

---

# 타부 탐색을 이용한 센서리스 벡터 제어

이양우\* · 박경훈\*

Sensorless Vector Control Using Tabu Search Algorithm

Yang-Woo Lee\* · Kyung-Hun Park\*

---

이 논문은 2006년도 동의대학교 교내 연구비를 지원받았음

---

## 요 약

최근 효율적인 유도전동기 운전은 벡터제어에 의해 고성능 산업응용 분야에 쓰이고 있으며, 전동기 회전속도를 검출하기 위해 속도센서를 모터에 부착한다. 그러나 속도센서가 있으므로 나타나는 단점인 전기적 노이즈 발생을 최소화하도록 케이블 배치를 하여야 하며, 견고성이 떨어지고, 가격이 상승하는 등 단점이 많아 속도 센서가 없는 속도센서리스 벡터제어 이론에 대한 연구가 진행되어 왔다. 본 연구에서는 타부 탐색을 이용한 유도전동기의 센서리스 벡터제어를 연구하였다. 제안된 유도전동기 센서리스 벡터제어기는 두 부분으로 구성되어 있다. 첫 번째 부분은 타부 탐색법을 이용하여 첫 번째는 속도 추정기 초기 PI 게인 파라미터를 최적화 하는 부분이며, 두 번째 부분은 속도 제어기 PI 게인 파라미터를 최적화 시키는 부분이다. 제안된 타부 탐색법은 이웃해 영역을 찾는 방법을 삼각형 랜덤 분포를 이용하여 탐색 성능을 향상 시켰다. 제안된 방법을 유도전동기 센서리스 벡터제어기에 적용하였고 성능을 시뮬레이션과 실험으로 검증 하였다. 그 결과 부하의 변동에도 안정적으로 동작하였으며 유용성을 입증하였다.

## ABSTRACT

Recently, a speed control method of induction motor by vector control theory is applied to highly efficient industrial field. The speed sensors attached to motor are used for detection of rotating speed. In the case using speed sensor, the installation of cable for minimization of electric noise, weaken maintenance, increase of price are demerit. Therefore the study of speed sensorless vector control theory performed activity. The design of sensorless vector controller for induction motor using tabu search is studied. The proposed sensorless vector control for Induction Motor is composed of two parts. The first part is for optimizing the speed estimation with initial PI parameters. The second part is for optimizing the speed control with initial PI parameters using tabu search. Proposed tabu search is improved by neighbor solution creation using Triangular random distribution. In order to show the usefulness of the proposed method, we apply the proposed controller to the sensorless speed control of an actual AC induction Motor System. The performance of this approach is verified through simulation and the experiment.

## 키워드

타부 서치, 벡터 제어, 센서리스 속도 제어

## Key word

Tabu Search, Vector Control, Sensorless Speed Control

## I. 서 론

최근 유도전동기의 가변속 제어기의 구성과 그 응용에 대한 연구가 많이 이루어지고 있다. 제어 속도 정밀도가 요구될 때 고성능 센서에 의한 전동기 속도측정이 필요하다. 예를 들어 광학 엔코더는 속도 측정기로 많이 이용되고 있지만 기계적인 면이나 경제적인 면에서 여러 가지의 단점을 가지고 있기 때문에 산업현장에서는 적용하기가 어려운 실정이다. 따라서 현재 연구들은 가격과 신뢰성을 높이기 위해 전동기 로터 속도 측정이 필요 없는 센서리스 백터제어에 관한 것으로 집중되어 있다. 그 성능 면에서 고속영역에서는 비교적 만족스런 결과를 얻을 수 있지만 저속영역에서는 성능이 매우 떨어진다[1]. 이를 해결하기 위한 최근의 연구 중에서 대부분의 방법은 회전자속과 백터제어에서의 파라미터 튜닝이다[2][3]. 이러한 방법의 대부분은 PI제어기가 주로 사용된다. PI 제어기는 쉽게 수정이 가능하고, 원하는 응답으로 접근하기가 용이하다[6], 그러나 고성능 제어를 원할 때, 즉 시스템 상태가 변화해도 안정한 제어를 원할 때에 보통의 PI제어기로는 원하는 만큼의 제어 성능을 얻을 수 없다. 왜냐하면 PI 제어기 파라미터는 주어진 제어 조건에 맞게 고정되어 있기 때문이다. 시스템의 파라미터가 변화 되었을 때, PI제어기를 사용하는 시스템의 응답은 쉽게 영향을 받는다. 따라서 시스템 파라미터 변화에 영향이 없는 원하는 동적 응답을 보장하는 적응형 속도 제어기의 개발이 필요하다.

이러한 문제점을 해결하기 위하여 최신제어인 상태 변수 제어[2], 최적제어 및 적응제어[3] 등의 다양한 현대제어기법을 이용하여 매우 양호한 성능의 드라이브를 개발하였다. 적응제어는 부하토크 및 부하관성과 같은 광범위한 동작변화에도 양호한 속도응답을 얻을 수 있는 우수한 기법으로 평가받았다[4]. MRAC(Model Reference Adaptive Control), SMC(Sliding Mode Control), VSC(Variable Structure Control), EKF(Extended Kalman Filter) 및 STC (Self Tuning Control)[4] 등의 다양한 적응제어 기법이 제시되었다. 이러한 종래의 적응제어는 수학적 모델링에 기초하고 계산시간이 많이 소요되는 알고리즘이기 때문에 매우 복잡하였다. 이러한 단점을 개선하기 위한 방법으로 인공지능제어 기법인 퍼지제어, 신경회로망[5] 및 GA(Genetic Algorithm)가 유도전동기

시스템에 많이 적용되고 있는 추세지만 퍼지제어의 적용규칙 및 신경회로망의 학습시간, GA의 이웃해의 생성규칙 등에 따라 시스템 성능이 다르게 나타나는 단점이 있다. 본 논문에서는 이러한 문제를 해결하기 위해, 지능제어 방법 중 구현방법이 간단하고 파라미터 최적화에 유용한 타부탐색 알고리즘을 응용하였다[7]. 즉 기존 PI 제어기에 타부탐색 알고리즘을 이용한 PI 계인 튜닝에 사용하였다, 본 방식은 전역 최적능력 뿐 아니라 강인성도 가지고 있어, 기존 방식의 약점을 극복하여, 산업현장에서 널리 사용될 수 있을 것으로 기대된다. 본 논문의 유용성을 시뮬레이션 및 실험을 통하여 입증하였다.

## II. 유도 전동기 회전자 속도예측

### 2-1. 유도 전동기 수학적 모델링

회전자 자속기준에 의한 유도전동기의 방정식은 식 (1)에서 식 (5)까지로 표현된다.

$$v_d = R_s i_d + L_s^* \frac{di_d}{dt} + (L_s' - L_s^*) \frac{di_m}{dt} - \omega_{dq} L_s^* i_q \quad (1)$$

$$v_q = R_s i_q + L_s^* \frac{di_q}{dt} + (L_s' - L_s^*) \omega_{dq} i_m + \omega_d L_s^* i_d \quad (2)$$

$$\frac{L_r'}{R_r} \frac{di_m}{dt} + i_m = i_d \quad (3)$$

$$\omega_{dq} = \omega + \frac{R_r i_q}{L_r i_m} \quad (4)$$

$$\tau_e = (L_s^* - L_s') i_m i_q \quad (5)$$

식 (1)과 식 (2)는 고정자 전압의  $d$ -축과  $q$ -축 성분을 나타내고 식 (4)는 회전자 속도를 나타내며 나머지 식 (5)는 전기적인 힘인 토크를 나타낸다.

여기서 사용된 변수들은 다음과 같다.

$i_d, i_q$	$d$ -축과 $q$ -축 전류
$i_m$	상호 유도 전류
$v_d, v_q$	$d$ -축과 $q$ -축 전압
$R_s, R_r$	고정자 회전자 저항

$L_s^*, L_s', L_r'$	인덕턴스 파라미터
$\tau_e$	전기적 토크
$\omega_{dq}$	$dq$ -축 상의 회전자속도
$\omega$	모터 속도

기본적인 센서리스 벡터 제어기의 구성은 그림 1과 같이 나타낼 수 있다.

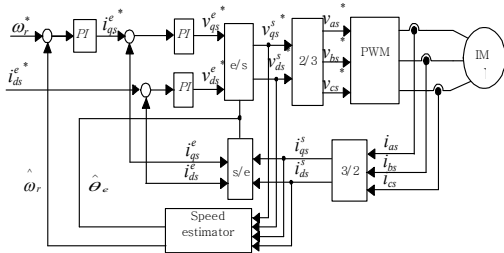


그림 1. 제어 블록도  
Fig. 1 Control Block Diagram

### 2.2 고정자 전류기반 회전자 속도예측

본 논문에서는 저속영역과 0[rpm]에서 센서리스 벡터제어의 성능을 향상시킬 수 있고 회로변수의 변화에 견실한 새로운 회전자 속도 추정방법을 제안한다. 제안한 방법에서 고정자 전류를 이용하여 회전자 속도를 구하는 방법을 이용하였다. 고정자 전류로 회전자 속도를 구하는 과정은 다음과 같다. 정지좌표계에서 회전자 자속은 다음과 같이 나타낼 수 있다.

$$\lambda_{qr} = \frac{L_r}{L_m} \left[ \int (v_{qs} - R_s i_{qs}) dt - \sigma i_{qs} \right] \quad (7)$$

$$\lambda_{dr} = \frac{L_r}{L_m} \left[ \int (v_{ds} - R_s i_{ds}) dt - \sigma i_{ds} \right] \quad (8)$$

그리고 유도전동기의 방정식에서 고정자전류를 회전자 자속과 속도를 이용하여 정리하면 식 (9)와 식 (10)과 같이 나타낼 수 있다.

$$i_{qs} = \frac{1}{L_m} (\lambda_{qr} - \omega_r T_r \lambda_{dr}) + \frac{T_r}{L_m} \rho \lambda_{qr} \quad (9)$$

$$i_{ds} = \frac{1}{L_m} (\lambda_{dr} + \omega_r T_r \lambda_{qr}) + \frac{T_r}{L_m} \rho \lambda_{dr} \quad (10)$$

위의 식에서 구한 회전자 자속과 회전자 속도 추정치  $\hat{\omega}_r$ 을 이용하여 정리하면 식 (11)과 식 (12)로 표현되어진다.

$$\hat{i}_{qs} = \frac{1}{L_m} (\lambda_{qr} - \hat{\omega}_r T_r \lambda_{dr}) + \frac{T_r}{L_m} \rho \lambda_{qr} \quad (11)$$

$$\hat{i}_{ds} = \frac{1}{L_m} (\lambda_{dr} + \hat{\omega}_r T_r \lambda_{qr}) + \frac{T_r}{L_m} \rho \lambda_{dr} \quad (12)$$

실제 고정자 전류와 추정된 고정자 전류 사이의 관계를 통해 고정자 전류의 차이를 나타내면 식 (13)과 식 (14)로 나타내어진다.

$$\hat{i}_{qs} - i_{qs} = \frac{T_r}{L_m} \lambda_{dr} (\omega_r - \hat{\omega}_r) \quad (13)$$

$$i_{ds} - \hat{i}_{ds} = \frac{T_r}{L_m} \lambda_{qr} (\omega_r - \hat{\omega}_r) \quad (14)$$

고정자 전류의 차이가 회전자 속도추정오차와 회전자 자속의 함수로 나타나므로 고정자 전류의 차이를 자속으로 나누면 회전자 속도추정오차로 표현된다. 그러나 순시적으로 변하는 회전자 자속 값 대신 일정하게 유지되는 회전자 자속의 크기를 이용하기 위해 식 (13)과 식 (14)에  $\lambda_{dr}$  과  $\lambda_{qr}$  을 각각 곱하면 식 (15)와 같다.

$$\begin{aligned} & (\hat{i}_{qs} - i_{qs}) \lambda_{dr} + (i_{ds} - \hat{i}_{ds}) \lambda_{qr} \\ &= \frac{T_r}{L_m} (\omega_r - \hat{\omega}_r) [(\lambda_{qr})^2 + (\lambda_{dr})^2] \end{aligned} \quad (15)$$

위 식을 회전자 속도의 오차에 대해 정리하면

$$\omega_r - \hat{\omega}_r = n [(i_{qs} - \hat{i}_{qs}) \lambda_{dr} + (i_{ds} - \hat{i}_{ds}) \lambda_{qr}] \quad (16)$$

과 같고

$$n = \frac{L_m}{T_r} \frac{1}{(\lambda_{qr})^2 + (\lambda_{dr})^2} \frac{R_r}{(\lambda_{qr})^2 + (\lambda_{dr})^2} \quad (17)$$

이다.

식 (17)을 이용하면 고정자 전류와 회전자 자속으로부터 속도추정 오차를 계산할 수 있고 속도추정오차를 적절한 비례이득 및 적분이득을 가지는 PI제어기에 인가하면 PI제어기는 입력 값인 속도 추정 오차를 영으로 만드는 방향으로 출력 값인 속도추정치를 제어하게 된다. 따라서 PI제어기의 출력 값 즉 속도추정치는 다음과 같이 실제회전자 속도를 추정할 수 있다.

$$\hat{\omega}_r = K_p e + K_i \int e dt \quad (18)$$

여기서

$$e = [(i_{qs}^{\wedge} - i_{qs})i_{dr} + (i_{ds} - i_{ds}^{\wedge})i_{dr}]/n \text{ 이다.}$$

### III. 타부 탐색

타부 탐색은 최적화문제를 해결하는 최적화 기법의 하나로 인간의 기억과정을 이용해 복잡한 해 영역에서 최적해를 얻기 위해 탐색을 유도하는 휴리스틱적인 기술이다[6]. 이것은 1970년대 후반에 글러버(Glover)에 의해서 소개되었고 유전자알고리즘(GA)이나 시뮬레이티드 어닐링(SA)등과 같은 전역적 탐색기법에 비해 대상 문제에 관한 지식을 활용하기에 유리한 탐색기법이다 [7][8]. 이런 타부탐색은 값이나 반복을 방지하기 위해 탐색영역에서 이전의 탐색했던 영역을 기억한다. 이 탐색 메커니즘은 목적함수 최소화를 위한 효과적인 해 개선을 통하여 최적해에 가까이 갈 목적으로 수행된다. 간단한 수행과정을 나타내면 그림 2와 같다.

- Step 1 :  $S \in X \rightarrow$  초기해  $S_0$ 를 선택한다.
- Step 2 :  $N(i)$ 내에서 이웃해  $N(S)$ 를 생성 한다.
- Step 3 :  $N(S)$ 에서 가장 좋은 해  $S^*$ 를 구한다.  
(i.e. such that  $f(S^*) \leq f(S) \forall S \in N(S)$ )
- Step 4 : 만약  $f(S^*) \geq f(S)$  이면 멈추고 탐색을 정지한다.  
그렇지 않으면  $S=S^*$  종료 조건이 될 때까지 Step 2

그림 2. 기본적인 타부 탐색  
Fig. 2 Basic Tabu search method.

전체 영역  $N(i)$  내에서 초기해  $S_0$ 를 설정한 후  $S$ 의 이웃해  $N(S)$ 를 생성한다. 생성된 이웃해를 평가함수  $f(S)$ 로 평가를 하여 최적의 해  $S^*$ 통해 최적해  $S^*$ 를 선정한다.  $S^*$ 가 원하는 최적해이면 종료하고 아니면  $S^*$ 를 타부리스트(Tabu\_List)에 놓고  $S^*$ 를 사용하여 위의 과정을 반복한다.

일반적으로 타부탐색의 성능은  $N(S)$ 의 정의 및 크기나 탐색 전략에 따라 달라진다.

그림 3은 이웃해를 생성하는 일반적인 방법을 나타낸 것이다.

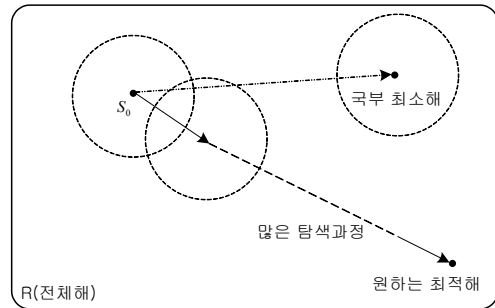


그림 3. 이웃해를 생성하는 일반적인 방법  
Fig. 3 general method for generating neighborhood

그림 3에서 알 수 있듯이 최적해가  $S_0$ 에서 멀리 존재할 때 최적해에 접근하기 위하여 많은 탐색과정을 거쳐야 된다. 또한 접근 방향이 최적해에 가까이 가지 못한다면 국부 최소해에 빠질 가능성이 크다. 이러한 문제점은 이웃해의 분포를 좀더 확장하여 해결할 필요가 있다. 이웃해를 확장할 때 기준해  $S_0$  근처의 이웃해가 최적해가 될 가능성이 높으므로 큰 밀도로 해가 존재하고 멀어질수록 최적해가 될 가능성이 적으므로 해가 존재할 가능성이 적다.

본 논문에서는 연산의 간략화를 위해 그림 4와 같이 삼각함수 분포(Triangular Random Distribution)로 단순화하여 이웃해를 선정하도록 하였다.

$S_0$ 는 기준해, 전체해 영역을  $R$  이웃해 존재구간을  $d$ 라고 하고  $(0 \leq R \leq 1)$ 라고 한다.

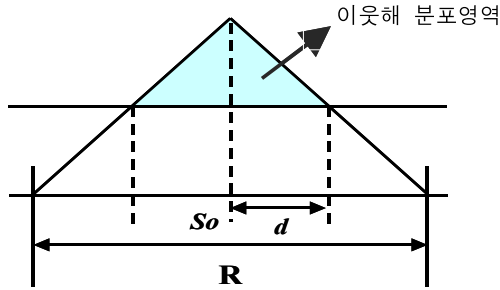


그림 4. 삼각함수 분포  
Fig. 4 Triangular Random Distribution

타부 탐색을 사용한 제안한 알고리즘의 수행은 다음과 같다.

1) 초기해 생성

초기해  $S_0$ 는 랜덤하게 생성한다.

$$S_0 = \{init1, init2, init3\} \quad (19)$$

2) 이웃해 생성

생성된 초기해 또는 현재의 최적해를 이용하여 가우시안 랜덤 분포를 이용하여 해를 생성한다.

해 생성시 안정한 범위 내에 존재하는 해를 찾는다. 그리고 생성된 해를 타부리스트와 비교하여 타부리스트에 존재하는 해를 제외시킨다.

3) 성능평가

위의 과정에서 생성된 이웃해  $N(S)$ 들로부터 각각의 성능지수(Performance Index)인 식 (20)과 같은  $O_j$  를 통해 새로운 최적의 해를 찾는다.

$$O_j = \text{Minimize} \sum_{k=1}^n E(k) \quad (20)$$

성능지수가 평가되는 동안  $N(S)$ 의 영역은 증가되거나 감소된다.

IV. 시뮬레이션 및 실험

4.1 시뮬레이션

본 절에서는 그림 5와 같이 수정형 타부탐색 알고리즘을 사용한 제안된 제어기의 성능을 시뮬레이션을 통해 나타내었다.

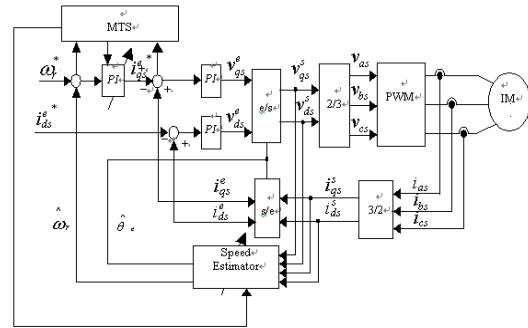


그림 5. 제안하는 제어 블록도  
Fig. 5 Proposed Control Block Diagram

본 시뮬레이션에 사용된 2.2KW 유도전동기의 파라미터는 표 1과 같다.

표 1. 시뮬레이션 파라미터  
Table. 1 Simulation parameters

고정자저항	$R_s$	0.921
회전자 저항	$R_r$	0.583
고정자 인덕턴스	$L_s$	67.1mH
회전자 인덕턴스	$L_r$	67.1mH
상호 인덕턴스	$L_m$	65 mH

위의 파라미터를 사용하여 수행된 시뮬레이션을 통해 지글러-니콜스(Ziegler-Nichols)에 의한 일반적인 PI 제어기와 제안된 PI제어기의 성능을 비교하였다. 제안된 타부탐색에 의한 PI제어기의 속도응답 특성을 그림 6에 나타내었다.

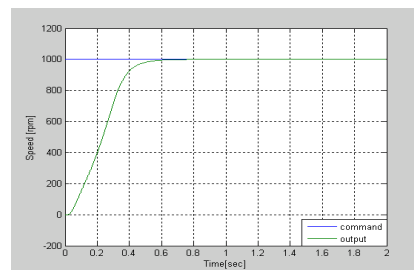
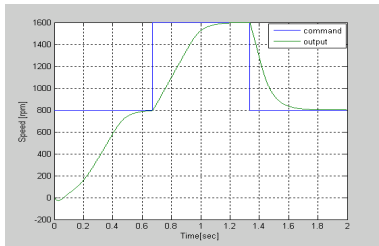
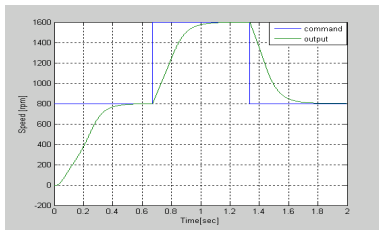


그림 6. 속도 계단 응답 특성(1000rpm)  
Fig. 6 Speed step response (1000rpm)

그림 7은 동일 조건 하에서 속도 지령치에 의한 응답을 PI 제어기와 제안된 PI제어기를 비교하여 나타낸 것이다.

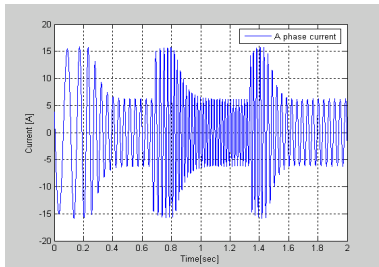


(a)

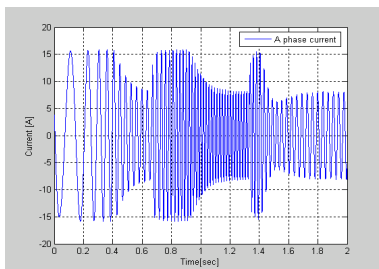


(b)

그림 7. 속도별 계단 응답  
(800rpm → 1600rpm → 800rpm)  
(a) 제안한 PI제어기 (b)기존 PI제어기  
Fig. 7 Step Responses of Motor Speed  
(800rpm → 1600rpm → 800rpm)  
(a) Proposed PI Control (b) General PI Control



(a)

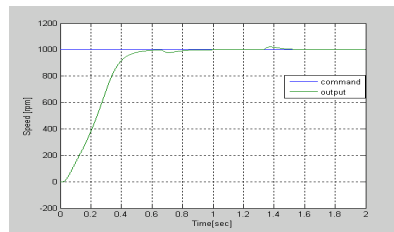


(b)

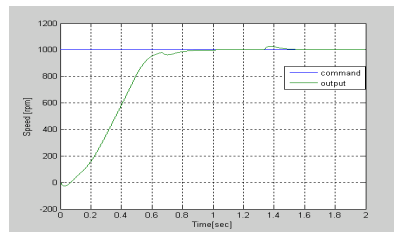
그림 8. 속도 계단 응답 시 출력전류  
(a) 제안한 PI제어기 (b) 기존 PI제어기  
Fig. 8 Output load Current of Motor Speed  
(a) Proposed PI Control (b) General PI Control

그림 8은 속도지령을 무부하 상태에서 800rpm → 1600rpm → 800rpm으로 지령속도를 변화 시켰을 때 응답 특성이다. 그림 9는 부하를 0% → 100% → 0%로 변화 시켰을 속도 변동 특성을 나타내었다. 이때 속도 지령 1000rpm 일 때이다. 시뮬레이션 결과 제안한 제어기 응답 성능이 빠른 것을 알 수 있다. 그림 9로부터 알 수 있는 바와 같이 본 논문에서 제안한 제어기가 부하변동대비 속도 안정도가 좋은 것으로 나타났다.

종합적인 시뮬레이션 결과로부터 수정형 타부담색 알고리즘을 적용한 제어기 성능이 우수함을 알 수 있었다.



(a)



(b)

그림 9. 부하변동에 대한 속도 제어 특성  
(a) 제안한 PI제어기 (b) 기존 PI제어기  
Fig. 9 Speed regulation against load  
(a) Proposed PI Control (b) General PI Control

#### 4.2 실험

제안한 방법의 센서리스 벡터제어 시스템을 시뮬레이션과 동일한 조건으로 실험을 수행하였다. 속도추정기의 입력은 상전류와 기준 상전압이다. 상전류는 전류 센서를 이용하여 직접 측정하였고 상전압은 전압센서를 설치하는 문제 때문에 기준전압을 실측전압 대신 사용하였다. 대신 입력전압과 기준전압의 측정오차를 줄이기 위하여 인버터의 데드타임보상과 직류전압변동도 측정하여 실제 전압과 거의 일치토록 하였다. 제어관련 대부분의 연산은 프리스케일(Free Scale)사의 DSP(56F807)

를 사용하였으며 실제 속도관측을 위해 1024(p/r) 펄스 엔코더를 사용하였고 부하는 그림 10과 같이 벡터제어 인버터와 2.2kW급 전동기를 사용하였다.



그림 10. 부하 시험장치  
Fig. 10 Load Test Equipment

그림 11과 그림12는 -0.5Hz(15rpm), 30Hz(900rpm)운전 시 부하 변동에 따라 추정 속도 변동을 나타내고 있다. 따라서 그림 12를 통해서 낮은 주파수의 속도에서도 부하 변동 시 5% 이내의 빠른 추종 특성을 볼 수 있었으며 속도 오차도 최대 2% 이내의 안정한 특성을 나타 내었다.

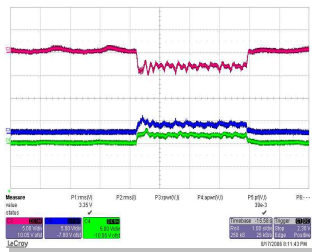


그림 11. 부하 변동에 따른 속도 제어 응답(-15rpm)  
(0%→100%→0%)  
Fig 11. Speed regulation against load variation  
(0%→100%→0%)



그림 12. 부하 변동에 따른 속도 제어 응답(900rpm)  
(0%→100%→0%)  
Fig 12. Speed regulation against load variation  
(0%→100%→0%)

그림 13은 100% 부하 인가 시 -30Hz(-900rpm) 에서 30Hz(900rpm)으로 정역운전 상태로 영속 근처의 속도 추정 능력이 뛰어난을 알 수 있다.

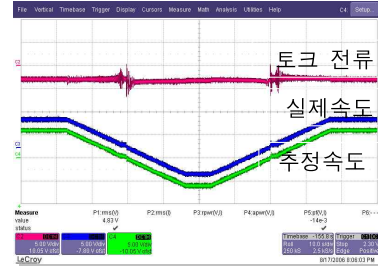


그림 13. 100% 부하 인가 시 정역 운전  
(-900rpm → 0rpm → 900rpm)  
Fig. 13 Forward and reverse operation in 100%  
load condition.(-900rpm→0rpm→900rpm)

그림 14는 토크 제어 특성을 알아보기 위하여 기준 토크 대비 전동기 발생 토크를 측정된 결과이다. 측정 결과 기준 토크 대비 오차가 2% 이내로 우수하게 측정되었다.

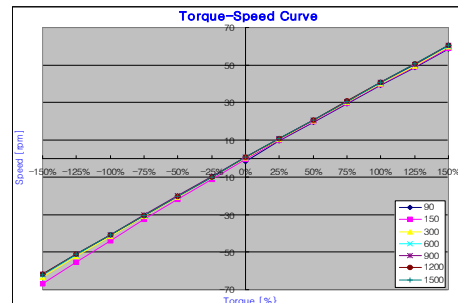


그림 14. 토크 선형성  
Fig. 14 Torque linearity test

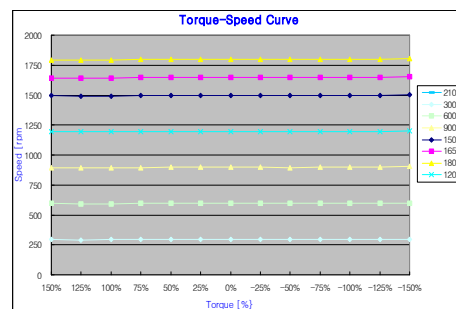


그림 15. 속도-토크 특성곡선  
Fig 15. Speed-torque curve

그림 15는 부하별 속도 제어 성능을 나타낸 것으로 제어 특성이 우수함을 알 수 있었다.

### V. 결 론

본 논문에서는 타부탐색 알고리즘을 이용하여 센서리스 벡터제어기의 성능을 향상시킬 수 있는 제어기를 제안하였다. 2.2kW급 유도전동기 센서리스 벡터제어를 구현하여 모의실험 및 실험으로 타당성을 확인하였다. 제안된 타부 탐색을 이용한 센서리스 벡터제어기는 구현이 간단하며, 오차가 작고, 모터 파라미터 변동 발생 시에도 강인한 특성이 있다. 모터 속도 추정기 및 속도제어기 계인 튜닝관련 초당 학습하는 시간이 적고 연산이 간단하다. 짧은 시간 안에 매우 빨리 속도 지령 값에 도달하며 약간의 부하 변동 시에도 속도변동 없이 강인하며 저가로 쉽게 구현 가능하며 프로그램 양도 줄일 수 있다. 또한 저속 및 고속의 전 영역에서 우수한 추종능력을 나타낸다. 상기 장점을 지닌 타부탐색을 이용한 제어기를 적용하면 산업현장에 적용이 간편하고 제어 성능 면에서도 유용하리라 본다.

### 참고문헌

[1] C. Schauder "Adaptive Speed identification for vector control of induction motor without rotational transducers" *IEEE Trans. Ind. Appl.* vol. 28 no. 5 pp. 1054-1061, September/October 1992.

[2] H. Kubota and K. Matsuse "Speed sensorless field-oriented control of induction motor with rotor resistance adaptation" *IEEE Trans. Ind. Appl.* vol. 30 no. 5 pp. 1219-1224, September/October 1994.

[3] F. Z. Peng and T. Fukao "Robust speed identification for speed-sensorless vector control of induction motors" *IEEE Trans. Ind. Appl.* vol. 30 no. 5 pp. 1234-1240 Sep./Oct. 1994.

[4] K. J. Astron and B. Wittenmark, "Adaptive control," Addison-Wesley, 1989.

[5] Farzan Rashidi, "Sensorless Speed Control of Induction Motor Derives Using a Robust and Adaptive

Neuro-Fuzzy Based Intelligent Controller", IEEE International Conference an Industrial Technology, pp.617-627, 2004.

[6] L. H. Keel, J. I. Rego, and S. P. Bhattacharyya, "A New Approach to Digital PID Controller Design", IEEE Tran. ON AUTOMATIC CONTROL, vol. 48, pp.687-692, 2003.

[7] F. Glover, "Future Paths for Integer Programming and Link to Artificial Intelligence," *Computers and Operations Research*, 12:533-549, 1986.

[8] D. Karaboga and A. Kalili, "Tuning PID Controller Using Tabu Search Algorithm", IEEE International Conference on Volume: 1, 14-17 Oct pp.134-136, 1996.

### 저자소개



이양우(Yang-Woo Lee)

1995년 부산대학교 공학박사  
1987년~현재 동의대학교 교수  
1997~1998년 휴스턴대 교환교수

※관심분야: 계측제어, 지능제어, 자동화설계



박경훈(Kyung-Hun Park)

1997년 동의대학교 공학석사  
2001년 동의대학교 박사과정 수료  
2001 ~ 현재 LS산전 자동화연구소  
선임

※관심분야: 제어기설계, 전력변환장치