A Fuzzy Dynamic Learning Controller for Chemical Process Control

Jeong Jun SONG , Sunwon PARK
Process Systems Laboratory , Dept. of Chem. Eng. , KAIST

ABSTRACT : A fuzzy dynamic learning controller is proposed and applied to control of time delayed, non-linear and unstable chemical processes. The proposed fuzzy dynamic learning controller can self-adjust its fuzzy control rules using the external dynamic information from the process during on-line control and it can create the new fuzzy control rules autonomously using its learning capability from past control trends. The proposed controller shows better performance than the conventional fuzzy logic controller and the fuzzy self organizing controller.
KEYWORDS : Fuzzy Set Theory, Fuzzy Control, Fuzzy Dynamic Learning Controller, Fuzzy Logic Controller, Fuzzy Self Organizing Controller, Chemical Process Control

1. Introduction

Conventional fuzzy logic controllers generally require a certain reasonable set of fuzzy control rules that integrate heuristics and intuition of human operators.[1,3] However in some cases, especially in the case of a system that has very complicated dynamic characteristics, we would encounter significant difficulties to find the best fitted or at least reasonable fuzzy rules to control such a system. In these cases the initial fuzzy rules we had set at the initial control stage may not be suitable to control those processes.

A fuzzy controller which can dynamically learn its control environment and create its fuzzy control rules based on the external dynamic information from the process is called a fuzzy dynamic learning controller (henceforth FDLC). This work deals with the development of a new fuzzy inference algorithm based FDLC which can self-adjust its fuzzy control rules using the information from the process during on-line control and create the new fuzzy control rules autonomously using its learning capability from past control trends. The proposed controller is applied to three chemical processes: a process with a time delay; a non-linear process; an open-loop unstable process. It's control performance is compared with conventional fuzzy logic controller (henceforth FLC) [1] and the fuzzy self organizing controller [6] (henceforth FSOC).

2. Building Blocks of the Proposed Fuzzy Dynamic Learning Controller

2-1. Overall Structure

The modularized configuration of the fuzzy dynamic learning controller proposed in this paper is shown in Fig. 1. The controller consists of two phases: the basic control phase and the fuzzy dynamic learning phase. The error and change of error terms are generated by comparing the process state with the desired setpoint, and they are fed into controller's fuzzification block (block F). In Fig. 1., GE, GCE, and GO are controller gain terms that represent gains of error, change of error, and control output respectively. The basic control phase has a fuzzy control rule base and a fuzzy inference module which takes up the fuzzy set operation during control. Defuzzified control command is produced from the controller (through DF, GO, and SUM modules that are defuzzified, output gain treatment and velocity form summation blocks respectively) during every control cycle. The functionality of FLC is similar to these tasks of the basic control phase. The dynamic learning module in the dynamic learning phase updates fuzzy control rules or creates new fuzzy control rules.

2-2. The basic control phase

Actually the basic control phase of the fuzzy dynamic learning controller can be thought as a simple FLC.[1,4,5,7] The main design concept of this basic control phase is the use of linguistic control rule which can be expressed in various fuzzy operation forms including linguistic conditional statement (i.e., IF-premise, THEN-action rules). This linguistic conditional statement can also be described in a fuzzy implication with the relation matrix R_k. The relation matrix R_k can act as the fuzzy relations among the error E_k which is generated using the deviation between the process response and the desired setpoint, the change of error C_k and the control action U_k. The fuzzy implication can be given in the cartesian product form:

$$R_k = E_k \times C_k \times U_k \qquad (1)$$

The overall relation matrix R in a fuzzy controller is calculated as the union of n individual relation matrices:

$$R = R_1 \cup R_2 \cup \ldots \cup R_k \cup \ldots \cup R_n = \bigcup_{k=1}^{n} R_k \qquad (2)$$

As shown in Eqs. (1) and (2), the controller output U can be inferred from the controller inputs (error term E, error change term C) and the control rule (relation matrix R) using Zadeh's composition rule of inference [9]:

$$U = (E \times C) \circ R$$

For each time step, U can be presented as:

$$U(kT) = (E(kT) \times C(kT)) \circ R \qquad (3)$$

2-3. The Fuzzy Dynamic learning Phase

The fuzzy dynamic learning phase interactively modifies the basic control phase using newly developed fuzzy inference rule and some pre-determined performance index which can improve the control performance of the controller. [6] This fuzzy dynamic learning phase mainly consists of control performance measure, modification of basic control phase (rule learning and adaptation phase), and the fuzzy inference module that upgrades the fuzzy rule base.

2-3-1. Control Performance Measure

To determine the control performance of a controller, we generally rely on the deviation between the desired setpoint and the actual process response as a rough indication. This control performance measure plays a role as a decision maker for modifying control action in the fuzzy dynamic learning phase.

The performance index which can be used as a "yardstick" for control performance of the controller is composed as shown in Table 1. This table presents the amounts of reinforcements on control action in the next cycle. Of course there is not a formal rule for making the control performance index; it can be varied according to controller designer's strategy for performance improvement.

In Table 1., the controller performances are inferred as the following linguistic conditional statements:

If the ERROR is PSM then
if the Change of Error is NB or NM then
the deviation of response is NS
else
If the ERROR is PSM then
if the Change of Error is NSM then
the deviation of response is ZR.

These linguistic conditional statements can be expressed in a more compact form using fuzzified variables as:

If $E(kT) = PSM$ then if $C(kT) = NB$ or NM
then $DEV(kT) = NS$
else
If $E(kT) = PSM$ then if $C(kT) = NS$
then $DEV(kT) = ZR$

here, $DEV(kT)$ is the deviation of the process response from the desired setpoint i.e., the reinforcement of future control action.
NB, NM, NSM, NS, ZR, and PSM are fuzzified variables of the deviation.

The fuzzy set operation in the control performance measure uses Zadeh's compositional rule of inference same as the generation of the control output in the basic control phase.

2-3-2. Modification of the Basic Control Phase

We can modify the action of the basic control phase by: (1) Modification of the fuzzy control rules, (2) Modification of controller's scaling factors (gain terms), or (3) Simultaneous modification of the fuzzy control rules and gain terms. Though the second method would improve the control performance of the controller, it requires a well tuned initial control rules. The third method would also improve the control performance, but how we can adjust the two factors at a time effectively still remains as a problem to be solved. Therefore in this paper, we focus on building a fuzzy dynamic learning mechanism by modifying its fuzzy control rules. The controller can modify its rules based on the experience gained from controlling the process.

Let's assume that our process has the time delay nT. That is, the control action at time step (k–n)T has most contributed to the process response at the present time step kT. Thus for the present state of the process, the controller output at (k–n)T is most responsible and should be modified through the dynamic learning phase.

So, the initial control rule of the controller before processing the dynamic learning phase is :

$$E(kT-nT) \longrightarrow C(kT-nT) \longrightarrow U(kT-nT)$$

This rule can be modified after processing in the dynamic learning phase as follows:

$$E(kT-nT) \longrightarrow C(kT-nT) \longrightarrow V(kT-nT)$$

where,

$E(kT-nT)$	=	$F\{ e(kT-nT) \}$,
$C(kT-nT)$	=	$F\{ c(kT-nT) \}$,
$U(kT-nT)$	=	$F\{ u(kT-nT) \}$
	=	actual control action to the process at time step (kT–nT)
$V(kT-nT)$	=	$F\{ u(kT-nT) + DEV(kT) \}$
	=	adjusted control action using the performance index
$DEV(kT)$	=	the reinforcement of control action at time step kT, which is computed using the performance index
$U(kT)$	=	$F\{ u(kT) \}$
	=	the control action at current time step kT
$F\{\ \}$	=	Fuzzification

We can formulate these modification phase with fuzzy implications as below:

$$R(kT) = E(kT) \times C(kT) \times U(kT) \qquad (4)$$

$$R^{\dagger}(kT)$$

$$= E(kT-nT) \times C(kT-nT) \times U(kT-nT) \quad (5)$$

$$R^{\dagger\dagger}(kT) = E(kT-nT) \times C(kT-nT) \times V(kT-nT) \quad (6)$$

where,

$R(kT)$ = the relation matrix among control action $U(kT)$, error $E(kT)$, and change of error $C(kT)$ at current time step kT

$R^{\dagger}(kT)$ = the relation matrix used for computing control action $U(kT-nT)$ exerted on the process at time step $(kT-nT)$

$R^{\dagger\dagger}(kT)$ = the relation matrix which must replace the relation matrix $R^{\dagger}(kT)$ after the control rules are processed through the rule modification phase.

2-3-3. The Fuzzy Rule Inference

The fuzzy dynamic learning controller can generate the adequate control action using its rule modification process described in section 2–3–2 so that the process state can follow the desired setpoint. Using the relation matrices defined in section 2–3–2, the modified relation matrices of the controller which will generate the current compensating control output are actually calculated using the fuzzy inference rule below [7]:

$R(kT)_{new}$

=[$R(kT)$ but not $R^{\dagger}(kT)$] else $R^{\dagger\dagger}(kT)$

i.e.,

$$R(kT)_{new} = [R(kT) \cap \overline{R^{\dagger}(kT)}] \cup R^{\dagger\dagger}(kT) \quad (7)$$

where,

$R(kT)$ = The relation matrix at current time step kT.

$R(kT)_{new}$ = The new modified relation matrix in the fuzzy dynamic learning phase. A new control action $U(kT)_{new}$ can be calculated using this matrix through and exerted to the process at current time step kT.

Using Eq. (7), the controller output at current time step kT becomes $U(kT)_{new}$ rather than $U(kT)$.

There can be several ways to modify the relation matrix using $R(kT)$, $R^{\dagger}(kT)$, and $R^{\dagger\dagger}(kT)$. We propose the inference logic of the fuzzy dynamic learning controller as follows:

$R(kT-nT)_{new}$

= [$R^{\dagger}(kT)$ but not $R^{\dagger}(kT)$] else $R^{\dagger\dagger}(kT)$

i.e.,

$$R^{\dagger}(kT)_{new} = [R^{\dagger}(kT) \cap \overline{R^{\dagger}(kT)}] \cup R^{\dagger\dagger}(kT) \quad (8)$$

where,

$R^{\dagger}(kT)$ = the relation matrix fired for computing control action $U(kT-nT)$ which exerted on the process at time step $(kT-nT)$

$R(kT-nT)_{new}$ = the relation matrix which should be replaced $R^{\dagger}(kT)$

The main idea of the proposed fuzzy inference operation is that the controller has an updating function of the past control rules i.e., the fired control rules, by modifying those relational matrices as in Eq. (8) thus the controller can provide effective compensative control action to the next control cycle. The controller which has the fuzzy rule inference mechanism as Eq. (8) shows better control performance than the controller of which the inference mechanism based on Eq. (7). The simulation results are discussed in the later section. Even though the terms within the bracket in Eq. (8) [$R^{\dagger}(kT)$ but not $R^{\dagger}(kT)$] looks like a null set, it is surely not a null set in fuzzy set operations. The meaning of Eq. (8) is that the controller can modify the fuzzy control rule that was fired at time step (kT–nT), using the weighted sum of the current process information (i.e., through performance index) and the control action which was taken at time step (kT–nT). On the other hand, when the controller actually attempts to compute the fuzzy set operations using Eqs. (7) or (8), the computing load to the computer system in which the controller algorithm is embedded increases very rapidly as the total number of fired rules in a time step and the desired accuracy increases. Thus the computing load in any fuzzy self organizing controller or fuzzy dynamic learning controller is a very important problem that must be resolved. We adopt a time–averaged and prioritized weighting concept on fired rules so that the most contributed rule within a time step is modified primarily in the controller's rule relation system.

3. Implementation of the Fuzzy Dynamic Learning Controller

3-1. Fuzzification and Quantization

The error and change of error that are used as input terms in the fuzzy dynamic learning controller are computed as follows:

error : $E(kT) = Ysp - Y(kT)$

change of error :

$C(kT) = Y((k-1)T) - Y(kT)$

$= E(kT) - E((k-1)T)$

where,

Ysp = desired setpoint of the process output

$Y(kT)$ = actual process output at the time step kT

The linguistic fuzzy variables for fuzzification of controller input terms (error and change of error) and output terms (control action) are defined in the universe of discourse [–1,1] in the order of Negative Big (NB), Negative Medium (NM), Negative Small Medium (NSM), Negative Small (NS), Zero (ZR), Positive Small (PS), Positive Small Medium (PSM), Positive Medium (PM), and Positive Big (PB). These nine fuzzy linguistic variables are normalized in the universe of discourse [–1 , 1] and quantized as nine fuzzy subsets.

Each fuzzy subset has a bell shaped membership function like:

$$\mu(X) = EXP ((-1/\alpha\hat{}2) * (X - \beta))$$

where,

X = controller input variable (E , C)

2α = half span of Bell shaped membership function

β = moving center (center of membership function)

Since all the fuzzy variables are normalized and quantized on the universe of discourse [–1 , 1], the parameter α has the constant value 0.125, β has the varying constants from –1 to 1 with an increment of 2α. These fuzzy variables and their membership functions are shown in Fig. 2. The maximum number of fuzzy control rules that the controller can possess is 81 because there exist nine fuzzy variables and two controller inputs (E and C).

3–2. Rule Base Structuring

The need for using the fuzzy dynamic learning controller exists in certain processes that have complex dynamic characteristics which are very difficult to identify properly. So in this paper, we assume that we cannot know the dynamic behavior of the process very well and consequently, the initial 19 fuzzy control rule sets in the controller's rule base are selected tentatively as shown in Table 2. The components of this initial rule base are continually updated or newly generated in the dynamic learning phase during on–line control cycles. We set the initial rule base for the FSOC same as Table 2.

The rule base of the FLC which would be compared with the proposed fuzzy dynamic learning controller consists of 25 fuzzy control rule elements that have been extracted from the typical response of a second order process to step changes of the process setpoints. Table 3. shows the rule table of the FLC.

3–3. Inference Logic and Defuzzification

We have tested two methods of computing compositional rule of inference for generating the control action:

Zadeh's method:

$OR(\mu_a,\mu_b) = \max (\mu_a , \mu_b)$

$AND(\mu_a,\mu_b) = \min (\mu_a , \mu_b)$

Lukasiewicz's method:

$OR(\mu_a,\mu_b) = \min (1 , \mu_a + \mu_b)$

$AND(\mu_a,\mu_b) = \max (0 , \mu_a + \mu_b - 1)$

We did not see significant differences in control performance between those two fuzzy operation methods applied to the proposed FDLC, the FSOC, and the FLC respectively. Thus we adopted Zadeh's AND and Lukasiewicz's OR operation for computing compositional rule of inference [2].

The singleton (crisp) value of the control action U(nT) which is actually fed into the process can be computed from the fuzzy value of control action U, using the center of gravity method [5].

4. Experimental Process Models

Three example systems have been simulated to demonstrate the control performance of the fuzzy dynamic learning controller proposed in this work. The first example is a second order plus dead time process (henceforth SOPDT process) within which our desired setpoints are varying in a cyclic manner. The second example system taken from Uppal et. al. [8] is a CSTR in which first order exothermic irreversible reaction is taking place. This process is highly non–linear. Finally, the third system is a first order open loop unstable process which has a pole at +5. Control performances of the proposed fuzzy dynamic learning controller (FDLC) for the three systems are compared with those of the FLC and FSOC. The first example is a servo problem; the remaining two are servo/regulatory problems of which the controller's task is both rejecting the external unmeasurable noisy disturbance and tracking the setpoint at the same time. The process models are as follows:

Example 1 : Second Order Plus Dead Time Process

$$y'' = -2.8\, y' - 4\, y + 4\, u(k-2)$$

where, y = process output, u = control action

Example 2 : Open–loop Unstable Process

$$y' = 0.5\, u + 5\, y$$

where, y = process output, u = control action

Example 3 : Non–linear Process : exo. irr. rxn. in CSTR

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix} + \begin{bmatrix} 0 \\ \beta \end{bmatrix} u + \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \end{bmatrix}$$

$$\begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix} = \begin{bmatrix} -x_1 + Da(1-x_1)\exp[x_2/(1+(x_2/\nu))] \\ x_2 + BDa(1-x_2)\exp[x_2/(1+(x_2/\nu))] - \beta(x_2 - x_{2_{co}}) \end{bmatrix}$$

Here, x_1 and x_2 are the dimensionless composition and temperature respectively; Da, B, ν, β, and $x_{2_{co}}$ are the standard dimensionless parameters; d_1 and d_2 are the dimensionless feed temperature and feed composition fluctuations, respectively. The control variable u is the dimensionless temperature of the cooling jacket unit. We select the dimensionless parameters to be: $Da = 0.072$, $B = 8$, $\beta = 0.3$, $\nu = 20$, and $x_{2_{co}} = 0$. The proposed value of x_1^{sp} is 0.5, x_2^{sp} is 3.03 [8].

5. Simulation Results

5–1. Example 1: Performance Improvement with the Proposed Fuzzy Dynamic Learning Controller

We can get the improved control performance with the proposed fuzzy dynamic learning controller for controlling the SOPDT process as shown in Fig. 3. The control performances of the FLC and the FSOC are also shown in the same figure. In this system we change the setpoints in a sequential manner such as 0.0 ⟶ 0.5 ⟶ 0.0 and the one control cycle is denoted as one RUN. The control task of the controllers is to track the changing setpoint.

For the case of the proposed controller, the initial control performance (denoted as 1 RUN) is rather unfavorable from the view of oscillation and overshoot but as the RUN sequences are repeated (i.e., as the time proceeded) the control performance gradually improves. The controller starts the control task using only 19 initial arbitrary fuzzy control rules among maximum 81 probable rules, the controller can create the missing rules from dynamic information of the process, and continually improve it during on–line control. The FSOC also shows favorable performance as time proceeds, the overshoot is rather bigger than that of the proposed fuzzy dynamic learning controller.

The FLC continually produces unfavorable process response even after the RUN sequences are repeated. This is mainly because the fuzzy control rule base of the FLC are made from the simple second order process model and so the initial fuzzy rules are inadequate for controlling the SOPDT process, and secondly because some self adaptive mechanism does not exist that modifies controller's rule base accordingly as the controller's external environment changes. (So the initial rules could not be modified at all.)

5–2. Example 2: Open–loop Unstable Process

An open–loop unstable process that has a pole at +5 is controlled by the proposed fuzzy dynamic learning controller (Fig. 4.), FSOC (Fig. 5.), and FLC (Fig. 6.). The control task of this system is to regulate the process at the desired setpoint against the external unmeasurable noisy disturbances changing in a step manner and at the same time, to track the varying process setpoint that varies every 100 cycles interval.

Control performance of the proposed controller is very satisfactory even though some overshoots appear when large external step disturbance is exerted on the process (at the time near 700 cycle). During the control phase, the controller autonomously set its own fuzzy control rule using the controller's dynamic learning mechanism. The FSOC made continuous oscillation over the control cycles, which is not desirable in controlling unstable processes.

The FLC in this case is characterized by severe oscillations and frequent on–off control actions; it is natural because the fuzzy control rules of the FLC are based on a simple second order process model.

5–3. Example 3: Non–Linear Process

Figs. 7. to 9. respectively show the performance of the FDLC, FSOC and FLC controlling a CSTR within which a first order irreversible exothermic reaction takes place. The setpoint is 0.5 and the desired operating range is ±0.5 from this operating point.

In spite of the severe non–linearity of the process the proposed controller shows good control performance over the control range. The FSOC shows worse performance than the proposed controller near the optimum operating point 0.5. (severe oscillation occurred) and reveals an unfavorable off–set near the setpoint 0.9 which is 0.4 higher than the operating point. The FLC shows similar control performance in the case of the unstable process; it causes very frequent on–off control actions over the control range.

6. Conclusions

We propose a fuzzy dynamic learning controller with an improved dynamic learning inference mechanism for control of complicated processes for which we can set up neither the reasonable initial fuzzy control rules extracted from operational experience nor mathematical dynamic model. The proposed controller has been tested with processes that have time delays, instability, or non–linearity and its control performance has been compared with those of the FLC and FSOC.

With the new dynamic learning mechanism using the proposed fuzzy inference operation, the controller can learn and create its own fuzzy control rules effectively using the past control trends and dynamic information from the process. And the controller can modify the control rules adequately by self–adapting to the varying external conditions such as frequently varying process setpoints or severe external unmeasurable disturbances to the process to be controlled. The distinct advantage of the proposed controller is that in spite of the complexity of the process and the uncertainty of process models (blackbox model), the controller provides robust control performance over the control cycles. Many simulation experiments indicate that the proposed controller can provide better control performance than the other fuzzy controllers compared in this study.

Acknowledgement : Financial support from KAIST and ERC Automation Research Center is gratefully appreciated.

REFERENCES

1. Assilian, S. and E. Mamdani, Int. J. Man–Machine Stud., 7, 1, 1975
2. HAO YING, et al., Automatica, Vol.*26*, No.3, 513, 1990
3. Kickert, W.J.M. and E.H. Mamdani, Fuzzy Sets and Systems, *1*, 29, 1978
4. King, P.J. and E.H. Mamdani, Automatica, Vol.*13*, 235, 1977
5. Lee, C.C., IEEE Trans. SMC, Vol.20, No.*2*, 404, 1990
6. Procyk, T.J. and E.H. Mamdani, Automatica, Vol.*15*, 15, 1979
7. Sugeno, M., Information Sciences, *36*, 59, 1985
8. Uppal, A., et. al., Chem. Eng. Sci., *31*, 205, 1976
9. Zadeh, L.A., IEEE Trans. Sys.,Man, and Cyb., Vol.SMC–3, No.*1*, 28, January, 1973

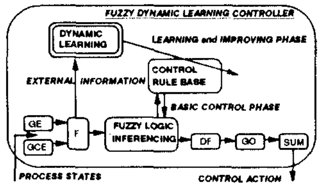

Fig. 1. Overall Structure of Fuzzy Dynamic Learning Controller

Fig. 2. Universe of Discourse and Membership for Linguistic Term Set

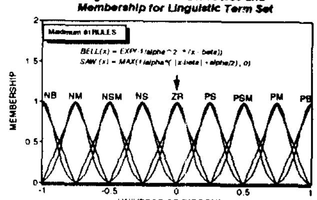

Fig. 3. Learning Capability Comparision
Servo SEQUENCE

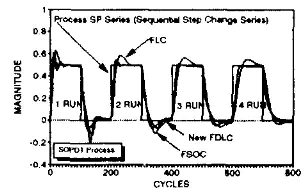

Fig. 4. Performance of New FDLC
Servo / Regulatory PARALLEL

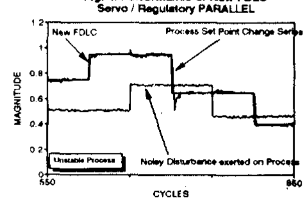

Fig. 5. New FDLC vs. FSOC
Servo / Regulatory PARALLEL

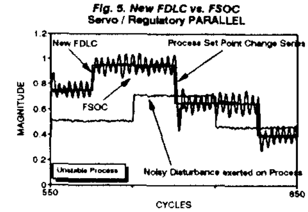

Fig. 6. New FDLC vs. FLC
Servo / Regulatory PARALLEL

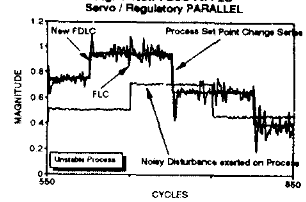

Fig. 7. Performance of New FDLC
Servo/Regulatory PARALLEL

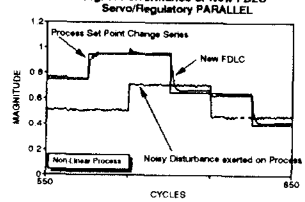

Fig. 8. New FDLC vs. FSOC
Servo/Regulatory PARALLEL

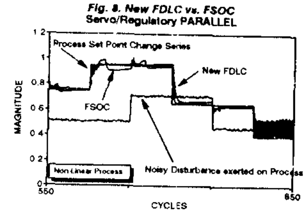

Fig. 9. New FDLC vs. FLC
Servo/Regulatory PARALLEL

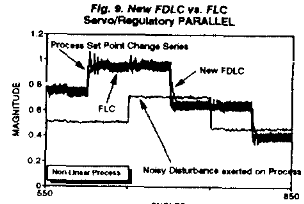

Table 1. Controller Performance Index
Max. 81 Quantizing Level

ERROR (Ysp - Yk) \ CHANGE OF ERROR (Yk-1 - Yk)	NB	NM	NSM	NS	ZR	PS	PSM	PM	PB
NB	NB	NM	NM	NSM	NSM	NS	NS	NS	ZR
NM	NM	NM	NSM	NSM	NS	NS	NS	ZR	PS
NSM	NM	NSM	NSM	NS	NS	NS	ZR	PS	PS
NS	NSM	NSM	NS	NS	NS	ZR	PS	PS	PS
ZR	NSM	NS	NS	NS	ZR	PS	PS	PS	PSM
PS	NS	NS	NS	ZR	PS	PS	PS	PSM	PSM
PSM	NS	NS	ZR	PS	PS	PS	PSM	PSM	PM
PM	NS	ZR	PS	PS	PS	PSM	PSM	PM	PM
PB	ZR	PS	PS	PS	PSM	PSM	PM	PM	PB

CHANGE OF ERROR (Yk-1 - Yk)

Table 2. Starting Rules for FDLC
Total 19 Rules among Max. 81 Rules

ERROR (Ysp - Yk) \ CHANGE OF ERROR (Yk-1 - Yk)	NB	NM	NSM	NS	ZR	PS	PSM	PM	PB
NB					NSM				
NM					NS				
NSM					NS				
NS					NS	ZR			
ZR	NSM	NS	NS	NS	ZR	PS	PS	PS	PSM
PS				ZR	PS				
PSM					PS				
PM					PS				
PB					PSM				

CHANGE OF ERROR (Yk-1 - Yk)

Table 3. Control Rules for FLC
Total 25 Rules among Max. 81 Rules

ERROR (Ysp - Yk) \ CHANGE OF ERROR (Yk-1 - Yk)	NB	NM	NSM	NS	ZR	PS	PSM	PM	PB
NB					NSM	NS			
NM					NS				
NSM					NS				
NS					NS		PS	PS	PS
ZR	NSM	NS	NS	NS	ZR	PS	PS	PS	PSM
PS	NS	NS	NS		PS				
PSM					PS				
PM					PS				
PB				PS	PSM				

CHANGE OF ERROR (Yk-1 - Yk)