

## Current Control of the Forklift using a Fuzzy Controller

Jong Il Bae\*

\* Dept. of Electrical Engineering, Pukyong National University, Pusan 608-739, Korea

Tel : 82-51-620-1437, Fax : 82-51-620-1437

E-mail : [jibae@pknu.ac.kr](mailto:jibae@pknu.ac.kr)

### ABSTRACT

In general, the forklift driven by DC motor drive system is used in the industrial field. Classically, the DC motor is controlled by current control using proportion control method, by output torque following the load on the plane like a manual operation. But in the industrial field, the forklift is demanded the robust drive mode. Some cases of the mode, there are trouble in torque control following slope capacity. The control is sensitive concerning about slope angle and output speed, various control method is studied for stability of speed control. In this paper, I apply current control for the self-tuning using the fuzzy controller to obtain robust, stable speed control and use stable, high efficiency control using DSP as main controller for high speed processor, embody dynamic characteristic of control compared the PI controller to the fuzzy controller.

### 1. INTRODUCTION

Adaptive control which adjusts gain of PID controller according to change in a system is used in order to control of the forklift. Under this adaptive control, however, it is practically hard to make an efficient controller due to complication of algorithm and any other reason. Also Variable Structure Control is applied as a nonlinearity control but it might cause high frequency characteristics disregarding high speed modeling on a sliding surface. In order to solve this problem the fuzzy controller with self-tuning is applied, which will perform high efficiency speed control. The efficiency of control algorithm is presented through a experiment and compared with the quality of PID controller.

### 2. METHOD

#### 2.1 DC motor in transfer function

DC motor in the forklift needs huge force of traction, which steady-state speed is determined by friction and force of traction. And rated speed of the motor, the highest terminal voltage, is controlled in a uniform current by regulating terminal voltage.

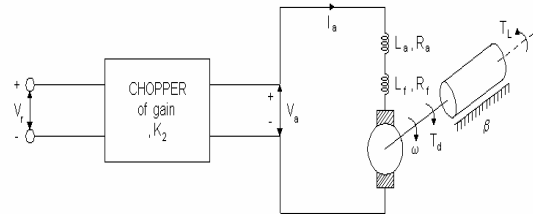


Fig. 1 Chopper-fed DC motor drive

As Fig. 1 indicates, the terminal voltage is related with the standard voltage on chopper's linearity gain  $k_2$ . Supposing that  $k_v$ , electromotive force coefficient is fixed regardless of armature current, the DC motor equation for motor system including load is induced as a formula,

$$V_a = K_2 V_r, e_g = K_v i_a \omega$$

$$V_a = R_m i_a + L_m \frac{di_a}{dt} + e_g \tag{1}$$

$$T_d = K_t i_a^2$$

$$T_d = J \frac{d\omega}{dt} + B\omega + T_L \tag{2}$$

In equation  $T_d = k_t \cdot i_a^2$ , it is characteristic of variable type nonlinear, controller is designed on limited range of operation transforming the nonlinearity system into the linearity system on the purpose of applying transfer function. So define system parameters at operating point as follows,

$$e_g = E_{g0} + \Delta e_g, i_a = I_{a0} + \Delta i_a$$

$$v_a = V_{a0} + \Delta v_a, T_d = T_{d0} + \Delta T_d \tag{3}$$

$$\omega = \omega_0 + \Delta \omega, v_r = V_{r0} + \Delta v_r$$

$$T_L = T_{L0} + \Delta T_L \tag{4}$$

where, we can find that are entirely small. The equation (2) from equation (1) can be linearization as follows.

$$\Delta v_a = K_2 \Delta v_r \Delta e_g = K_v (I_{ao} \Delta \omega + \omega_o \Delta i_a) \quad (5)$$

$$\Delta v_a = R_m i_a + L_m \frac{d(\Delta \omega)}{dt} + B \Delta e_g \quad (6)$$

$$\Delta T_d = 2K_v I_{ao} \Delta I_a$$

$$\Delta T_d = J \frac{d(\Delta \omega)}{dt} + B \Delta \omega + \Delta T_L \quad (7)$$

Which are written in the transform of Laplace space as follows.

$$\Delta V_a(s) = K_2 \Delta V_r(s) \quad (8)$$

$$\Delta E_g(s) = K_v [I_{ao} \Delta \omega(s) + \omega_o \Delta I_a(s)] \quad (9)$$

$$\Delta V_a(s) = R_m \Delta I_a(s) + sL_m \Delta I_a(s) + \Delta E_g(s) \quad (10)$$

$$\Delta T_d(s) = 2K_v I_{ao} \Delta I_a(s) \quad (11)$$

$$\Delta T_d(s) = sJ \Delta \omega(s) + B \Delta \omega(s) + \Delta T_L(s) \quad (12)$$

Which mean that the change of either reference voltage or load torque is the change of speed block diagram for the change of reference voltage and load torque is drawn in Fig. 2 and Fig. 3

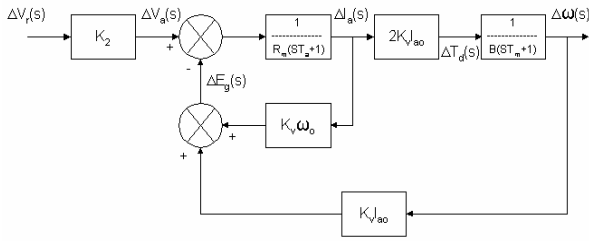


Fig. 2 Block diagram for reference voltage

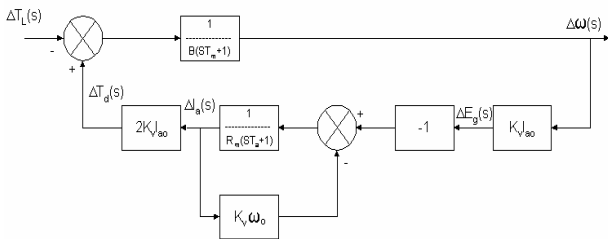
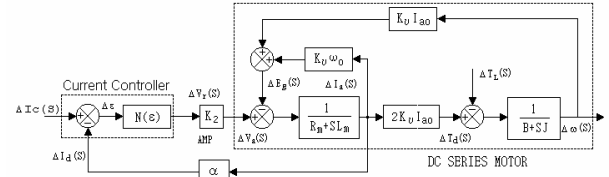


Fig. 3 Block diagram for load torque disturbances

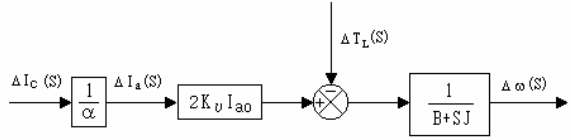
## 2.2 Current controller with closed transfer function

We can get the dynamic equation of the motor as Kirchoff's voltage equation is applied to armature field system circuits and easily obtain the dynamic equation through Newton's dynamic laws that are applied to rotation-machinery part as the electrical and mechanical combined relation equation is applied to motor current and generation torque.

In section 2.1, current sensor is connected to power circuit to convert closed-loop system, so the output sensor is amplified in proportion to supply current of rotor by  $\alpha$ -factor and generates error voltage ( $\Delta \varepsilon$ ) comparing it with current ( $\Delta I_c(s)$ )



(a) Block diagram of DC motor



(b) Simple block diagram

Fig. 4 Closed-loop block diagram of DC motor

(1) Let us solve that closed-loop step response ( $\Delta \omega(s)$ ) caused change of reference current in the block diagram

$$\frac{\Delta \omega(s)}{\Delta I_c(s)} =$$

$$\frac{N(\varepsilon) K_2 2K_v I_{ao}}{(R_m + sL_m)(B + sJ) + N(\varepsilon) K_2 \alpha (B + sJ) + K_v \omega_o (B + sJ) + sK_v^2 I_{ao}^2}$$

according to final value theorem

$$\lim_{s \rightarrow 0} \frac{\Delta \omega(s)}{\Delta I_c(s)} = \frac{N(\varepsilon) K_2 2K_v I_{ao}}{(R_m B + N(\varepsilon) K_2 \alpha B + K_v \omega_o B)}$$

(2) Speed change in normal operating- $\Delta \omega(s)$

$$\lim_{s \rightarrow 0} \frac{\Delta \omega(s)}{\Delta T_L(s)} = \frac{R + K_2 N(\varepsilon) \alpha + K_v \omega_o}{B(R_m + K_2 N(\varepsilon) + K_v \omega_o) + 2K_v^2 I_{ao}^2}$$

$$\frac{\Delta \omega(s)}{\Delta T_L(s)} =$$

$$\frac{(R_m + sL_m) + K_2 N(\varepsilon) \alpha + K_v \omega_o}{(B + Js)[(R_m + sL_m) + K_2 N(\varepsilon) \alpha + K_v \omega_o] + 2K_v^2 I_{ao}^2}$$

## 2.3 Self-tuning of the fuzzy controller

Fig.5 shows the hardware diagram for tuning membership function of the fuzzy controller. IBM PC tune membership function of the fuzzy controller. 818 Lab card will transfer a signal and DSP board will control current.

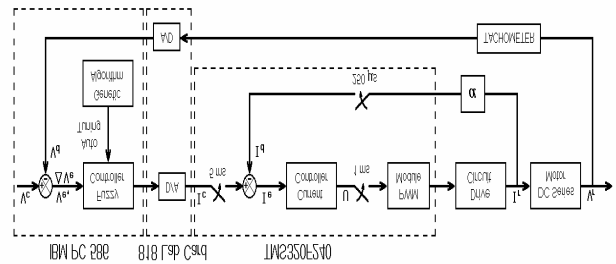


Fig. 5 The hardware diagram for magnetic tuning the fuzzy controller using PC

### 3. THE CURRENT CONTROL OF DC MOTOR USING DSP (TMS320F240).

#### 3.1 Configuration of the hardware.

We use the PI current controller for internal loop to control DC motor as Fig. 6. Configuration of hardware system shows in the Fig. 7. The internal loop consists of the current controller, PWM module, motor driver circuit and current sensor. The external loop consists of the current controller. As the Fig. 8, the current controller changes current command every 5[msec]. Setups about interrupt, PWM follow the one of PI current controller.

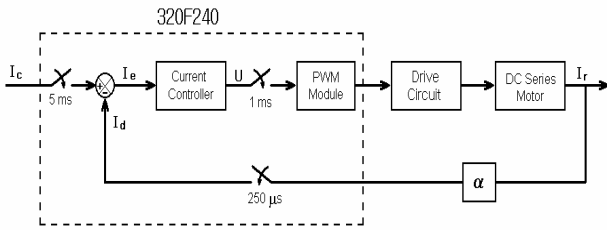


Fig. 6 The hardware diagram for current control of real system

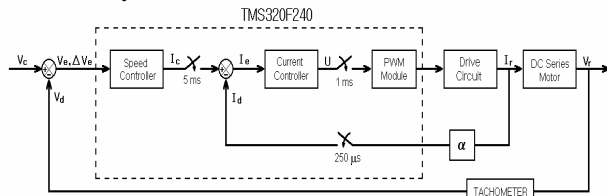


Fig. 7 The hardware diagram for speed control of DC motor

#### 3.2 The current control using PI controller.

(1) The architecture of controller.

It is possible to control of the PI controller in the Fig 8. The input value of current control is the gap between reference speed command and feedback value from the tachometer. Out put value is current command from the current controller which is internal loop. The equation used in the PI controller is the equation (13).

$$u = k_p \times e + k_i \times \int e dt$$

$$= k_p \times e + k_i \times \sum e \cdot \Delta t \quad (13)$$

It is possible to remove the noise from the tachometer, current sensor and potentiometer by taking five simple moving mean value using speed information every 5[msec], current every 1[msec] and pedal every 1[msec].

The moving mean method correspond with digital low pass filter and represented the equation (14). If weight is uniform, it can be simple movement mean method.

$$y(i) = \frac{1}{w} \sum_{j=-m}^0 \omega(j)x(i+j) \quad (14)$$

(2) The control of PI controller gain

At the graph of test results, the speed response of the top side is that 1[V] shows 500[rpm] and the current of the low side means that 1[V] shows 100[A].

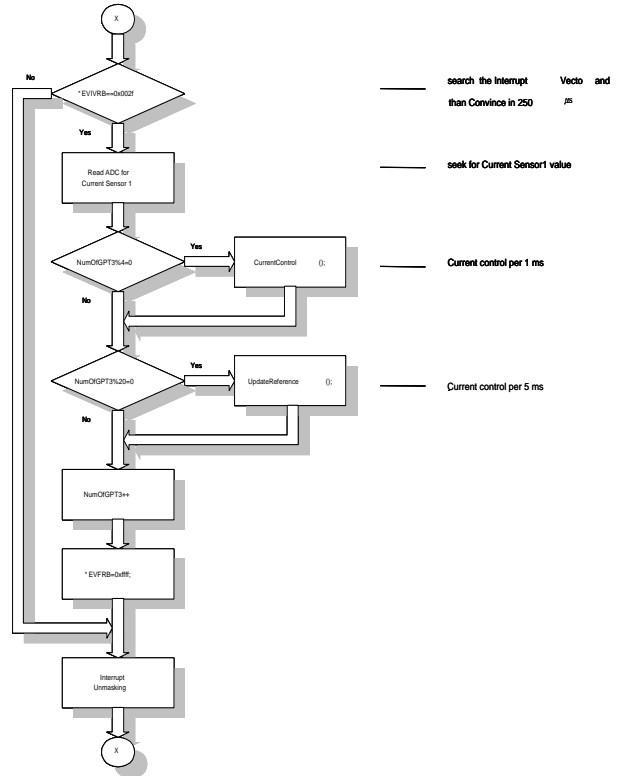


Fig. 8 Flow chart of 250[µs] interrupt subprogram

The current scale of downward oscilloscope channel 2 means that 100[mV] represents 1[V/div]. It shows the load of the weight of the forklift body concerning that load becomes 100[A] and the 160[A]. Load shows the load of weights of the forklift body and something loaded.

a) Response in changing a proportional gain

At the Fig. 9, the response is showed as  $k_p$  changes when speed command value  $k_i$ ,  $k_p$  are equal to 1500[rpm], 1, 100[A] respectively.

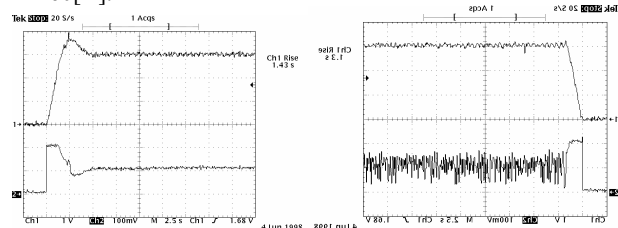
b) Response in changing an integral gain

Fig.10 shows response of changing  $k_i$  when speed command value is 1500[rpm],  $k_p$  is 5.0 and load is 100[A].

(3) Result of Test

When load is being at 100[A], Fig. 11 shows the response of changing reference speed at  $k_p = 5.0$ ,  $k_i = 0.5$

Due to such limits, the wrong response appears like steady-state error and the asymptotic slope increases with keeping the load in 160[A].

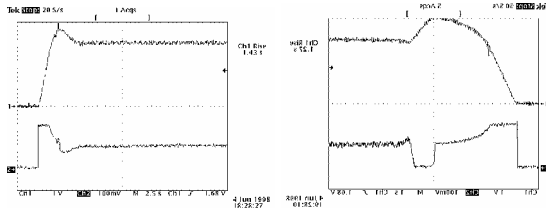


(a)  $k_i = 0.1$

(b)  $k_i = 4.0$

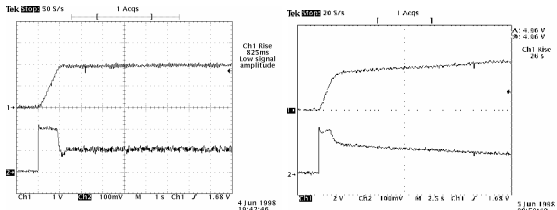
( $k_i = 1$ , load 100[A], reference speed 1500[rpm])

Fig. 9 Experiment results 1 of PI controller



(a)  $k_I = 0.1$  (b)  $k_I = 4.0$   
 $(k_I = 1, \text{load } 100[\text{A}], \text{reference speed } 1500[\text{rpm}])$

Fig. 10 Experiment results 2 of PI controller



(a) Reference speed 1000[rpm] (b) Reference speed 2500[rpm]  
 Fig. 11 Experiment results of PI controller when load 100[A] (max. 200[A])

### 3.3 Current control using the fuzzy controller

#### 3.3.1 Configuration of controller

We constitute the fuzzy controller from speed controller in Fig. 7 for speed control. Inputs of the fuzzy controller become the reference command value, the error and the change ratio of the feedback speed value from tachometer, so output has the command value of reference current of current controller like initial loop.

#### 3.3.2 Current control of the fuzzy controller of using tuned belonging function

As we consider the condition used for the fuzzy controller of fuzzy rule uses the value in Table 1 based on phase plane and belonging function uses the value in Table 2 tuned by a genetic Algorithm. Because all the belonging functions in Table 2 are normalized, we need scale values, so let's speed error scale, error change rate and output go to 0.5, 1.0 and 2.5 respectively. Here, speed error and error change rate mean 500[rpm] when they are 1.0 and output means 100[A] when it is 1.0.

Table 1 Table of 49 control rules

$e \backslash \Delta e$	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

Language Value

NB : Negative big, PS : Positive small  
 NM : Negative medium, PU : Positive medium  
 NS : Negative small, PB : Positive big

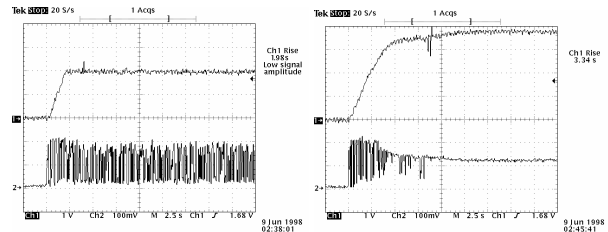
Table 2 Belonging function of tuning the fuzzy controller

Kind	NB	NM	NS	ZO	PS	PM	PB
e	-1.00	-0.42	-0.08	0.00	0.03	0.21	1.00
$\Delta e$	-1.00	-0.61	-0.25	0.00	0.31	0.74	1.00
u	-1.00	-0.62	-0.35	0.00	0.32	0.57	1.00

(1) Test when load 100[A]

a) Classification the fuzzy controller

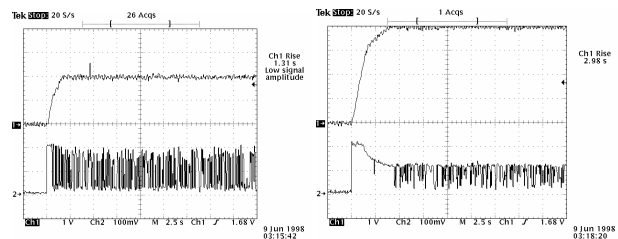
The result is showed in Fig. 12 when the fuzzy controller with self-tuning belonging function is executed by a genetic algorithm and load is 100[A]. Here, the speed information is the value from five moving mean method every 5[msec] and we get error and error rate by using that. As the response, steady-state error doesn't occur at reference speed of 1000[rpm] and the asymptotic slope to steady state reaches zero. But steady state error occurs at 2000[rpm] because output fuzzy scale doesn't match.



(a) Reference speed 1000[rpm] (b) Reference speed 2000[rpm]  
 Fig. 12 Outputs of basic fuzzy controller (load 100[A])

b) The fuzzy controller using fuzzy singleton.

The fuzzy singleton generates ON-OFF signal not like general fuzzy group. The Fig 13(a) shows that the quick response becomes slow because of the fuzzy controller from the transition response. The Fig 22(b) represent that the quick response time is improved comparing with existing fuzzy controller and steady-state error, asymptotic slope disappear.

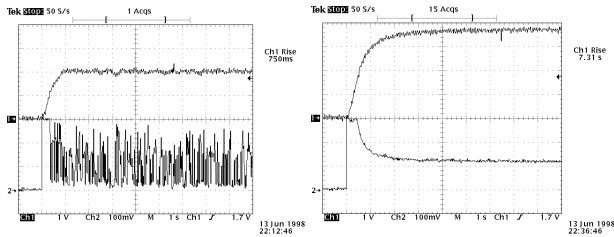


(a) Reference speed 1000[rpm] (b) Reference speed 2000[rpm]  
 Fig. 13 The fuzzy controller output using fuzzy singleton (load 100[A])

c) The fuzzy controller using individual moving mean method.

We get speed error and error rate from five moving mean value every 5[msec] using fuzzy singleton. Individual means taking five moving mean every 5[msec] from the speed information and error rate every 1[msec]. The speed error can be taken from the moving mean of speed information. As the

response, quick response improves much at the reference speed 1000[rpm] but the steady-state error occurs at 2000[rpm], because of not matching output fuzzy scale as the Fig. 14(a).



(a) Reference speed 1000[rpm](b) Reference speed 2000[rpm]  
Fig. 14 Outputs of the fuzzy controller using individual moving Mean method (load 100[A])

### 3.3.3 Result of the test

A steady-state error was not generated if a clause in some degree is satisfied in the fuzzy controller.

Fig. 15 shows quick responses of an each the fuzzy controller at load 100[A], different reference speed. So responses using individual method show very good. The fuzzy controller using individual moving method was not converged because output scale is small in the response of 2000[rpm].

Fig. 16 shows quick responses of the fuzzy controller using individual moving method when load is defined 100[A].

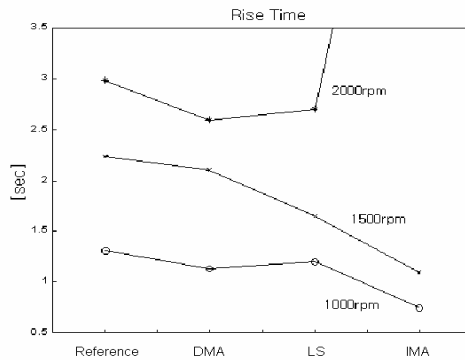
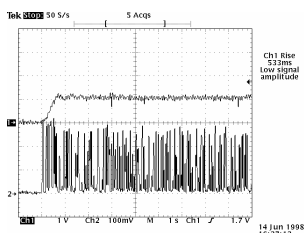
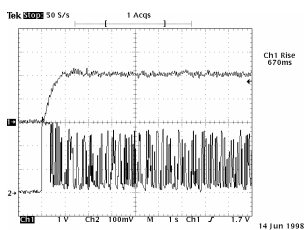


Fig. 15 Quick response each reference speed (load 100[A])



(a) load 100[A], reference speed 500[rpm]



(c) load 100[A], reference speed 1000[rpm]

Fig. 16 Responses of the fuzzy controller using individual moving method

## 4. RESULT

When load changes, the fuzzy controller didn't need tuned belonging function to be recontrolled and meted with the change of the load only as controlling output scale.

The fuzzy controller influences the change rate of error as noise exists in speed feedback, so current ripple occurs. In this paper, We apply the simple moving mean method and the individual moving mean method to decrease the effect of the noise. The result shows that the individual moving mean method is most effective performance. We need more study of the PI type fuzzy controller to decrease current ripple occurring at the fuzzy controller.

Table 3 is comparison of PI controller response when reference speed is 1000[rpm], the load is 100[A], and response of the fuzzy controller using individual moving method. It shows the fuzzy controller is better quick response and characteristics of steady-state than PI controller.

Table 3 Result comparison of PI controller and the fuzzy controller(reference speed 1000[rpm])

Function \ Controller	PI controller	Fuzzy controller
	100[A]	100[A]
Rising time [sec]	0.825	0.670
Steady State error [rpm]	20	0
Asymptotic slope rate [rpm/sec]	0	0

## 5. REFERENCE

- [1] G. C. D. Sousa and B. K. Bose, "A fuzzy set theory based control of a phase controlled converter DC machine drive," Conf. Rec, IEEE IAS Ann. Meeting, pp. 854-861, October, 1991
- [2] Sang-Rae Lee and Kwang-Won Lee, "A new variable structure position control for DC motor using fuzzy logic," KIEE Trans, Vol. 41, No. 6, pp. 625-632, 1992
- [3] TMS320C24X DSP Controllers Reference Set-Volume 1, Texas Instruments, 1997
- [4] TMS320C24X DSP Controllers Reference Set-Volume 2, Texas Instruments, 1997
- [5] Chae-suk, Young-Sik Oh, Fuzzy Theory and Control, Chung Moon Kag, 1995
- [6] Rainer Palm, Dimiter Driankov, Hans Hellendoorn, Model Based Fuzzy Control, Springer, 1997
- [7] David B. Fogel, Evolutionary Computation, IEEE PRESS, 1995
- [8] TMS320C2x/C2xx/C5x Optimizing C Compiler-User's Guide, Texas Instrument, 1995
- [9] Dimiter Driankov, Hans Hellendoorn, Michael Reinfrank, An Introduction to Fuzzy Control, Springer, 1996
- [10] TMS320C5x C Source Debugger-User's Guide, Texas Instrument, 1994
- [11] TMS320C1x/C2x/C2xx/C5x Assembly Language Tools-User's Guide, Texas Instrument, 1995
- [12] B. Kosko, Neural Networks and Fuzzy Systems, Prentice Hall, 1992