 현재해로 설정된다. 본 논문에서는 모의해가 Tabu일지라도 현재까지 탐색한 최적해보다 우수한 경우를 열망조건으로 사용하였다.

**단계 6) 현재해 갱신:** 모의해가 Tabu List에 포함되지 않으면 다음 탐색의 현재해로 설정된다.

**단계 7) 종료조건:** 일정 반복횟수까지 단계 2) ~ 단계 6) 과정을 반복한다.

#### 4. 사례 연구

제안한 퍼지제어기의 제어성능을 검증하기 위하여 AC 서보 모터의 속도제어에 적용하여 컴퓨터 시뮬레이션을 수행하였다. AC 서보 모터의 속도제어에 사용된 Tabu 탐색법의 시뮬레이션 계수는 표 1과 같다. 표 1과 같은 조건에서 Tabu 탐색법으로 퍼지제어기의 입·출력 이득을 튜닝하였고, 튜닝된 퍼지제어기의 이득은 각각 1.24, 0.78, 0.14 이다.

그림 4은 Tabu 탐색법을 이용하여 퍼지제어기의 설계에 사용된 1000(rpm)에 대한 제안한 퍼지제어기와 PI 제어기의 응답 특성을 비교하였다. 그림 4에서 보는 것처럼, 제안한 퍼지제어기가 PI 제어기보다 더 나은 제어성능을 보임을 알 수 있다. 그림 5는 제안한 퍼지제어기의 강인성을 평가하기 위해 퍼지제어기 설계에 사용되지 않은 조건인 500(rpm)에 대한 속도 응답 특성을 나타내었다. 그림 5에서도 퍼지제어기가 PI 제어기보다 더 나은 제어 성능을 보였다.

표 1 Tabu 탐색법의 시뮬레이션 계수

적용방법	이웃해의 수	Tabu List 수	반복횟수
Tabu 탐색법	50	5	50

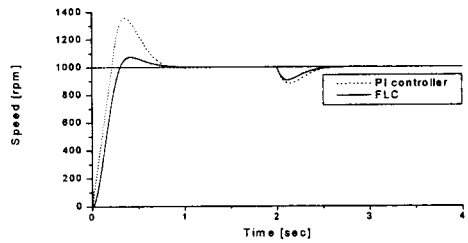


그림 4 1000[rpm]일 때 속도응답 특성

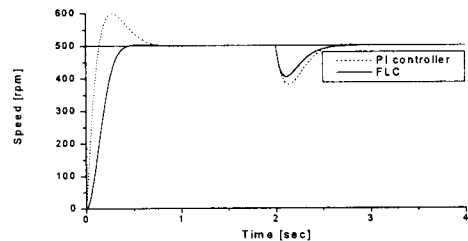


그림 5 500[rpm]일 때 속도응답 특성

#### 5. 결론

본 논문에서는 경험적인 탐색알고리즘인 Tabu 탐색법을 이용하여 AC 서보 모터의 속도제어용 퍼지제어기를 설계하였다. Tabu 탐색법을 이용하여 퍼지제어기의 입·출력 이득을 튜닝하였다. 제안한 방법의 유용성을 입증하기 위해 AC 서보 모터의 속도제어에 적용하여 컴퓨터 시뮬레이션을 수행하였다. 컴퓨터 시뮬레이션 결과, 다양한 속도에 대해서 제안한 퍼지제어기가 PI 제어기보다 좋은 제어성능을 나타냄을 알 수 있었다.

#### [참고 문헌]

- [1] F. Glover, "Tabu Search-Part I," ORSA Journal on Computing Vol. 1, No. 3, pp. 190-206, 1989.
- [2] F. Glover and M Laguna, "Tabu Search" pp. 111-116
- [3] Fred Glover, Manuel Laguna, "Tabu Search", Kluwer Academic Publishers
- [4] Emmanuel G. Collins, Jr. and Chun Fan, Richard Millett, "Automated PI Tuning for a Weigh Belt Feeder via unfalsified Control", Proceedings of the 38th IEEE Conference on Decision and Control, Vol. 1, pp. 785-790, 1999
- [5] Dragan Kukulj, Filip Kulic and Emil Levi, "Artificial Intelligence Based Gain Scheduling of PI Speed Controller in DC Motor Drives", Proceedings of the IEEE International Symposium on Industrial Electronics, Vol. 1, pp. 425-429, 1999
- [6] Fan JC, and Kobayashi T, "A Simple Adaptive PI Controller for Linear Systems with Constant Disturbances", IEEE Transactions on Automatic Control, Vol. 43, No.5, pp. 733-736, May, 1998