3-2002

A Stable Self-Tuning Fuzzy Logic Control System for Industrial Temperature Regulation

Zhiqiang Gao
Cleveland State University, Z.GAO@csuohio.edu

Thomas A. Trautzsch
Multi-Dimensional Imaging, trautzsch@mdivac.com

James G. Dawson
Setech Inc.,, dawson@setechusa.com

Original Citation

Zhiqiang, G., Trautzsch, T. A., & Dawson, J. G. (2002). A stable self-tuning fuzzy logic control system for industrial temperature regulation. IEEE Transactions on Industry Applications, 38, 2, 414-424.

Repository Citation

Gao, Zhiqiang; Trautzsch, Thomas A.; and Dawson, James G., "A Stable Self-Tuning Fuzzy Logic Control System for Industrial Temperature Regulation" (2002). *Electrical Engineering and Computer Science Faculty Publications*. 105.
https://engagedscholarship.csuohio.edu/enece_facpub/105

A Stable Self-Tuning Fuzzy Logic Control System for Industrial Temperature Regulation

Zhiqiang Gao, *Member, IEEE*, Thomas A. Trautzsch, and James G. Dawson, *Member, IEEE*

***Abstract*—A closed-loop control system incorporating fuzzy logic has been developed for a class of industrial temperature control problems. A unique fuzzy logic controller (FLC) structure with an efficient realization and a small rule base that can be easily implemented in existing industrial controllers was proposed. The potential of FLC in both software simulation and hardware test in an industrial setting was demonstrated. This includes compensating for thermo mass changes in the system, dealing with unknown and variable delays, operating at very different temperature set points without retuning, etc. It is achieved by implementing, in the FLC, a classical control strategy and an adaptation mechanism to compensate for the dynamic changes in the system. The proposed FLC was applied to two different temperature processes and performance and robustness improvements were observed in both cases. Furthermore, the stability of the FLC is investigated and a safeguard is established.**

***Index Terms*—Fuzzy logic, self-tuning, temperature control.**

I. Introduction

While modern control theory has made modest inroads into practice, fuzzy logic control has been rapidly gaining popularity among practicing engineers. This increased popularity can be attributed to the fact that fuzzy logic provides a powerful vehicle that allows engineers to incorporate human reasoning into the control algorithm. As opposed to the modern control theory, fuzzy logic design is not based on the mathematical model of the process. The controller designed using fuzzy logic implements human reasoning that can be programmed into fuzzy logic language (membership functions, rules, and the rule interpretation).

Industrial interest in fuzzy logic control as evidenced by the many publications on the subject in the control literature has created an awareness of its increasing importance in the academic community [1]–[6], [13]–[24]. The fast improvements in the processing power of modern digital control technology make the FLC viable and appealing in many industry sectors. The self-tuning fuzzy logic design was investigated by many researchers for various problems; see, for example, [21]–[24]. The results reported in the literature are usually application specific and not easily portable. The complexity of the control algorithm and the lack of intuition in tuning limit the scope of the applications for many proposed methods.

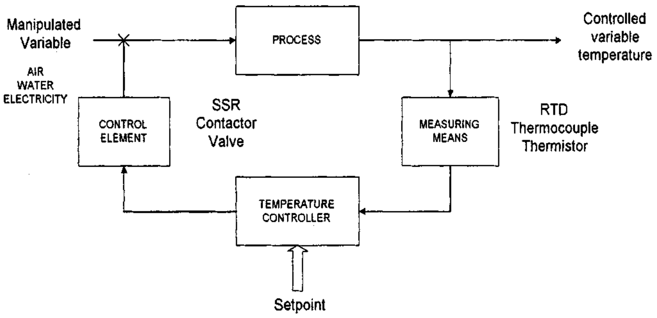


Fig. 1. A typical industrial temperature control problem.

In this paper, we concentrate on fuzzy logic control as an effective alternative to the current proportional-integral-derivative (PID) method used widely in industry. The controller, including the self-tuning algorithm, must be simple to understand and implement by practicing engineers. Consider a generic temperature control application shown in Fig. 1.

The temperature is measured by a suitable sensor such as thermocouples, resistive thermal devices (RTDs), thermistors, etc., and converted to a signal acceptable to the controller. The controller compares the temperature signal to the desired set-point temperature and actuates the control element. The control element alters the manipulated variable to change the quantity of heat being added to or taken from the process. The objective of the controller is to regulate the temperature as close as possible to the set point.

To test the new fuzzy logic control algorithms, two temperature regulation processes were used in this research. One uses hot and cold water as the manipulated variable and a valve as the controller element, and the other uses electricity as a power source to a heater, actuated by a solid-state relay (SSR). The new algorithms were tested extensively in both simulation and the hardware tests.

A. Motivation

Currently, the classical PID control is widely used with its gains manually tuned based on the thermal mass and the temperature set point. Equipment with large thermal capacities requires different PID gains than equipment with small thermal capacities. In addition, equipment operation over wide ranges of temperatures (140°–500°), for example, requires different gains at

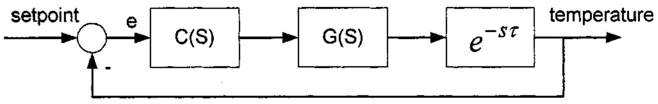


Fig. 2. A closed-loop temperature control system.

the lower and higher end of the temperature range to avoid overshoots and oscillation. This is necessary since even brief temperature overshoots, for example, can initiate nuisance alarms and costly shutdowns to the process being controlled. Generally, tuning the PID constants for a large temperature control process is costly and time consuming. The task is further complicated when incorrect PID constants are sometimes entered due to the lack of understanding of the temperature control process.

The difficulty in dealing with such problems is compounded with variable time delays existing in many such systems. Variations in manufacturing, new product development, and physical constraints place the RTD temperature sensor at different locations, inducing variable time delays (dead time) in the system.

It is also well known that PID controllers exhibit poor performance when applied to systems containing an unknown nonlinearity such as dead zones saturation and hysteresis. It is further understood that many temperature control processes are nonlinear. Equal increments of heat input, for example, do not necessarily produce equal increments in temperature rise in many processes, a typical phenomenon of nonlinear systems.

The complexity of these problems and the difficulties in implementing conventional controllers to eliminate variations in PID tuning motivate us to investigate intelligent control techniques such as fuzzy logic as a solution to controlling systems in which time delays, nonlinearities, and manual tuning procedures need to be addressed.

B. Time-Delay Problem and Existing Solutions

To study the temperature control problem using classical control techniques, a simplified block diagram, in Fig. 2, is used, instead of Fig. 1, where $C(s)$ represents the controller and $G(s)e^{-s\tau}$ the plant with a pure time delay of τ. It is well known that the time delay makes the temperature loops hard to tune. The time delay problem may be characterized by *large* and *small* delays. A linear time invariant system with finite delay τ can be modeled as $G(s)e^{-s\tau}$, where $G(s)$ is a rational transfer function of s. Note that the delay corresponds to a phase shift of $-\omega\tau$, where ω denotes the frequency. Small phase shifts at frequencies of interest may be viewed as perturbations and incorporated into a delay free design with sufficient phase margin. A large delay is classified as a delay that significantly affects the stability and phase margins to the point that delay-free design methods will not be sufficient.

A number of time-delay compensation and prediction schemes have been developed and/or improved with modifications as shown in [7]–[12]. The performance of Smith predictor control (SPC) was studied experimentally in [8]. It shows that the system performs well if the process model is accurate, but that performance degrades rapidly with inaccuracy in the process parameters and time delay. Clearly, for an unknown or variable time delay, Smith predictive compensation is no longer a viable technique.

Several control design methods for systems with varying time delays have appeared in recent literature including an estimation and self-tuning method proposed in [10], a variable-structure controller in [11], and a model reference adaptive approach in [12], to name a few.

For systems with large time delays, most design approaches use a prediction mechanism as part of the controller to simulate the process for given system parameters and time delay. In the well-known Smith predictor [7], the controller output is fed through models of the process with delay, and the process without delay, respectively. The difference of the output signals is added to the actual plant output and then fed back to the controller, thus allowing the controller to act on the prediction of the plant output.

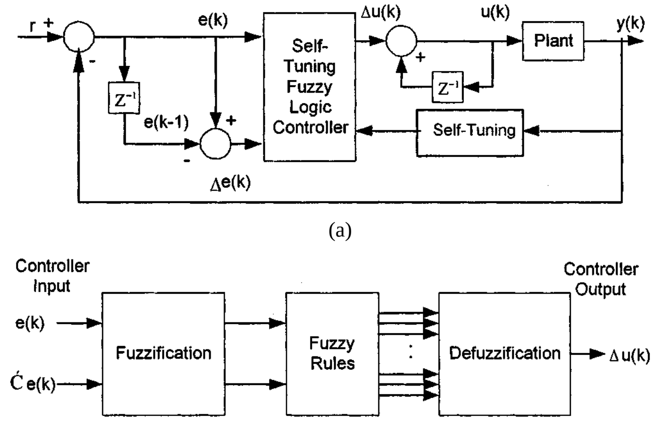
Using this well-known time-delay compensation technique on a simple first-order plant in an industry standard PID controller such as Bailey's Infi-90 single-loop controller is still not an easy task. The predictor parameters, including the plant gain, time constant, and time delay, in addition to the three PID parameters, must be determined. These six parameters used in a predictive compensator increase tuning and operational complexity on even the simplest plants. The additional complexity of the Smith predictor is the main reason industry still uses non-predictive PI or PID control for time delay using tuning methods such as Ziegler–Nichols method.

C. Fuzzy Logic Control

Fuzzy control is an appealing alternative to conventional control methods when systems follow some general operating characteristics and a detailed process understanding is unknown or traditional system models become overly complex [6]. The capability to qualitatively capture the attributes of a control system based on observable phenomena is a main feature of fuzzy control. These aspects of fuzzy control have been demonstrated in the research literature; see [13]–[16], [19], and [20], and commercial products from vendors like Reliance Electric and Omron. The ability of fuzzy logic to capture system dynamics qualitatively, and execute this qualitative idea in a real-time situation, is an attractive feature for temperature control systems.

Of course, fuzzy logic control has its own limitations. The analytical study of fuzzy logic is still trailing its implementation and much work is still ahead, particularly in the area of stability and performance analysis. Furthermore, as solutions to practical problems, fuzzy logic control design is problem dependent and the adaptation of an existing FLC to a different control problem is not straightforward. The available design tools, such as the Fuzzy Toolbox provided by The MathWorks Inc., Natick, MA, generally require further improvements before they become acceptable to control engineers.

In this paper, the validity of fuzzy logic control as an alternative approach in temperature control applications is investigated.



(a)

(b)

Fig. 3. Fuzzy logic control system. (a) Closed-loop FLC system. (b) Structure of a fuzzy controller.

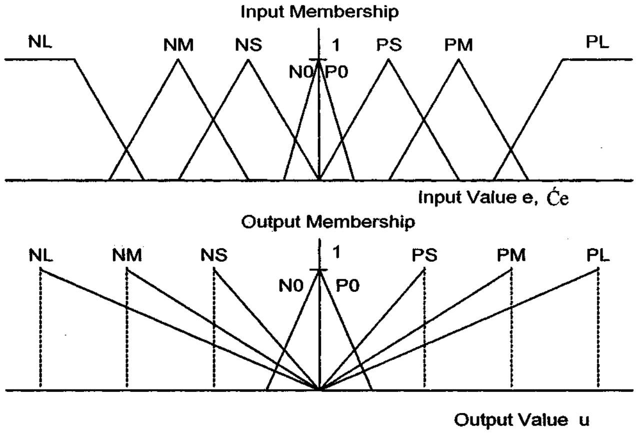


Fig. 4. Fuzzy membership functions.

II. Fuzzy Logic Control Design

The FLC developed here is a two-input single-output controller. The two inputs are the deviation from set-point *error* $e(k)$ and *error rate* $\Delta e(k)$. The FLC is implemented in a discrete-time form using a zero-order hold as shown in Fig. 3(a). The operational structure of the fuzzy controller is shown in Fig. 3(b).

A. Fuzzification/Defuzzification

Fuzzification and defuzzification involve mapping the fuzzy variables of interest to "crisp" numbers used by the control system. Fuzzification translates a numeric value for the error $e(k)$ or error rate $\Delta e(k)$ into a linguistic value such as positive large with a membership grade. Defuzzification takes the fuzzy output of the rules and generates a "crisp" numeric value used as the control input to the plant.

The FLC membership functions are defined over the range of input and output variable values and linguistically describe the variable's universe of discourse as shown in Fig. 4. The triangular input membership functions for the linguistic labels zero, small, medium, and large had their membership tuning center values at 0, 0.2, 0.35, and 0.6, respectively. The universe of discourse for both e and Δe may be normalized from -1 to 1, if necessary. The left and right halves of the triangle membership functions for each linguistic label were chosen to provide membership overlap with adjacent membership functions. The straight-line output membership functions for the labels zero, small, medium, and large are defined as shown in Fig. 4, with end points corresponding to 10%, 30%, 70%, and 100% of the maximum output, respectively. Both the input and output variables membership functions are symmetric with respect to the origin.

Selection of the number of membership functions and their initial values is based on process knowledge and intuition. The main idea is to define partitions over the plant operating regions that will adequately represent the process variables.

Once the input variables are fuzzified and run through the fuzzy rule base, which is discussed below, the output of the rules are then aggregated and defuzzified. Aggregation of the results of fuzzy rules takes the logical sum of all the output fuzzy sets. Then, a numerical control signal is generated. A typical formula for this purpose is the so-called centroid method [6], where the control signal is calculated as

$$\Delta u(k) = \frac{\sum_{i=1}^{n} F_i S_i}{\sum_{i=1}^{n} F_i} \tag{1}$$

where F_i is the membership grade and S_i is the membership function singleton position. This method is used in our simulation study in Section III. For the industrial implementation shown in Section V, however, the smallest of maximum (SOM) defuzzification method is used where the control signal is obtained as

$$\Delta u(k) = \min_i \left(\max_i (F_i) S_i \right). \tag{2}$$

This is because the SOM defuzzification is less computationally intensive than the centroid calculation.

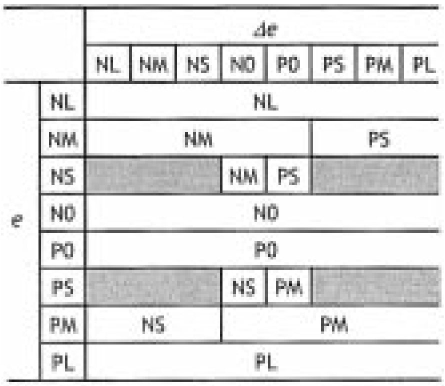
B. Rule Development

Our rule development strategy for systems with time delay is to regulate the overall loop gain to achieve a desired step response. The output of the FLC is based on the current input $e(k)$ and $\Delta e(k)$ without any knowledge of the previous input and output data or any form of model predictor. The main idea is that if the FLC is not designed with specific knowledge of the mathematical model of the plant, it will not be dependent on it. The rules developed in this paper are able to compensate for varying time delays online by tuning the FLC output membership functions based on system performance.

The FLC's rules are developed based on the understanding of how a conventional controller works for a system with a fixed time delay. The rules are separated into two layers: the first layer of FLC rules mimics what a simple PID controller would do when the time delay is fixed and known; the second rule layer deals with the problem when the time delay is unknown and varying.

In developing the first-layer rules, consider the first-order plant $G(s)e^{-s\tau}$, where $G(s) = a/(s+a)$. In the PID design, the following assumptions are made.

TABLE I
FLC CONTROL RULES



		Δe							
		NL	NM	NS	NO	PO	PS	PM	PL
e	NL	NL							
	NM	NM					PS		
	NS	▨	▨	▨	NM	PS	▨	▨	▨
	NO	NO							
	PO	PO							
	PS	▨	▨	▨	NS	PM	▨	▨	▨
	PM	NS			PM				
	PL	PL							

Shaded Areas Represent Zero Control Action

- The time delay τ is known.
- The rise time t_τ or, equivalently, the location of the pole is known.
- t_τ is significantly smaller than τ.
- The sampling interval is T_s.

The conventional PI-type controller in incremental form is given by

$$u(k) = u(k-1) + \Delta u(k) \tag{3}$$

where $\Delta u(k) = f(e, \Delta e)$ is computed by a discrete-time PI algorithm. This control algorithm was applied to a first-order plant with delay. Initial tuning of PI parameters was carried out by using the Ziegler–Nichols method. The step response obtained has about a 20% overshoot for a fixed time delay.

Next, a fuzzy logic control law was set up where $\Delta u(k) = F(e, \Delta e)$, the output of the FLC for the kth sampling interval, replaces $f(e, \Delta e)$ in the incremental controller described in (3). The rules and membership functions of the FLC were developed using an intuitive understanding of what a PI controller does for a fixed delay on a first-order system. They generalized what a PI controller does for each combination of e and Δe in 12 rules, as shown in Table I.

The output from each rule can be treated as a fuzzy singleton. The FLC control action is the combination of the output of each rule using the weighted average defuzzification method and can be viewed as the center of gravity of the fuzzy set of output singletons.

C. Tuning of Membership Functions in Design Stage

Since there is little established theoretical guidance, the tuning of rules and membership functions in the design stage is largely an iterative process based on intuition. The membership functions were tuned subject to the stability criteria derived later in Section IV, based on observations of system performance, such as rise time, overshoot, and steady-state error.

The number of membership functions can vary to provide the resolution needed. Note that the number of rules can grow exponentially as the number of input membership functions increases. The input membership functions for e and Δe generate 64 combinations which can be grouped into 12 regions corresponding to each rule in Table I.

The center and slopes of the input membership functions in each region is adjusted so that the corresponding rule provides an appropriate control action. In the case when two or more rules are fired at the same time, the dominant rule, that is, the rule corresponding to the high membership grade, is tuned first. Modifying the output membership function adjusts the rules contribution relative to the output universe of discourse. Once input membership rule tuning is completed, fine tuning of the output membership functions is performed to achieve the desired performance.

Although this FLC is constructed based on the assumption that the time delay is fixed and known, the only element of the controller that is a function of the delay is the universe of discourse for the output. It is shown below that, with some adjustment and extra rules, the FLC can be made to adapt to an unknown nature or change in delay.

D. Self-Tuning

The FLC structure presented above can be directly modified to compensate for changes in the plant dynamics and variable time delays by adding a second layer of self-tuning rules to the FLC. In the case of varying time delay, the FLC gain must be adjusted to offset the effects of the changes in delay. It was observed that the maximum gain or control action is inversely proportional to the time delay. Therefore, if the delay increases, we should decrease the FLC gain to reduce the control action, and vice versa. Based on this relationship, the system performance can be monitored by a second layer of rules that adapts the output membership functions of the first layer of rules to improve the performance of the fuzzy controller.

Consider an output membership function tuned for a nominal delay. When the true system time delay is larger than the nominal delay, the control action determined by the nominal delay causes the control output to be too large for the true system. This condition effectively increases the controller gain, and as the difference between the true and nominal delay becomes large, system stability problems could arise. Conversely, when the true delay is smaller than the nominal delay, the controller gain will be too small and the system becomes sluggish.

The output membership functions (see Fig. 4) of the FLC are defined in terms of the maximum control action. A viable mechanism to compensate for a varying time delay is to adjust the size of the control action under the assumption that the number of control rules remains fixed and the linguistic control strategy is valid for different values of time delay. These conditions are reasonable given that the plant parameters are known and that the control strategy developed is based on a plant with delay.

To adjust the FLC online for systems with varying time delay, a second layer of six rules was added as an adaptation mechanism to modify the output membership function used by the first layer rules with a scaling factor. This effectively changes the FLC control output universe of discourse (i.e., the maximum control action) based on system performance. These rules adjust the FLC output based on rise time and overshoot. The overshoot is monitored and classified as large (L), medium (M), and

TABLE II
FLC OUTPUT ADJUSTMENT

Rise Time Rules *If* Tracking *is* L_1 *then* Adjust *is* L_2		**Overshoot Rules** *If* Overshoot *is* L_3 *then* Adjust *is* L_4	
L_1	L_2	L_3	L_4
SS	PS	L	NL
MS	PM	M	NM
VS	PL	S	NS

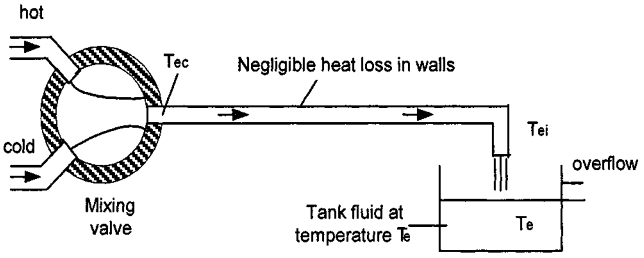


Fig. 5. Tank temperature control.

small (S). It is observed that changes in overshoot are indicative of a change in time delay. A longer delay results in a larger overshoot. Such effects can be alleviated by reducing the output scaling factor appropriately. Rise-time performance is classified as Very Slow (VS), Medium Slow (MS), and Slightly Slow (SS), and an increase in the output scaling factor can help to speed up the response.

The design strategy for the second layer of rules is based on two different aspects of tracking performance, i.e., rise time and overshoot calculated from $(e, \Delta e)$. The second layer rules are listed in Table II. They monitor the plant response and reduce or increase the FLC output universe of discourse. The fuzzy membership functions are defined using a membership configuration similar to the control strategy in Fig. 3. The adjustment rules perform two actions; they reduce the FLC gain when the plant is significantly overshooting the desired response, and increase the gain when rise time performance is slow.

Remark: A unique fuzzy control system is presented in this section. Although a PI controller is used as a guideline for setting up the FLC, it by no means limits its ability to perform more complicated tasks. Similar approaches can be used to set up an FLC that mimics more complex controllers. The emphasis here, however, is to deal with unknown dynamics and variable time-delay problems which we have difficulty with using analytical approaches. In particular, the self-tuning capability demonstrated in the proposed FLC design, although limited to a narrow class of problems with large set-point changes, shows the potential of incorporating human intelligence into such a control strategy.

III. SOFTWARE SIMULATION

The FLC developed above was simulated for the tank temperature control system shown in Fig. 5. The temperature of the tank fluid with constant flow rates in and out is to be controlled by adjusting the temperature of the incoming fluid. The incoming fluid temperature is determined by a mixing valve which controls the ratio of hot and cold fluid in the supply line to the tank. The distance between the mixing valve and the supply line discharge to the tank illustrates the classic material transport delay in pipes. The temperature/pressure of the fluids will also affect the delay.

The transfer function for the tank temperature control problem in Fig. 5 is given by

$$G(s) = \frac{T_e(s)}{T_{ec}(s)} = \frac{e^{-s\tau}}{s/a + 1} \tag{4}$$

where T_e = tank temperature, T_{ec} = temperature at exit of mixing valve, τ = time delay for material transport in the pipe $a = \dot{m}/M$, $\dot{m}$ = mass flow rate ($= \dot{m}_{in} = \dot{m}_{\text{out}}$), and M = fluid mass contained in the tank.

A. Simulation Results

The FLC was applied to the plant described in (4) with $a = 1$. Assuming the hot and cold supply enters the mixing valve at a constant pressure, the time delay from the material transport will also be constant. Conversely, if the hot and cold supply pressure is varying, the transport delay will also vary. The variable time-delay aspects of this system are investigated in the following simulations.

The simulation results are obtained using an 18-rule FLC, the 12 first-layer rules in Table I provide the control strategy, and the six second-layer rules in Table II adjust the control output membership function universe of discourse based on the system performance. For comparison purposes, simulation plots include a conventional PID controller, an SPC, and the fuzzy algorithm. The PID, SPC, and FLC were tuned on the plant with a 10-s time delay with the response shown in the top plot of Fig. 6. As expected, the SPC has the fastest response in the presence of an accurate plant model and a known time delay, but the PID and FLC provide good performance in terms of rise time and overshoot in the absence of a prediction mechanism. The middle and bottom plot of Fig. 6 shows how the controllers react as the true system time delay increases from the nominal 10-s delay used to tune the controllers. The FLC algorithm adapts quickly to longer time delays and provides a stable response while the PID controller drives the system unstable and the SPC oscillates around a final value due to the mismatch error generated by the inaccurate time-delay parameter used in the plant model.

From the simulations, clearly, the SPC provides the best response with an accurate model of the plant and delay. In the presence of an unknown or possibly varying time delay, the proposed FLC shows a significant improvement in maintaining performance and preserving stability over standard SPC and PID methods. Note that it is assumed here that the delay is unknown and time varying. If this is not the case, then perhaps an "adaptive" PID with Smith predictor can be used where the delay is estimated online, as pointed out by an anonymous reviewer.

Remark: The purpose of this simulation study is to show that although all three methods provide adequate performance at the nominal delay, the stability problem arises in PID and SPC when the delay gets longer. The insight of this stability robustness of FLC is given below.

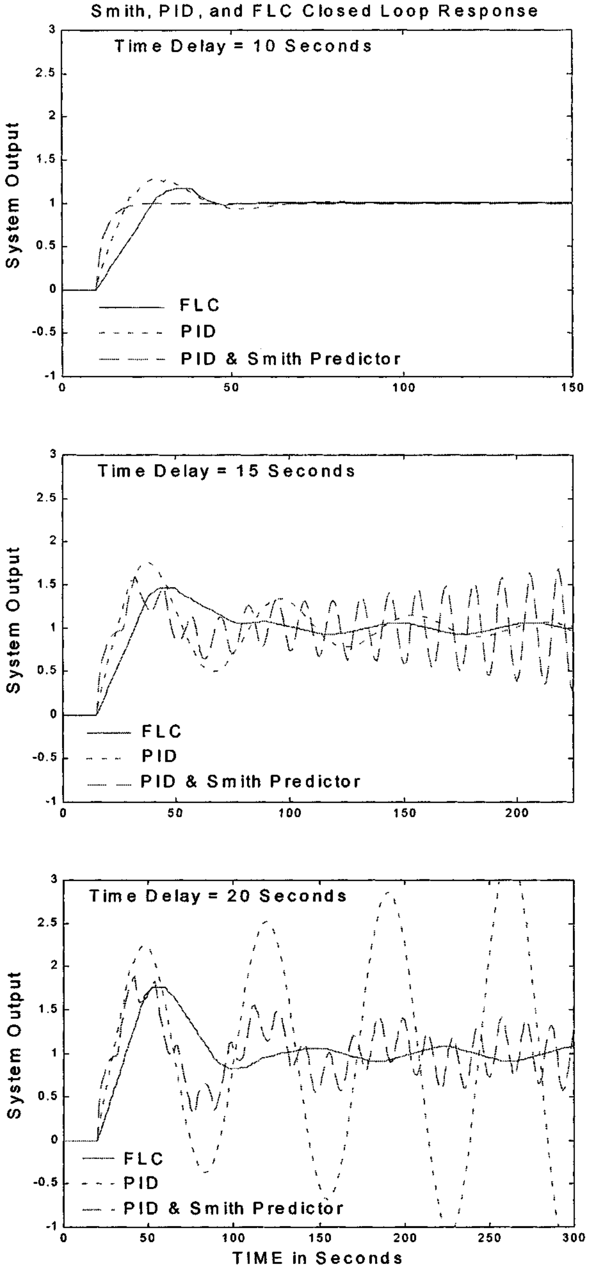


Fig. 6. PID, SPC, and FLC comparison.

IV. Stability Analysis

Most proposed FLCs in the literature do not have any stability proof because of the difficulty in analysis. However, for the FLC to be considered as a serious contender in industrial control design, a measure of stability or a certain degree of safety must be provided. Noting that the FLC can be viewed as a nonlinear time-varying controller, the stability issue is addressed below.

Consider a single-input–single-output(SISO) fuzzy logic control system where the FLC control law is given as $\varphi(e)$, where $\varphi(e)$ is a memoryless nonlinear function of e. The FLC developed above can be viewed as a nonlinear integral controller with a variable gain. We are interested in developing constraints on $\varphi(e)$ such that the closed-loop system is globally stable. For the sake of convenience, the FLC system in continuous time, shown in Fig. 7, is used for the analysis.

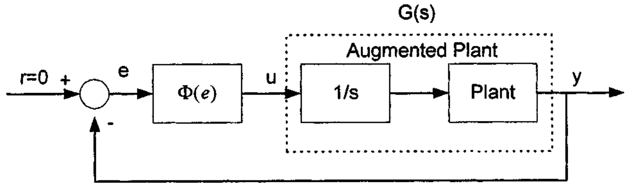


Fig. 7. System structure for stability analysis.

Assume that the state-space representation of the augmented plant is given by

$$\dot{x} = Ax + B\phi(e) \quad y = Cx \tag{5}$$

where A is Hurwitz, and $[A\ B\ C]$ is a minimal realization of $G(s)$. Note that Fig. 7 and the system equations (5) describe a classical nonlinear stability problem. Next, the Popov Method is used to derive stability conditions for the proposed FLC which can be used as a guideline to help set up the fuzzy controller.

The Popov Method states that a system described by (5) is absolutely stable for all nonlinearities $\varphi(e) \in (0, k)$ if there exists a strictly positive number α such that

$$\forall\,\omega \geq 0 \qquad Re[(1 + j\omega)\,G(j\omega)] + \frac{1}{k} \geq 0 \tag{6}$$

the origin is globally asymptotically stable. The Popov Method provides the stability guarantee of (6) using a quadratic Lyapunov function. Therefore, for a strictly positive α, a bound on k can be found to ensure the derivative of the Lyapunov function is negative and the system in (5) is absolutely stable. More details may be found in [25] and [26].

To carry out the stability analysis, the first-order Padé approximation for $e^{-s\tau}$ is used, which is given by $e^{-s\tau} \approx (1 - s\tau/2)/(1+s\tau/2)$. Using the Popov Method, we will determine the sector condition on $\varphi(e)$ such that the system is absolutely stable. Rewriting the augmented plant with the Padé approximation in the form $G(j\omega) = G_1(\omega) + jG_2(\omega)$, we have

$$\begin{aligned} &G(j\omega) \\ &\quad = \frac{1}{j\omega}\,\frac{a}{(j\omega + a)}\,\frac{(1 - j\omega\tau/2)}{(1 + j\omega\tau/2)} \\ &\quad = \frac{-a\omega(1{+}a\tau{-}\omega^2\tau^2/4){-}ja(a{-}\omega^2\tau/2(2{+}a\tau/2))}{\omega[a^2 + \omega^2(1 + a^2\tau^2/4) + \omega^4\tau^2/4]}. \end{aligned} \tag{7}$$

Substituting (7) into (6), the Popov inequality becomes

$$\begin{aligned} &\frac{1}{k}\,\frac{1}{a}\,(a^2 + \omega^2(1 + a^2\tau^2/4) + \omega^4\tau^2/4) \\ &\qquad > 1 + a\tau - a\alpha + \alpha\omega^2\tau/2(2 + a\tau/2) - \omega^2\tau^2/4 \end{aligned} \tag{8}$$

through straightforward, but rather tedious manipulations, (8) is reduced to

$$\begin{cases} \dfrac{1}{k} > \tau - \alpha + \dfrac{1}{a} = x_1 \\ \dfrac{1}{k} > \dfrac{a\tau/2[\alpha(2 + a\tau/2) - \tau/2]}{1 + a^2\tau^2/4} = x_2. \end{cases} \tag{9}$$

That is, for a first-order system with delay described by (5) and Fig. 7, the sector condition $\varphi(e) \in (0, k)$ to maintain absolute stability for a time delay τ is given by

$$\frac{1}{k} > \max(x_1, x_2). \tag{10}$$

Note that the stability constraint in (10) is a function of τ, a, and α, where τ and a are parameters of the plant and α is any positive real number. For example, if we let $\alpha = 1/a$, then from (9) $x_1 = \tau$, and $x_2 = \tau/(1 + a^2\tau^2/4)$. Using (10) now gives the value $k < 1/\tau$ as the maximum gain to guarantee stability. In general, if the range of τ and a are known, a maximum bound on k can be determined by varying α iteratively and determining $\min_{\tau,\,\alpha,\,a}\, \max(x_1, x_2))$.

Remark: The bound in (10) guarantees the asymptotic stability of the system. That is, the derivative of a certain quadratic Lyapunov function [15], [16] is strictly negative. Because of the conservative nature of the Lyapunov approach, (10) may be too restrictive for practical implementation. Our approach is to enforce the sector condition as a stability safeguard in the large error region, while in the low to intermediate error range the FLC is tuned to provide good tracking performance. This strategy proves to be quite successful in the simulations shown in Section III. The tradeoff is that the origin of the system is not necessarily asymptotically stable. However, the error is guaranteed to be bounded. The proof is rather straightforward. As the error becomes “large,” as defined in the membership function, the corresponding FLC gain, designed subject to (10), forces the derivative of the Lyapunov function to be strictly negative and, therefore, the error to reduce.

V. An Industrial Application

The fuzzy logic control temperature control scheme is further tested in an industrial application where several components in a machine have to be temperature regulated. These components are of different thermo mass and may be regulated at different temperatures. Currently, a separate PID controller is tuned for each component at each temperature set point, which is quite labor intensive. Furthermore, the PID parameters need frequent adjustments due to the changes in operating conditions. The goal of fuzzy control is to replace this set of PID controllers with one self-tuning fuzzy controller and to eliminate the needs for further tuning, once the machine is in operation.

A. Hardware Setup

A generic diagram of the process that applies to all components of the machine is shown in Fig. 8.

This heating equipment of high-temperature liquids has a large thick metal plate on the underside of the tank between the bottom and the inside of the tank, as shown in Fig. 8. There are many variations in the dynamics of the system. The thermo capacity is proportional to the size of the tank, which is quite different from one component to another. The time delay in the system is quite sensitive to the placement of the RTD. The heater can be found to be undersized or oversized.

The heater on and off are controlled by a 24-V pulsewidth modulated (PWM) signal applied to the SSR. The digital control system includes a local microprocessor as well as a host processing system in a configuration known as *distributed control*, as shown in Fig. 9.

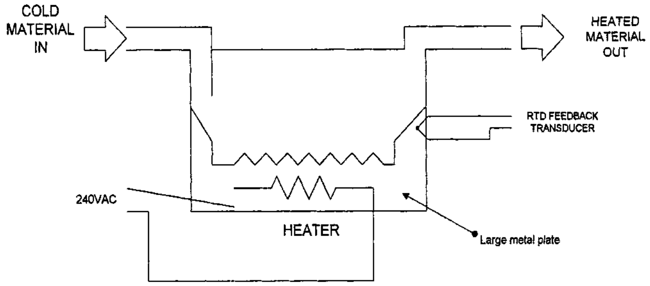


Fig. 8. An industrial temperature control application.

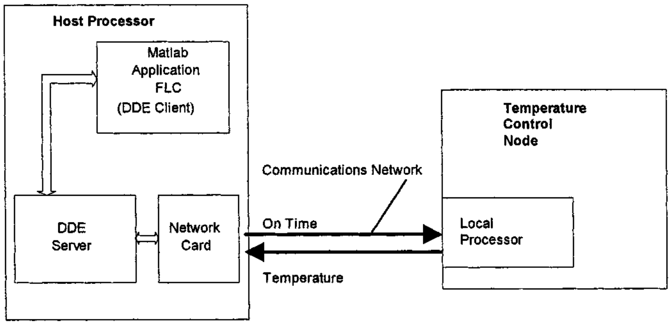


Fig. 9. Distributed control system configuration.

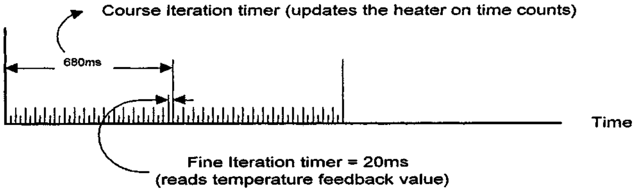

Course Iteration timer (updates the heater on time counts)
680ms
Time
Fine Iteration timer = 20ms
(reads temperature feedback value)

Fig. 10. Local processor timing.

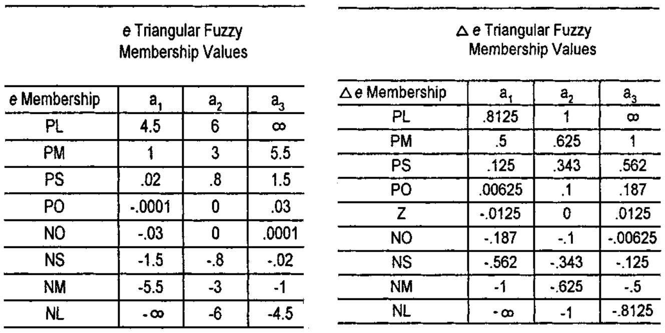
The host processor receives the process temperatures while supplying the appropriate heater *on time* to the temperature control node. The local microprocessor in the temperature control node (TCN) executes the application code which defines the local timing for reading the feedback temperatures and turning on the SSRs (see Fig. 10).

B. FLC Adjustments

The self-tuning FLC, shown in detail in Section II, was applied to this industrial problem. Because this industrial temperature process is quite different from the one studied previously in Section III, a few adjustments were made in the FLC parameters. In particular, the membership functions are shown in Table III and the rule base in Table IV. Note that Table III is not normalized on purpose to show the resolution of the FLC controller with the unit of °F.

This FLC design is quite intuitive and transparent to the user. The heater *on time* is at maximum until the temperature response is within 10% of the set point. It is at this time or, more precisely, in this error region, in which the FLC begins to decrement the heater *on time*, driving the temperature to its desired set point. The rule base applies the appropriate control action depending

TABLE III
MEMBERSHIP FUNCTIONS



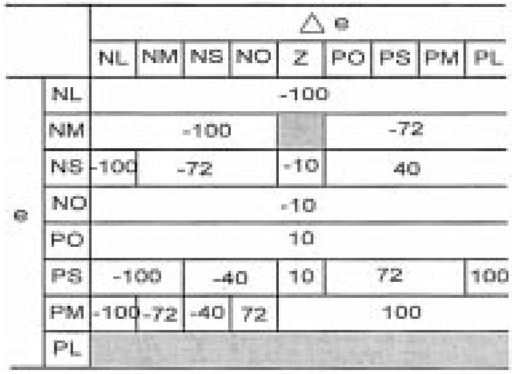
e Triangular Fuzzy Membership Values

e Membership	a_1	a_2	a_3
PL	4.5	6	∞
PM	1	3	5.5
PS	.02	.8	1.5
PO	-.0001	0	.03
NO	-.03	0	.0001
NS	-1.5	-.8	-.02
NM	-5.5	-3	-1
NL	-∞	-6	-4.5

Δ e Triangular Fuzzy Membership Values

Δ e Membership	a_1	a_2	a_3
PL	.8125	1	∞
PM	.5	.625	1
PS	.125	.343	.562
PO	.00625	.1	.187
Z	-.0125	0	.0125
NO	-.187	-.1	-.00625
NS	-.562	-.343	-.125
NM	-1	-.625	-.5
NL	-∞	-1	-.8125

TABLE IV
FLC RULE BASE AS PERCENTAGE OF *ON TIME* ADJUSTMENT



		Δ e								
		NL	NM	NS	NO	Z	PO	PS	PM	PL
e	NL	-100								
	NM	-100				▨	-72			
	NS	-100	-72			-10	40			
	NO	-10								
	PO	10								
	PS	-100		-40		10	72			100
	PM	-100	-72	-40	72	100				
	PL	▨								

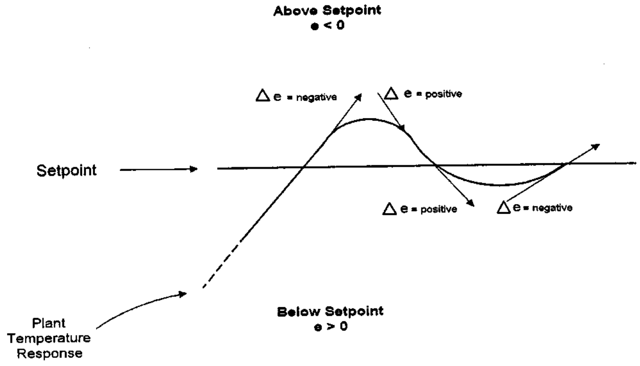


Fig. 11. Polarity of the temperature vector.

on how far the temperature response is from the set point, and how fast the response is moving toward the set point (i.e., error and error rate). Fig. 11 illustrates the effectiveness of having direct control over the error and change of error in driving the temperature to a prescribed set point.

The polarity and magnitudes of e and Δe, for example, can be thought of as *temperature vectors* having a defined direction and velocity at every kth sampling interval. When the error has a positive polarity, the response is below the set point and when the error is negative, the response is above the set point (or in an overshoot state); see Fig. 11. Knowing the relative position from the set point, an appropriate control action is applied based on the velocity of the plant converging to the set point. When Δe is negative and the magnitude is large, for example, the response is moving very quickly toward the set point. Thus, a large negative control action is applied to slow the response and prevent overshoot (see rules in Table IV). When the plant is in an overshoot state (i.e., $e < 0$), the FLC responds to this new magnitude and direction of Δe by applying a negative control action to drive the plant back to the set point. As the plant begins to approach the set point, the polarity of Δe becomes positive and the FLC applies a positive small (PS) control action to prevent the response from approaching too quickly, thus preventing an undershoot state of the response. Furthermore, additional rules were added to make the controller automatically adjust itself to the different dynamics of the processes.

C. Self-Tuning Rule Modifications

In addition to the self-tuning rules in Table II, which were used to deal with longer than expected times delays, additional rules were added to cope with unique problems in this industrial problem. Specifically, it was observed that the plant behaves quite differently at low (<160 °F) and high (>300 °F) temperatures. The transport delay seems to be larger at higher set point and, therefore, the output universe of discourse is compressed accordingly to prevent overshoot, in the same manner of the overshoot rules in Table II. Furthermore, the different thermo masses make it especially difficult for a single controller to handle. A single self-tuning rule is added to resolve this problem. It adjusts the output universe of discourse based on the velocity of temperature change initially during the open-loop control stage where the heater is full on. Note that this velocity reflects the equivalent thermo mass in the plant. If the velocity is observed to be large, which means the thermo mass is small, then the universe of discourse is again compressed to effectively reduce the gain and prevent overshoot, and vice versa. The effect of this single adjustment made a dramatic impact on our hardware test.

D. Hardware Test Results

The proposed fuzzy control algorithm was compared experimentally with the existing PID control used in industry. In this application, it is important to prevent overshoots which seriously affect the quality of the product. It is also desirable to have a smooth control signal that does not require excessive on and off actions in the heater.

The results obtained by actually controlling the process in its industrial setting are shown in Figs. 12–15. The top portion of each figure is a comparison of the PID versus fuzzy temperature response, while the bottom portion is their respective heater *on times*.

1) Basis of Comparison and the Significance of Improvements: While the basis for comparison is primarily stability in the simulation study, the performance criteria for the industrial application are quite different. In particular, it is important to prevent overshoots which seriously affect the quality of the product. It is also desirable to have a smooth control signal that does not require excessive on and off actions in the heater. The oscillatory control signal itself indicates poor stability margin and robustness and, therefore, should be avoided as much as possible. Furthermore, the controller must also be capable of handling the changes in operating conditions such as set-point

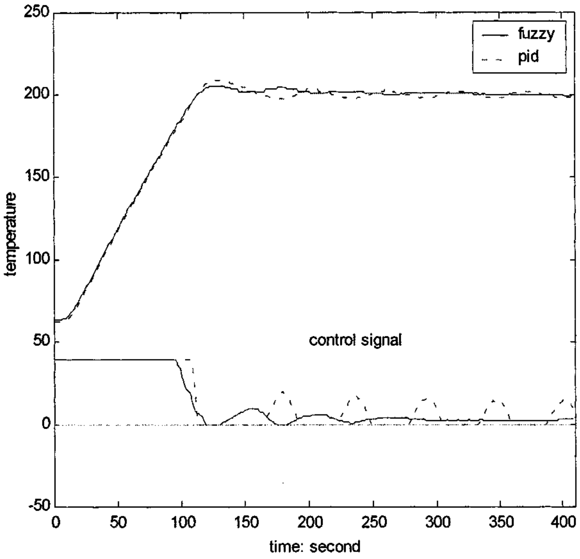


Fig. 12. PID and fuzzy controller at high temperature.

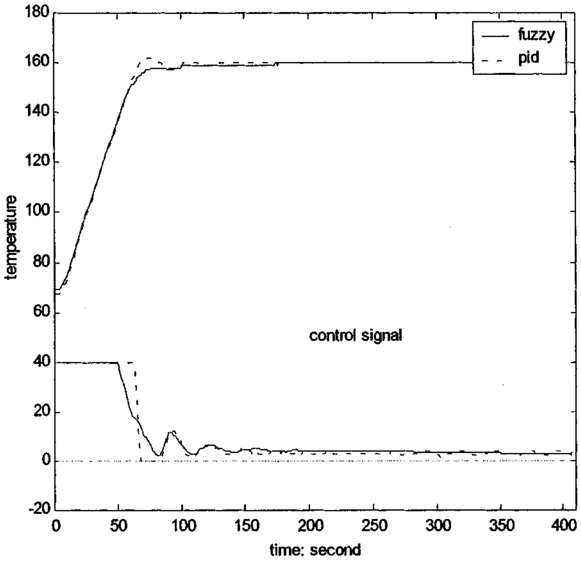


Fig. 14. PID and fuzzy comparison with time delay.

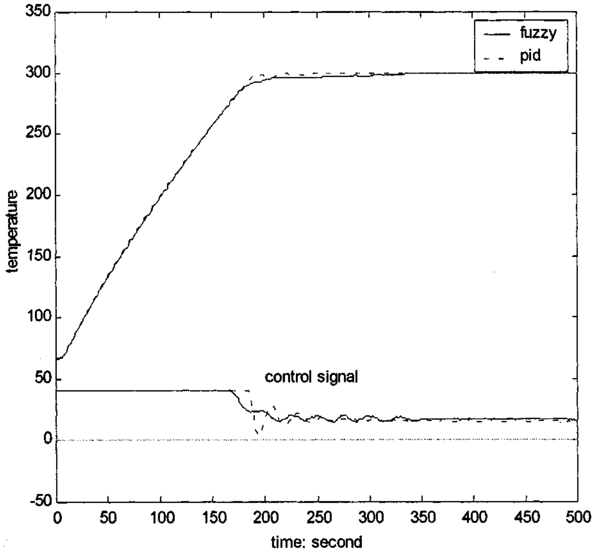


Fig. 13. PID and fuzzy comparison at low temperature.

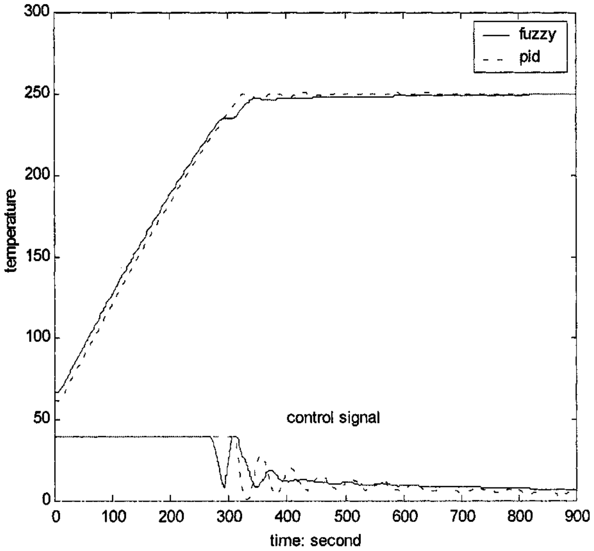


Fig. 15. PID and fuzzy comparison with large thermo mass.

change, unknown delay variations caused by different placements of RTD, changes in thermo mass, etc. The PID and FLC are compared along these lines, although it is hard to find one performance index to cover all of them.

The performance of the standard industrial PID in Figs. 12–14 require retuning, which is not the case for FLC. As a matter of fact, a single FLC was able to deal with all of the above scenarios with no tuning whatsoever. This unique feature of the FLC was well received by our industrial partner.

2) Detailed Comparisons: Figs. 12 and 13 show the comparison of the PID and FLC outputs at 300 °F and 160 °F, respectively. Both controllers performed well at 300 °F. Although the PID was tuned with its optimal gains, the FLC reduces the heater *on time* more gradually as the plant approaches the setpoint. This is a characteristic design of the FLC in that transitions to the set point are made more smoothly or conservatively for systems which cannot afford much overshoot. At 160 °F, the PID is too aggressive and the response overshoots. The FLC, on the other hand, automatically compensated for the low set point by providing a conservative control action, thus producing a smoother response (Fig. 13). It should be noted that the FLC converges to its final *on-time* value, whereas the PID on time is still somewhat oscillatory at the steady-state temperatures.

Fig. 14 shows the result of physically displacing the RTD on a more remote location, thus inducing a larger time delay. The setpoint is 200 °F. The PID controller oscillated, which indicates poor stability margin, and had difficulty converging to the set point. The FLC, in comparison, produced a converging on-time count and a temperature response accurate to within 1 °F of the set point. Note that the oscillation is much smaller than that of the simulation study in Section III for PID and FLC because the time delay here is only a fraction of the thermo time constant of the plant. The difference in the performance of the two controllers is nonetheless significant.

The temperature response shown in Fig. 15 illustrates how the FLC compensates thermo mass change. The FLC monitors how fast the plant approaches the desired setpoint (i.e., Δe). If the magnitude of Δe is above and beyond the normal magnitude for this class of temperature control problems, the FLC aggressively adjusts the control to avoid overshoot and instabilities.

Remarks: In summary, a practical FLC has been developed, with marked advantages over the PID controller. The FLC utilizes self-tuning mechanisms to effectively overcome issues not easily addressed in the PID controller. The "self-tuning" mechanisms of the FLC are not "all encompassing," but compensate for issues tested in this research. The flexibility associated with the FLC, however, would easily allow the controller to be expanded into a full-range self-tuning control, should it become necessary.

A serious drawback of FLC is in its complexity. Even though every effort was made to keep the number of rules and membership functions to the minimum, the FLC presented here is quite more complex than its PID counterpart. This must be weighed in the design selection carefully to see if it is cost effective. It is likely that PID will still be dominant in many applications where the control problem is not as challenging as the ones studied in this paper.

VI. Conclusions

Unlike some fuzzy controllers with hundreds, or even thousands, of rules running on dedicated computer systems, a unique FLC using a small number of rules and straightforward implementation has been proposed to solve a class of temperature control problems with unknown dynamics or variable time delays commonly found in industry. Additionally, the FLC can be easily programmed into many currently available industrial process controllers. The FLC was first simulated on a tank temperature control problem with promising results. Then, it was applied to an entirely different industrial temperature apparatus. The results show significant improvement in maintaining performance and stability over a wide range of operating conditions. The FLC also exhibits robust performance for plants with significant variation in dynamics. The stability characteristics were investigated and a stability safeguard was derived.

Acknowledgment

The authors would like to express their appreciation to the anonymous reviewers who made numerous suggestions that helped to improve the manuscript.

References

[1] J. Dawson, "Fuzzy logic control of linear systems with variable time delay," M.S. thesis, Dept. Elect. Eng., Cleveland State Univ., Cleveland, OH, June 1994.

[2] T. A. Trauztch, "Self-tuning temperature control using fuzzy logic," M.S. thesis, Dept. Elect. Eng., Cleveland State Univ., Cleveland, OH, June 1996.

[3] D. J. Elliott, "Fuzzy logic positional servo motor control development platform," M.S. thesis, Dept. Elect. Eng., Cleveland State Univ., Cleveland, OH, June 1997.

[4] W. Nonnenmacher, "Fuzzy logic position control of a servo motor," M.S. thesis, Dept. Elect. Eng., Cleveland State Univ., Cleveland, OH, April 1997.

[5] N. Dhayagude, Z. Gao, and F. Mrad, "Fuzzy logic control of automated screw fastening," *J. Robot. Comput. Aided Manuf.*, vol. 12, no. 3, pp. 235–242, 1996.

[6] K. Passino and S. Yurkovich, *Fuzzy Control*. Reading, MA: Addison-Wesley, 1998.

[7] O. J. M. Smith, "A controller to overcome dead time," *ISA J.*, vol. 6, no. 2, pp. 28–33, Feb. 1959.

[8] P. S. Buckly, "Automatic control of processes with dead time," in *Proc. IFAC World Congr.*, Moscow, U.S.S.R., 1960, pp. 33–40.

[9] J. E. Marshall, *Control of Time Delay Systems*. Stevenage, U.K.: Peregrinus, 1979.

[10] Q. Brone and S. Harris, "Varying time delay estimation and self-tuning control," in *Proc. 1991 American Control Conf.*, vol. 2, pp. 1740–1741.

[11] K. Shu and J. Yan, "Robust stability of uncertain time delay systems and its stabilization by variable structure control," *Int. J. Control*, vol. 57, no. 1, pp. 237–246, 1993.

[12] G. P. Liu and H. Wang, "Adaptive controller for continuous-time systems with unknown varying time delay," in *Proc. 1991 IEE Int. Conf. Control*, vol. 2, pp. 1084–1088.

[13] C. C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller—Part 1," *IEEE Trans. Syst., Man, Cybern.*, vol. 20, pp. 404–418, Mar./Apr. 1990.

[14] C. C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller—Part 2," *IEEE Trans. Syst., Man, Cybern.*, vol. 20, pp. 419–435, Mar./Apr. 1990.

[15] P. J. King and E. H. Mamdani, "The application of fuzzy control systems to industrial processes," *Automatica*, vol. 11, pp. 235–242, 1977.

[16] S. Chiu, S. Chand, D. Moore, and A. Chaudhary, "Fuzzy logic for control of roll and moment for a flexible wing aircraft," *IEEE Contr. Syst. Mag.*, vol. 11, pp. 42–48, June 1991.

[17] J. A. Bernard, "Use of a rule-based system for process control," *IEEE Contr. Syst. Mag.*, vol. 8, pp. 3–13, Oct. 1988.

[18] J. Litt, "An expert system to perform on-line controller tuning," *IEEE Contr. Syst. Mag.*, vol. 11, pp. 18–23, Apr. 1991.

[19] S. Tzafestas and N. Papanikolopoulos, "Incremental fuzzy expert PID control," *IEEE Trans. Ind. Electron.*, vol. 37, pp. 365–371, Oct. 1990.

[20] P. Oliveria, P. Lima, and J. Sentierio, "Fuzzy supervision of direct controllers," in *Proc. 5th IEEE Int. Symp. Intelligent Control*, 1990, pp. 638–643.

[21] P. Isomursu and T. Rauma, "Self-tuning fuzzy logic controller for temperature control of superheated steam," in *Proc. IEEE Int. Conf. Fuzzy Systems*, vol. 3, 1994, pp. 1560–1563.

[22] Y. Wang and D. J. Birdwell, "Nonlinear PID-type controller utilizing fuzzy logic," in *Proc. IEEE IFAC Joint Symp. Computer Aided Control System Design*, 1994, pp. 89–94.

[23] K.-W. Li, B. Turksen, and K. Smith, "FPGA implementation of a self-tuned fuzzy controller," in *Proc. Biennial Conf. North American Fuzzy Information Processing Society—NAFIPS*, 1996, pp. 285–288.

[24] S.-Y. Oh and D.-J. Park, "Self-tuning fuzzy controller with variable universe of discourse," in *Proc. IEEE Int. Conf. Systems, Man, and Cybernetics*, vol. 3, 1995, pp. 2628–2632.

[25] J. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, NJ: Prentice-Hall, 1991.

[26] R. R. Mohler, *Nonlinear Systems: Volume I, Dynamics and Control*. Englewood Cliffs, NJ: Prentice-Hall, 1991.