Drive of Induction Motors Using a Pseudo-On-Line Fuzzy-PID Controller Based on Genetic Algorithm

Taechon Ahn, Yangwon Kwon, and Haksoo Kang

Abstract: This paper proposes a novel method with pseudo-on-line scheme using the optimized look-up table based on the genetic algorithm which does not use the gradient and finds the global optimum of an un-constraint optimization problem. The technique is a pseudo-on-line method that optimally estimates the parameters of fuzzy PID(FPID) controller for systems with non-linearity, using the genetic algorithm. The proposed controller(GFPID) with the auto-tuning function is applied to the on-line and real-time control of speed at 3-phase induction motor, and its computer simulation is carried out. Simulation results show that the proposed method is more excellent than conventional FPID and PID controllers.

Keywords: induction motor, FPID, genetic algorithm, pseudo-on-line, GFPID

I. Introduction

Electric-power vehicles use a variety of motor drive systems. The recent introduction of purpose-built electric vehicles indicates that both AC induction and BLDC motors are gaining in popularity for traction motor applications[1].

The induction motor control problem has been widely studied with the objectives of obtaining better results in terms of stability, robustness to parameters variation and disturbances rejection. The voltage or current and frequency are the basic control variables of the induction motor. Many algorithms have been employed to improve the performance of the induction motor control[2].

The 3-phase induction motor is a representative plant, and the conventional PID controllers are used extensively in its control[3]. They are inexpensive and very effective for simple linear systems. Use of these conventional controllers is often adequate when the non-linearities of process are mild and plant operations are constrained to small region at a nominal steady-state. Model-based nonlinear control techniques can be used when high performance is required over a broader range of operating conditions. However, the approaches require an accurate model of the process. A simpler alternative, although with some loss in performance, is to use linear controllers with gain scheduling. The design of discrete-time FPID controllers in various combinations results in a new fuzzy version of the result of the conventional PID controllers[4]. These controllers have the same linear structure as the conventional PID controllers in the proportional, integral and derivative parts, but have non-constant gains, namely, the proportional, integral, and derivative gains are nonlinear functions of the input signals. The FPID controllers thus preserve the simple linear structure of the conventional controllers, yet enhance the self-tuning control capability for non-linearity[5][6].

In this paper, a pseudo-on-line method(GFPID) is proposed that auto-tunes the parameters of controller by the genetic algorithm[7] and the fuzzy clustering technique[4], for the improvement and optimization of systems. This method is applied to FPID controller of the drive system of induction motor. The proposed GFPID controller with the auto-tuning function executes the speed control of the system. Authors divide the region of errors that have influence on the system parameters, into several error levels and then make each level the optimized look-up table using the genetic algorithm. This makes on-line and real-time control of the drive system possible. Computer simulations would show that the proposed controller has high performance better than conventional FPID and PID controllers.

II. Induction motor modeling

As the stator or rotor is assumed to have symmetrical air gap, it is possible to express its voltage equations of the three-phase induction motor in stationary coordinates *as, bs* and *cs*, as follows [1][8]:

$$\boldsymbol{v}_{abcs} = \boldsymbol{R}_s \boldsymbol{i}_{abcs} + \frac{d\boldsymbol{\lambda}_{abcs}}{dt} \tag{1}$$

$$\boldsymbol{v}_{abcr} = \boldsymbol{R}_r \boldsymbol{i}_{abcr} + \frac{d\boldsymbol{\lambda}_{abcr}}{dt} \tag{2}$$

where $\boldsymbol{v}_{abc*}$, $\boldsymbol{i}_{abc*}$ and $\boldsymbol{\lambda}_{abc*}$ are instantaneous voltage, current and flux linkage vectors of the rotor and stator, respectively, in the stationary frame.

The $d-q$ reference frames are usually selected on the basis of convenience or compatibility with the representations of other network components. Those of the induction machine in the stationary reference frame can be obtained by setting.

The $d-q$ transformation matrix S_{dq} is given by:

$$S_{dq} = \frac{2}{3}\begin{bmatrix} \cos(\omega t) & \cos(\omega t - \frac{2\pi}{3}) & \cos(\omega t + \frac{2\pi}{3}) \\ \sin(\omega t) & \sin(\omega t - \frac{2\pi}{3}) & \sin(\omega t + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}. \tag{3}$$

The transformation equation from $a-b-c$ to $d-q$ reference frame is follows:

Manuscript received: Feb. 12, 2000., Accepted: June. 14, 2000.
Taechon Ahn, Yangwon Kwon: School of Electrical and Computer Engineering, Wonkwang University
Haksoo Kang: Dept. of Electrical, Jeonju Technical College
※ This paper was supported by Wonkwang University in 1999. The first author gives thanks for the financial support.

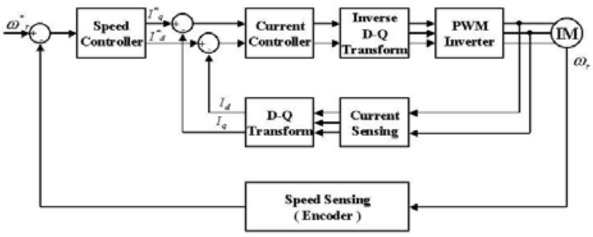

Fig. 1. The scheme of control system considered.

$$f_{dq0} = S_{dq} f_{abc} \tag{4}$$

where f_{dq0} is $[f_q f_d f_0]^T$.

The stater $d-q$ voltage equation in a synchronously rotating frame can be written in terms of component d and q voltages as follows:

$$v_{qs} = R_s i_{qs} + \frac{d\lambda_{qs}}{dt} + \omega_e \lambda_{ds} \tag{5}$$

$$v_{ds} = R_s i_{ds} + \frac{d\lambda_{ds}}{dt} - \omega_e \lambda_{qs} \tag{6}$$

where i_{qs} and i_{ds} are the stator *d- and q*-axes currents, λ_{qs} and λ_{ds} are the stator *q*- and *d*-axis flux linkages, respectively.

The rotor $d-q$ voltage equation in a synchronously rotating frame can be also written in terms of component d and q voltages as follows:

$$v_{qr} = R_r i_{qr} + \frac{d\lambda_{qr}}{dt} + (\omega_e - \omega_r)\lambda_{dr} \tag{7}$$

$$v_{dr} = R_r i_{dr} + \frac{d\lambda_{dr}}{dt} + (\omega_e - \omega_r)\lambda_{qr} \tag{8}$$

where ω_e is the stator angular frequency, ω_r is the rotor angular frequency and $\omega_{sl} = (\omega_e - \omega_r)$ is the slip angular frequency. i_{qr} and i_{dr} are the rotor *d- and q*-axes currents, λ_{qr} and λ_{dr} are the rotor *q*- and *d*-axis flux linkages, respectively.

The stator and rotor flux linkage expressions in terms of the currents can be written compactly as follows [8]:

$$\lambda_{qs} = L_{ls} i_{qs} + L_m(i_{qs} + i_{qr}) \quad \lambda_{ds} = L_{ls} i_{ds} + L_m(i_{ds} + i_{dr})$$
$$\lambda_{qr} = L_{lr} i_{qr} + L_m(i_{qs} + i_{qr}) \quad \lambda_{dr} = L_{lr} i_{dr} + L_m(i_{ds} + i_{dr}) \tag{9}$$

where, L_{ls} is stator leakage inductance, L_{lr} is rotor leakage inductance and L_m is mutual inductance.

The induction motor state space modeling, which describes its electric behavior in $d-q$ synchronously rotating reference frame, is written below in a standard notation.

$$\begin{bmatrix} v_{qs} \\ v_{ds} \\ v_{qr} \\ v_{dr} \end{bmatrix} = \begin{bmatrix} R_s + \frac{d}{dt}L_s & \omega_e L_s & \frac{d}{dt}L_m & \omega_e L_m \\ -\omega_e L_s & R_s + \frac{d}{dt}L_s & -\omega_e L_m & \frac{d}{dt}L_m \\ R\frac{d}{dt}L_m & \omega_{sl} L_m & R_r + \frac{d}{dt}L_r & \omega_{sl} L_r \\ -\omega_{sl} L_m & \frac{d}{dt}L_m & -\omega_{sl} L_r & R_r + \frac{d}{dt}L_r \end{bmatrix} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{qr} \\ i_{dr} \end{bmatrix}. \tag{10}$$

The expression for the electromagnetic torque in terms of current as follows:

$$T_e = \left(\frac{3}{2}\right)\left(\frac{P}{2}\right) L_m (i_{qs} i_{dr} - i_{ds} i_{qr}) \tag{11}$$

The vector control of the induction motor[3] is a very accepted method when high performance of the system response is required. It is based on the decoupling of the magnetizing and torque producing components of the stator current. Under this condition, the q-axis component of the rotor flux is set to zero, while the d-axis reaches the nominal value of the magnetizing flux. The torque equation becomes as follows:

$$T_e = \frac{3}{2}\frac{P}{2}\frac{L_m}{L_r}(i_{qs}\lambda_{dr} - i_{ds}\lambda_{qr}) \tag{12}$$

where L_r is $L_{lr} + L_m$ and P is the number of poles.

As the λ_{qr} is set to zero, the torque is as follows:

$$T_e = \frac{3}{2}\frac{P}{2}\frac{L_m}{L_r}(i_{qs}\lambda_{dr}). \tag{13}$$

Then the torque component current i_{qs} is as follows:

$$i_{qs} = \frac{2}{3}\frac{2}{P}\frac{L_r}{L_m}\frac{1}{\lambda_{dr}} T_e. \tag{14}$$

In this paper, the above-mentioned 3-phase induction motor is controlled using the pseudo-on-line method based on genetic algorithm. The scheme of control system considered is like Fig. 1.

III. The direct FPID control algorithm

In the design and stability analysis of the FPID controller, the determination of the control gains that gave acceptable outputs was done by manually tuning these gains until a set that gave satisfactory outputs were obtained[9].

In general, the control input u of the fuzzy controllers is decided by the proportional combination of error term and derivative term of error as eqn. (15). It corresponds to the PD of conventional PID as eqn. (16).

$$u = \sum_{i=1}^{k} (\mu(e_i) \cap \mu(de_i)) \quad \langle Fuzzy \rangle \tag{15}$$

$$u(t) = k_p e(t) + k_d \frac{de(t)}{dt} \quad \langle PD \rangle. \tag{16}$$

Because the fuzzy control input does not consider the integral term of error in the PID control, the controller didn't get high performance in the steady state. To obtain the high performance, fuzzy control system carried out separately transient control in the transient state and precise control in the steady state. However, its control strategy is more complex, when the controller is applied to induction motor. Therefore self-tuning/organizing fuzzy controllers were developed in various fields[10].

In this paper, to remove defects of the fuzzy controller, the direct FPID controllers are designed by using the conventional PD+I controller design method[11]. To obtain the increment of fuzzy control input, this method directly applies the control gains to PID control input concept. The fuzzy reasoning is executed using the eqn. (17).

$$du = k_p \cdot e + k_i \cdot ie + k_d \cdot de. \tag{17}$$

The values of e, de and ie are described as follows:

$$e_0 \le e \le e_m \tag{18}$$

$$de_0 \le de \le de_m \tag{19}$$

$$ie_0 \le ie \le ie_m \tag{20}$$

where e_m , de_m and ie_m are the maximum values of error, derivative of error and integral of error and e_o, de_0 and ie_o, the minimum values of error, derivative of error and integral of error.

The fuzzy sets of e, de and ie are described as Fig. 2:

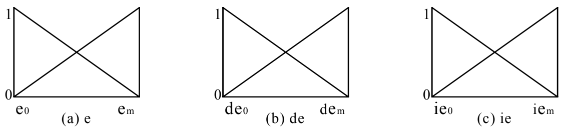

Fig. 2. Fuzzy sets of e, de, ie.

Each fuzzy rule can be described as follows by simpled fuzzy reasoning method.

Rule 1 : e_0 and de_0 and $ie_0 \Rightarrow f_1$
Rule 2 : e_0 and de_0 and $ie_m \Rightarrow f_2$
Rule 3 : e_0 and de_m and $ie_0 \Rightarrow f_3$
Rule 4 : e_0 and de_m and $ie_m \Rightarrow f_4$
Rule 5 : e_m and de_0 and $ie_0 \Rightarrow f_5$
Rule 6 : e_m and de_0 and $ie_m \Rightarrow f_6$
Rule 7 : e_m and de_m and $ie_0 \Rightarrow f_7$
Rule 8 : e_m and de_m and $ie_m \Rightarrow f_8$
Fact : e de ie

The determining process of the weighting of each rule and its consequence is drawn as Fig. 3 describe Each rule of fuzzy inference.

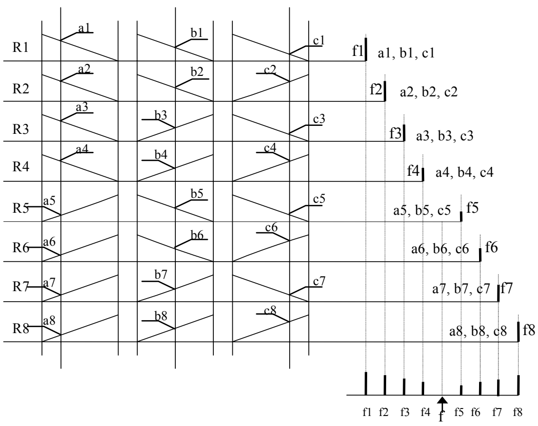

Fig. 3. Illustration of fuzzy reasoning.

As the results of fuzzy reasoning, the output of fuzzy controller is described as follows:

$$f = d\boldsymbol{u} = \frac{f_{11}+f_{22}}{h_{11}+h_{22}} = f_{11}+f_{22} \\ = k_p \cdot e + k_d \cdot de + k_i \cdot ie \tag{21}$$

where,

$$f_{11} = abc \cdot f_2 + ab(1-c) \cdot f_2 + a(1-b)c \cdot f_3 + a(1-b)(1-c) \cdot f_4$$
$$f_{22} = (1-a)bc \cdot f_5 + (1-a)b(1-c) \cdot f_6 + (1-a)(1-b)c \cdot f_7 + (1-a)(1-b)(1-c) \cdot f_8$$
$$h_{11} = abc + ab(1-c) + a(1-b)c + a(1-b)(1-c)$$
$$h_{22} = (1-a)bc + (1-a)b(1-c) + (1-a)(1-b)c + (1-a)(1-b)(1-c)$$
$$h_{11} + h_{22} = 1$$
$$a = \mu e_0(e) = \frac{e_m - e}{e_m - e_0}$$
$$b = \mu de_0(de) = \frac{de_m - de}{de_m - de_0}$$
$$c = \mu ie_0(ie) = \frac{ie_m - ie}{ie_m - ie_0}.$$

The output of fuzzy controller, that is, eqn. (21) is used as the control input of induction motor.

IV. Auto-tuning methods of FPID controller

1. Heuristic algorithm

Fuzzy controllers achieve inferred values of the control inputs using triangular membership functions. Recent literature has suggested that other forms of input membership function can be used to provide different properties for the controller. However, the triangular membership functions [Fig. 4] provide an idal means of developing control capability and so are used in general.

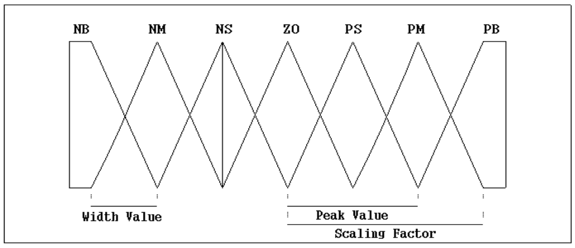

Fig. 4. Scaling factors of membership function.

The change of fuzzy control in the fuzzy look-up table have much influence on the performance of a system. Therefore, the use of estimated control rule obtains better results in terms of stability, robustness to parameters variation and disturbances rejection.

A fuzzy model[11] consists of a finite number of fuzzy implication rules. The fuzzy modelling is related to the construction of fuzzy rules based on a set of input reference command signal and output measurement. Using this input-output data set, fuzzy clustering separates this data set into several local sets so that it provides an accurate representation of the system's behavior. Fuzzy clustering method, that is, a batch-mode unsupervised classification scheme, provides a analytical way for the structure of the fuzzy model.

As an example, we consider the Fuzzy PD control of induction motor. The estimated look-up table of control rules is generally tuned as Table 1(a), using the heuristic algorithm. Table 1(b) shows the revised look-up table, when the shaded PM in the Table 1(a) changes to the shaded PB.

Fig. 5 depicts the characteristic of induction motor speed, according to the change of a control rule. The settling time and the overshoot are decreased, the rise time increased, in the Table 1(b). Therefore, according to objective function, we can choose the PM or PB and obtain the optimal look-up table through the iteration of the same process.

In the paper, define the weighted objective function to appreciate the fitness of induction motor as eqn. (22) [13].

$$\boldsymbol{F(k)} = s(k) + \frac{1}{2}\, r(k) + 10 \frac{v_p(k) - v_{ref}(k)}{v_{ref}(k)} \tag{22}$$

where $s(k)$ is settling time, $r(k)$, rising time, $v_p(k)$, maximum overshoot, and $v_{ref}(k)$, reference speed.

The minimum value of F is the optimal result. The fitness of Table 1(a) and (b) is 4.65 and 4.25, respectively. Then, the response of Table 1(b) is more excellent than (a).

Table 1. Change of tuned look-up table.

e \ de	NB	NM	NS	ZO	PS	PM	PB
NB	NB	NB	NB	NB	NM	NS	ZO
NM	NB	NB	NB	NM	NS	ZO	PS
NS	NB	NB	NM	NS	ZO	PS	PM
ZO	NB	NM	NS	ZO	PS	PM	PB
PS	NM	NS	ZO	PS	PM	PB	PB
PM	NS	ZO	PS	PM	PB	PB	PB
PB	ZO	PS	PM	PB	PB	PB	PB

(a)

e \ de	NB	NM	NS	ZO	PS	PM	PB
NB	NB	NB	NB	NB	NM	NS	ZO
NM	NB	NB	NB	NM	NS	ZO	PS
NS	NB	NB	NM	NS	ZO	PS	PM
ZO	NB	NM	NS	ZO	PS	PM	PB
PS	NM	NS	ZO	PS	PM	PB	PB
PM	NS	ZO	PS	PM	PB	PB	PB
PB	ZO	PS	PB	PB	PB	PB	PB

(b)

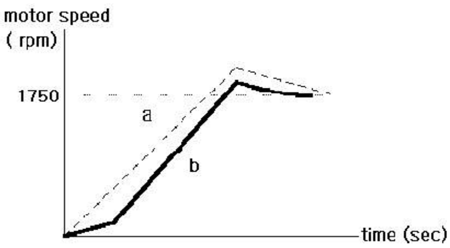

Fig. 5. Motor speed response by change of fuzzy control rule.

The heuristic method generally shows that the control system not only have a good response but also keep the stability of system. But this method need the expert with much information for induction motor.

2. Genetic algorithms

Genetic algorithms (GAs) [7][12] are directed to random search techniques, which can find the global optimal solution in complex multidimensional search spaces. GAs employ different genetic operators to manipulate individuals in a population of solution over several generations to improve their fitness, gradually. Normally, the parameters to be optimized are represented in a binary string.

To start the optimization, GAs use randomly produced initial solutions created by random number generator. This method is preferred when a priori knowledge about the problem is not available.

The flow chart of GAs [7] is shown in Fig. 6 in general. Three genetic operators are used, in order to generate and explore the neighborhood of a population and to select a new generation. These operators are reproduction, crossover and mutation. After randomly generating the initial population of N solutions, the GAs use the three genetic operators to yield N new solutions at each iteration. In the selection operation, each solution of the current population is evaluated by its fitness value obtained from a objective function. Individuals with higher fitness value are selected for survival of next generation.

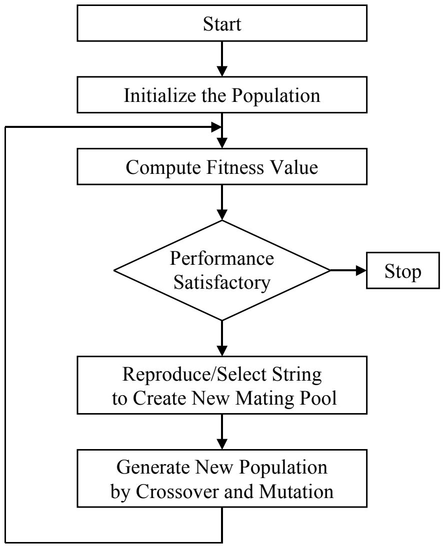

Fig. 6. Flow chart using GAs.

In this paper, to easily conduct the crossover operator, input variables are multiplied by 1000, roundoff the fractions, transformed into integer and converted to binary digital system. This integers become new input variables. Takagi's formula[13] is used as the objective function that defined by function of input variables like eqn.(23).

$$\boldsymbol{F(k)} = \sqrt{(e(k)^2 + de(k)^2 + ie(k)^2)} \tag{23}$$

where $e(k)$, $de(k)$ and $ie(k)$ is error, derivative of error and integral of error as input variables, respectively. The number of populations use 10 and the number of chromosomes gain 20 through the conversion of five decimal places to binary. Selection of genetics uses probability theory and random variable. The crossover and mutation rates also use random variable. The algorithm is repeated until a predefined result has been produced.

Through the genetic algorithm, a look-up table is made of the optimized results and it is used to the On-line system. Each table consists of 125, 17000 and 1000000 databases that is divided into 5, 30 and 100 levels for each of input variables, that is, $e(k)$, $de(k)$ and $ie(k)$. Each level is

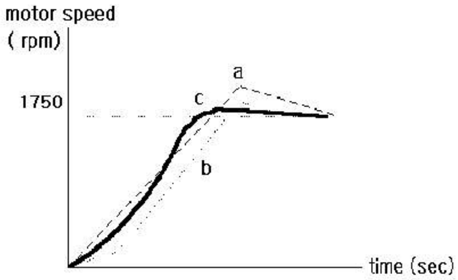

Fig. 7. Motor speed response of tuned fuzzy rule using GAs.

divided by proportion to the square of error for steady state. The scope of $e(k)$ is chosen to set the difference between reference speed and initial speed as -100% and 100% overshoot as 100%. The scope of $de(k)$ is chosen to set 1000 as 100% and -1000 as -100%, because $de(k)$ approaches to infinity. The scope of $ie(k)$ is chosen to set 1 as 100% and -1 as -100%, because $de(k)$ is limited from -1 to 1.

When the number of levels increases, the excellent result can be obtained. But increasing levels cause that the performance is getting bad, owing to large computer capability and the low access speed. Therefore it is to be suitable to select levels between 30 to 100.

Table 2 shows scaling factors in the case of 5 input levels. The increase of levels makes only the table more complex.

In the Fig. 7, the pseudo-on-line method, that is, GFPID, uses the proposed look-up table based on the genetic algorithm. This method gives more performance than GAs on the on-line or off line. Heuristic algorithm that changes fuzzy control rule to the experience knowledge, have good performance in the steady-state error, but do not give so good performance in the transient-state error. The pseudo-on-line method provides good performances in the steady-state and transient-state errors.

Table 2. Results of optimization using 5-level.

ie \ e \ de		level 1	level 2	level 3	level 4	level 5
level 1	level 1	0.0018	0.0037	0.0024	0.0104	0.0028
	level 2	0.0043	0.0063	0.0052	0.0098	0.0050
	level 3	0.0036	0.0044	0.0039	0.0073	0.0049
	level 4	0.0078	0.0094	0.0081	0.0135	0.0095
	level 5	0.0040	0.0064	0.0049	0.0080	0.0078
level 2	level 1	0.0019	0.0037	0.0023	0.0094	0.0028
	level 2	0.0044	0.0064	0.0052	0.0098	0.0051
	level 3	0.0037	0.0050	0.0039	0.0084	0.0049
	level 4	0.0078	0.0095	0.0081	0.0138	0.0095
	level 5	0.0040	0.0065	0.0050	0.0082	0.0030
level 3	level 1	0.0020	0.0038	0.0023	0.0094	0.0029
	level 2	0.0044	0.0064	0.0052	0.0085	0.0050
	level 3	0.0038	0.0051	0.0040	0.0085	0.0050
	level 4	0.0079	0.0095	0.0082	0.0138	0.0095
	level 5	0.0042	0.0066	0.0051	0.0083	0.0031
level 4	level 1	0.0019	0.0037	0.0024	0.0092	0.0028
	level 2	0.0043	0.0063	0.0051	0.0086	0.0049
	level 3	0.0038	0.0061	0.0041	0.0084	0.0049
	level 4	0.0078	0.0094	0.0094	0.0135	0.0096
	level 5	0.0040	0.0064	0.0051	0.0082	0.0030
level 5	level 1	0.0018	0.0036	0.0023	0.0091	0.026
	level 2	0.0042	0.0063	0.0050	0.0085	0.0048
	level 3	0.0037	0.0059	0.0041	0.0082	0.0049
	level 4	0.0077	0.0096	0.0093	0.0130	0.0029
	level 5	0.0038	0.0062	0.0050	0.0081	0.0025

V. Simulations

Several simulations have been carried out to examine the feasibility of the proposed pseudo-on-line algorithm for induction motor system that is described as the type of fifth-order nonlinear differential equation. Using the Runge-Kutta method, the numerical solution was obtained.

$$\frac{d}{dt} i_{qs} = \frac{-1}{L_s L_r - L_m^2} (R_s L_r i_{qs} + \omega_r L_m^2 i_{ds} - R_r L_m i_{qr} + \omega_r L_r L_m i_{dr} - L_r V_{qs})$$
$$\frac{d}{dt} i_{ds} = \frac{-1}{L_s L_r - L_m^2} (-\omega_r L_m^2 i_{qs} + R_s L_r i_{ds} - \omega_r L_r L_m i_{qr} + R_r L_m i_{dr} - L_r V_{ds})$$
$$\frac{d}{dt} i_{qr} = \frac{-1}{L_s L_r - L_m^2} (R_s L_m i_{qs} - \omega_r L_s L_m i_{ds} + R_r L_s i_{qr} - \omega_r L_r L_s i_{dr} - L_r V_{qs})$$
$$\frac{d}{dt} i_{dr} = \frac{-1}{L_s L_r - L_m^2} (-\omega_r L_s L_m i_{qs} - R_s L_m i_{ds} + \omega_r L_r L_s i_{qr} + R_r L_s i_{dr} - L_r V_{ds})$$
$$\frac{d}{dt} \omega_r = \frac{3L_m}{2J} (i_{qs} i_{dr} - i_{ds} i_{qr}) - \frac{B}{J} \omega_r - \frac{1}{J} T_i . \quad (24)$$

Torque equation can be as follows[8]:

$$T_e = \frac{3}{2} \frac{P}{2} \frac{1}{\omega_b} (\phi_{qr} i_{dr} - \phi_{dr} i_{qr}) \quad (25)$$

where $\phi = \omega_b \lambda$, $\omega_b = 2\pi f_{rated}$ electric radians per second and f_{rated} being the rated frequency in Hertz of the machine.

Voltage source frequency ω can be related to the rotor frequency ω_r as follows[14]:

$$\omega_e = \frac{R_r L_m}{\phi_{dr} L_r} \cdot i_{qs} + \omega_r . \quad (26)$$

Table 3 shows the rated values and the nominal parameters of a tested machine. Simulation results are depicted in Fig. 8 ~ 13, when the motor speed is changed from -600[rpm] to 600[rpm], applied the control techniques proposed in the previous section. In these figures, 30 levels case is compared for the given method in viewpoint of

Table 3. Motor parameters.

Normal Output	P_0	1.1Kw
Normal Rotational Frequency		1000RPM
Stator Resistance	R_s	0.2842Ω
Rotor Resistance	R_r	0.2878Ω
Stator Inductance	L_{ls}	0.02827H
Rotor Inductance	L_{lr}	0.02827H
Mutual Inductance	L_m	0.02682H
Leakage Coefficient	σ	0.116
Number of Pole Pairs	P	3
Moment of Inertial	J	$0.0179Kg\text{-}m^2$
Stator Current	I_{ds}	15.4A
Stator Voltage	V_{qs}	137V
Natural Frequency	W_n	160 rad/sec
Damping Coefficients	δ	0.539
Electrical Time Constant	T_E	5.37 msec
Mechanical Time Constant	T_M	6.73 msec

motor speed and torque component current. We use also eqn.(27) as performance index (PI) of induction motor.

$$PI = \int \sqrt{e^2} \tag{27}$$

where, e is error of motor speed.

As shown in Fig. 8~10 with no load, the proposed controller reduces the rise time and improves maximum overshoot. But the torque component current has a little non-linearity. To compare with no-load, each method shows almost same performance. But torque component current had a little amplitude of bias.

As shown in Fig. 11~13 with load, the proposed controller has the same performance for the rise time and the performance index. But this method improves maximum overshoot and settling time. The torque component current has a little non-linearity, that is, the torque component current oscillates in the steady state. This is a general characteristic of powerful controller like fuzzy controller that the small errors occur the large reactions.

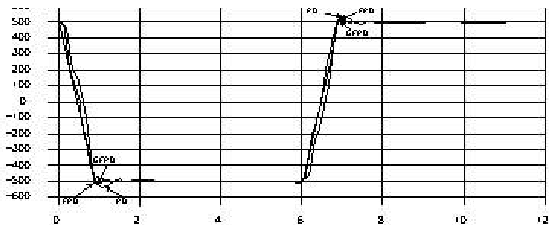

Fig. 8. Motor speed (x-sec., y-rpm, No Load).

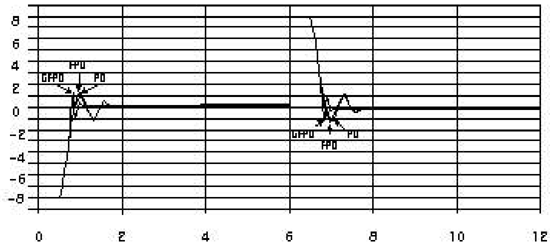

Fig. 9. Torque component current (x-sec., y-mA, No Load).

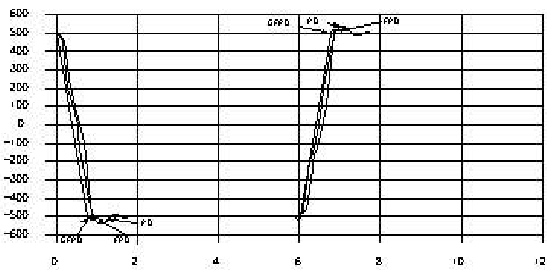

Fig. 10. Emulator output (x-sec., y-rpm, No Load).

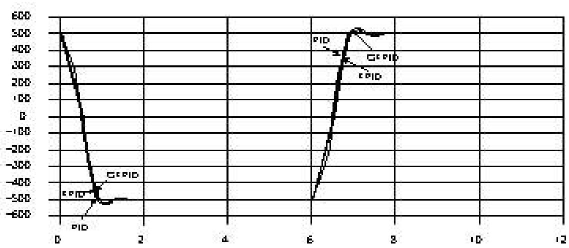

Fig. 11. Motor speed (x-sec., y-rpm, Load).

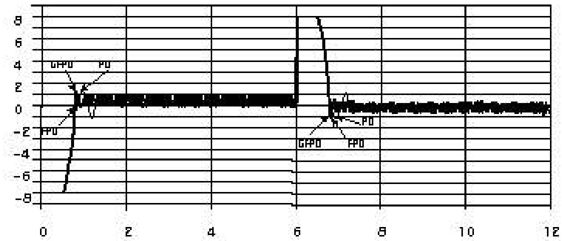

Fig. 12. Torque component current(x-sec., y-mA, Load).

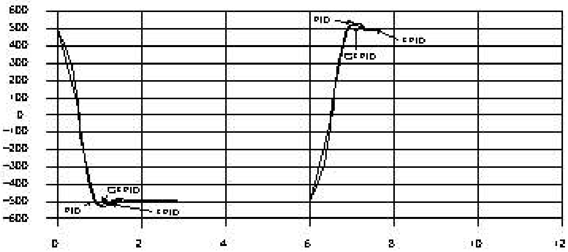

Fig. 13. Emulator output (x-sec., y-rpm, Load).

Table 5 represent abilities of each controller for no load. The GFPID method gives good results in rise time and settling time.

Table 6 represent abilities of each controller for load. The propose method especially gives good results in rise time and settling time.

Table 5. Ability of each controller for no load.

Evaluation / Method	Estimated Value (PI)	Rise Time(s)	Maximum Overshoot(%)	Settling Time(s)
PID	13.577	0.640	4.4	1.844
FPID	12.425	0.639	4.3	1.728
GFPID	11.532	0.634	1.9	1.324

Table 6. Ability of each controller for load.

Evaluation / Method	Estimated Value(PI)	Rise Time(s)	Maximum Overshoot(%)	Settling Time(s)
PID	13.834	0.641	4.5	3.557
FPID	12.664	0.639	4.4	1.794
GFPID	11.624	0.635	1.9	1.328

VI. Conclusions

This paper proposed a novel method with on-line scheme using look-up table based on the genetic algorithm. This technique is an pseudo-on-line method (GFPID) that optimally estimate the parameters of FPID controller for speed control of induction motors with nonlinear plant using the genetic algorithm.

To prove the high performance, the proposed controller is applied to the induction motor for the speed control and its computer simulation is carried out.

The results of the controller are as follows:

1) The speed control of induction motor showed that the proposed controller gains optimal performance with load

and no load obtained.

2) Through the division of input parameter region and optimal look-up table, on-line real time control with off-line performance was possible based on genetic algorithm.

3) The proposed controller achieved high performance better than conventional PID controllers. Especially, It obtained the high performance of rising time, overshoot and settling time at transient state when the level of input variable increased.

4) The proposed controller with auto-tuning method proved that it is possible to control the speed for electric vehicles with drive system of induction motor.

References

[1] B. K. Bose, "Power electronics and AC drivers", Prentice-Hall, 1986.

[2] J. Cleland and W. Turner, "Fuzzy logic control of AC induction motor," *IEEE Research Tringle Institute*, pp. 843-850, 1992.

[3] P. Vas, "Vector control of AC machines," Clarendon Press. Oxford, 1990.

[4] Y. Yoshinari, W. Pedrycz, and K. Hirota, "Construction of fuzzy models through clustering techniques" *Fuzzy sets Syst*., vol. 54, no. 2. pp. 157-165, 1993.

[5] M. Mizumoto, "Realization of PID control by fuzzy control method," *IEEE Division of Information and Computer Sciences*, pp. 709-715, 1992.

[6] W. C. Dangherity, B. Rathakrishnan, and J. Ten, "Performance evaluation of a self-tuning fuzzy controller," *IEEE Department of Computer Science,* Texas A and M University College Station TX 77843, pp. 389-397, 1992.

[7] R. Caponetto, L. Fortuna, S. Graziani, and M. Xibilia, "Genetic algorithms and applications in system engineering: a survey," *Trans. Inst. Meas. Control*, vol. 15, pp. 143-156, 1993.

[8] P. C. Krause, "Analysis of electric machinery," MeGraw-Hill Book, 1987.

[9] Y. F. Li and C. C. Lau, "Development of fuzzy algorithms for servo systems," *IEEE Control System Magazine*, pp. 65-71, Apr. 1989.

[10] L. Zheng, "A practical guide to tune of proportional and integral(PI) like fuzzy controller," *IEEE* Yamatake-Honey Co., Japan, 1992.

[11] H. A. Malki, H. Li, and G. Chen, "New design and stability analysis of fuzzy proportional-derivative control system" *IEEE Transactions on Fuzzy Systems*, November, 1994.

[12] J. H. Holland, "Adaptation in neural and artificial aystems," Ann Arbor, MI: University of Michigan Press, 1975

[13] T. Takagi and M. Sugeno, "Fuzzy identification of system and its applications to modeling and control," *IEEE Trans. on SMC-15*, no. 1 Jan./Feb., pp. 116-132, 1985.

[14] S. H. Chae, "An induction motor drive using an auto-tuning fuzzy PID control algorithm," Chung-Ang University, 1992.

Taechon Ahn

He was born in Inchon, KOREA, on October 11, 1995. He receved the B.S., M.S. and Ph.D degress in Electrical Engineering from Yonsei University in 1978, 1980 and 1986. He has been with the school of Electrical and Computer Engineering, Wonkwang University since 1981, where he is a Professor. His fields of interest are in the Digital Control and Systems Theory, Intelligent Systems Design for Control Measurement, and Application of Microprocessor Controller.

Haksoo Kang

He was born in cheju, Korea, on November 4, 1952. He received the B.S and M.S. degress in Electrical Engineering from Wonkwang University in 1983 and 1987. He is currently a Ph.D student at electric machine control laboratory, the department of Electrical Engineering, Wonkwang University. He has been with the Department of Electrical Engineering, Jeonju Technical College since 1989, where he is a professor. His fields of interest are Control of Electric Machine.

Yangwon Kwon

He was born in Chongeup, Korea, on October 20, 1972. He received the B.S degree in Department of Control & Instrumentation Engineering from Wonkwang University in 1999. He is currently a M.S student at electric machine control laboratory, the department of Electrical Engineering, Wonkwang University since 1999. His fields of interest are Control of Electric Machine and Intelligent Control.