ON LINE LOWER-ORDER MODELING USING FUZZY SYSTEMS

H. F. Ho, A. B. Rad*, Y. K. Wong, and W. L. Lo

The Hong Kong Polytechnic University
Department of Electrical Engineering
Hung Hom, Kowloon
Hong Kong SAR, China
E-mail: eeabrad@polyu.edu.hk

Abstract: In this paper, we present a novel on-line approximation technique to find the parameters of a First-Order plus Time Delay (FOPTD) model of higher-order systems using fuzzy reasoning. Based on the information obtained from the model, the parameters of a PID controller can be adjusted on-line. The performance of this algorithm is verified by simulation studies. The simulated examples demonstrate the feasibility and adaptive property of the proposed algorithm.

Keywords: Adaptive Control, fuzzy Logic, gradient descent

1. INTRODUCTION

Many systems are represented mathematically by high order dynamics. However, a lower-order model is sufficient for controller tuning (Ashworth, 1982). It is widely accepted that for the purpose of controller design, a First Order Plus Time Delay (FOPTD) model can approximate such systems adequately and hence facilitate controller design. In general, the parameters of this model, namely system gain, apparent time constant and apparent time delay can be used to tune a PID controller. There are many techniques to determine the parameters of FOPTD (Ziegler, 1942, Smith, 1967, 1997, Sundaresan, 1978). However, most of them are off-line approximating methods, for which the parameters are obtained from process reaction curve. In such cases, it is difficult to apply these methods to describe adequately the time-varying characteristic of the plant.

In recent years, there has been an unprecedented increase in applications of the so-called Soft-computing methodologies in identification and control of dynamic systems (Bha, 1990, Czogala, 1981, Lu, 1997, Narendra, 1990). Soft-computing methods are referred to techniques that employ fuzzy systems, neural networks and genetic algorithms either alone or in hybrid form.

In particular, fuzzy logic theory (Zadeh, 1965) has been the focus of much research in the areas of control and identification. Its integration with model-based systems theory has produced a unique approach entailing the human knowledge and heuristic methods with rigourous mathematical methods for stability and convergence analysis and several successful applications in control and identification have been reported (Chen, 1998, Wang, 1996). Whereas majority of earlier efforts was focused in fuzzy controllers, the emerging area of fuzzy identification has become very important in fuzzy system theory in the last decade (Sugeno, 1986, Babuska 1996). Fuzzy identification methods fall into three categories, linguistic fuzzy model (Wang, 1996), fuzzy relational modelling (Wang, 1997) and Takagi and Sugeno (TS) modeling (Sugeno, 1986). It is interesting to note that not much attention has been paid to reduced order modeling. This may be due to the fact that fuzzy logic systems are essentially model free approaches. This has motivated the authors to develop an on-line approximation method to determine the parameters of FOPTD using fuzzy systems.

This paper presents a simple and new approach to the on line lower-order model identification for unknown processes using fuzzy system. The idea is to integrate a fuzzy system with a model generator with known structure. The parameters learning task is performed using the gradient descent algorithm (Wang, 1997).

The rest of this paper is organized as follow: Section 2 is devoted to the idea of approximating a high-order system with a FOPTD model using fuzzy system. The proposed method combined with PID controller is derived in Section 3. In Section 4 simulations studies are presented. Finally, the paper is concluded in Section 5.

2. LOWER ORDER APPROXIMATION OF HIGHER-ORDER SYSTEMS WITH FUZZY SYSTEM

2.1 The On-line Approximating Approach

It is well known that high-order processes dynamic can be described with sufficient accuracy by a first order plus time delay model (Sundersan, 1978). Consider:

$$\frac{Y(s)}{U(s)} = e^{-sq}\,\frac{b_0 s^n + b_1 s^{n-1} + \ldots + b_m}{s^n + a_1 s^{n-1} + \ldots + 1} \approx \frac{K \cdot e^{-t \cdot s}}{Ts + 1} \qquad (1)$$

where K is the system gain, T is the dominant time constant, t is the apparent dead time and *Y(s)* and *U(s)* are the Laplace transformed output and input signals respectively. The proposed approach is conceptually simple and is realized by cascading a fuzzy system and model generator in parallel with the process to be identified as shown in Figure 1. The input signal *u(t)* is applied to the high-order system, the fuzzy system, and the FOPTD model generator at the same time. The fuzzy system has three parameters, namely, the gain *K*, the time constant *T*, the dead time t. These three parameters are fed to the first-order plus dead time model generator to get the output of the model. The error between the output of the plant and the output of the model is used to train the consequent part of the fuzzy system. The training process tends to force the output of the FOPTD model generator to approximate the output of the system. Thus, the inputs of the FOPTD model generator are the approximating parameters of the first-order representation of the high-order system. The output of the FOPTD model is expected to match the output of the high-order system after the model converges.

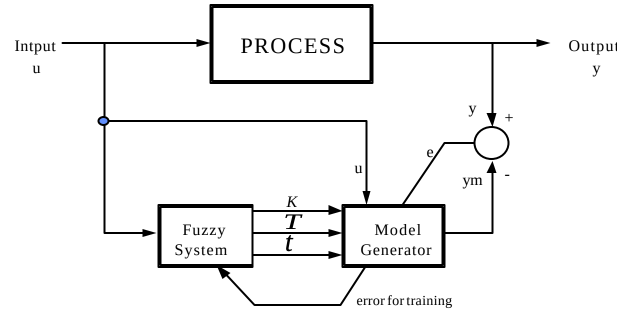

Fig. 1 Block Diagram of the proposed method

The transfer function of the FOPTD model generator is rewritten below:

$$\frac{Y_m(s)}{U(s)} = \frac{K \cdot e^{-t \cdot s}}{Ts + 1} \qquad (2)$$

2.2 The Fuzzy System Structure and Training Algorithm

In this paper, we apply the fuzzy identification techniques to obtain the process model directly. The *i*th rule of the fuzzy model is of the following form:

Rule i: If x_1 is A_{i1} and ····· and x_n is A_{in} then $\hat{p}_i$ is c_i

where $x \in R^n$ and $p_i \in R$ are the input vector (process input *u(t)*) and output value (estimated model parameters) of the fuzzy system respectively, A_{ij} ,*i=1,2.,m, j=1…n,* are the(1) fuzzy sets. Given the input data x, by using a singleton fuzzifier, product fuzzy inference and weighted average defuzzifier, the output value of the fuzzy system is inferred as follows:

$$\hat{p}_i(\mathbf{x}) = \frac{\sum_{i=1}^{m} c_i \prod_{j=1}^{n} m_{ij}(x_j)}{\sum_{i=1}^{m}\prod_{j=1}^{n} m_{ij}(x_j)} \qquad (3)$$

if we fix the m_{ij} *(Membership value of* $\mathbf{x}$ *for* A_{ij}*)* and view the c_i (*Consequence of Rule i*) as adjustable parameters, then equation (3) can be rewritten as:

$$\hat{p}_i(\mathbf{x}) = \mathbf{j}^T \mathbf{x}(\mathbf{x}) \qquad (4)$$

where $\mathbf{j} = (c_1, \cdots\cdots, c_m)^T, \mathbf{x}(\mathbf{x}) = (\mathbf{x}_1(\mathbf{x}), \cdots\cdots, \mathbf{x}_m(\mathbf{x}))^T$ is a regression vector defined as

$$x_i(\mathbf{x}) = \frac{\prod_{j=1}^{n} m_{ij}(x_j)}{\sum_{i=1}^{m}\prod_{j=1}^{n} m_{ij}(x_j)} \qquad (5)$$

To train the above fuzzy system, a direct learning the gradient descent algorithm (Wang, 1997) is employed. The consequent parameters are adjusted in each iteration is derived below. The error function E is define as:

$$E = \frac{1}{2}e^2 = \frac{1}{2}[y_m(t) - y(t)]^2 \tag{6}$$

where y and y_m are the output of the plant and the output at the FOPDT model at any time instant t. Within each time interval from t to $t+1$, the gradient descent algorithm is used to update the consequent parameters according to the following relationship:

$$c_i(t+1) = c_i(t) - \eta_i \cdot \frac{\partial E}{\partial c_i} \tag{7}$$

where η_i is the learning rate. Using the chain rule, one has

$$\begin{aligned}\frac{\partial E}{\partial c_i} &= \frac{\partial E}{\partial y_m(t)} \cdot \frac{\partial y_m(t)}{\partial p_i(t)} \cdot \frac{\partial p_i(t)}{\partial c_i} \\ &= \frac{\partial E}{\partial y_m(t)} \cdot \frac{\partial y_m(t)}{\partial p_i(t)} \cdot \frac{\prod_{j=1}^{n} \mu_{ij}(x_j)}{\sum_{i=1}^{m} \prod_{j=1}^{n} \mu_{ij}(x_j)} \\ &= \frac{\partial E}{\partial y_m(t)} \cdot \frac{\partial y_m(t)}{\partial p_i(t)} \cdot \xi_i(x)\end{aligned} \tag{8}$$

$\hat{\boldsymbol{p}} = [K\ T\ \tau]$ is a 3 1 input vector of the FOPDT model (the output of vector of the fuzzy system)

$$\frac{\partial y_m}{\partial \boldsymbol{p}} = \left[\frac{\partial y_m(t)}{\partial K}, \frac{\partial y_m(t)}{\partial T}, \frac{\partial y_m(t)}{\partial \tau}\right]^T \tag{9}$$

To find the partial derivatives of the output $y_m(t)$ of the model generator (FOPTD) w.r.t. gain (K), dominant time constant (T) and apparent dead-time (τ), respectively, please refer to Appendix A1.

$$\frac{\partial y_m(t)}{\partial T} = L^{-1}\left[\frac{-sKe^{-\tau s}}{(T \cdot s + 1)^2} U(s)\right] \tag{10}$$

$$\frac{\partial y_m(t)}{\partial \tau} = L^{-1}\left[\frac{-sKe^{-\tau s}}{T \cdot s + 1} U(s)\right] \tag{11}$$

$$\frac{\partial y_m(t)}{\partial K} = L^{-1}\left[\frac{e^{-\tau s}}{T \cdot s + 1} U(s)\right] \tag{12}$$

From equations (7) and (8), we can rewrite the update rule as follows:

$$c_i^1(t+1) = c_i^1(t) - \eta_1 \cdot e(t) \cdot \frac{\partial y_m(t)}{\partial K} \xi_i(\boldsymbol{x}) \tag{13}$$

$$c_i^2(t+1) = c_i^2(t) - \eta_2 \cdot e(t) \cdot \frac{\partial y_m(t)}{\partial \tau} \xi_i(\boldsymbol{x}) \tag{14}$$

$$c_i^3(t+1) = c_i^3(t) - \eta_3 \cdot e(t) \cdot \frac{\partial y_m(t)}{\partial T} \xi_i(\boldsymbol{x}) \tag{15}$$

where *1*,*2* and *3* is the indication of the FOPTD model parameter gain, time constant and time delay and η_1 ,η_2 and η_3 are the learning rate of each fuzzy sub-system respectively. Figure 2 show the membership function of each sub-system .We have used the value of a equal to 1 in the following simulations. Therefore the fuzzy system consists of three fuzzy sub-systems as shown in Figure 2 and the output value can be obtained from equation 16.

$$\hat{p}_1(\boldsymbol{x}) = \varphi_1^T x(\boldsymbol{x}),\ \hat{p}_2(\boldsymbol{x}) = \varphi_2^T x(\boldsymbol{x}),\ \hat{p}_3(\boldsymbol{x}) = \varphi_3^T x(\boldsymbol{x}) \tag{16}$$

where $\varphi_3 = (c_1^3, \cdots\cdots, c_m^3)^T$, $\varphi_2 = (c_1^2, \cdots\cdots, c_m^2)^T$ and $\varphi_1 = (c_1^1, \cdots\cdots, c_m^1)^T$ are the regression vector of each sub-system as given in equation (5). The three fuzzy sub-systems have similar structures. In this paper, 2 fuzzy rules are used for each fuzzy sub-system.

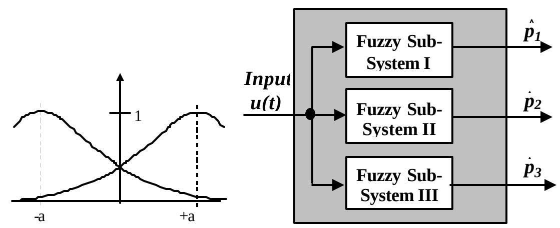

Fig 2 Fuzzy Subsystems

3. ON LINE PID TUNING METHOD USING FUZZY SYSTEM

In order to show the effectiveness of the proposed method, we combine the fuzzy algorithm with a standard PID controller to make an adaptive control algorithm. The control structure is shown in Figure 3. There are two parts in the control structure of the on line PID tuning method. The first part, which was described in the previous section, is the approximation of high order systems with FOPTD using fuzzy system, and the second part is the design of the PID controller. The parameters of the PID controller can be obtained from the corresponding parameters of the estimated FOPTD by fuzzy system. We have used the Ziegler-Nichols ultimate cycle tuning method (17) to compute the parameters of the PID controller:

$$K_p = 0.6\ K_u \qquad T_I = 0.5\ T_u, \qquad T_d = 0.125 T_u \tag{17}$$

Here, K_p, T_i, T_d, K_u and T_u are the proportional gain, integral time constant, derivative time constant, the ultimate gain and the ultimate period respectively. The ultimate gain and the ultimate period are calculated from the FOPDT model of the high order plant (Rad, 1997) It should be emphasized that other control algorithms could also be used. The PID controller is implemented in the following form:

$$\begin{aligned}&u(t) = K_p[e_t(t) + \frac{1}{T_i s}\int e_t(t)dt - T_d \frac{dy_f}{dt} \\ &e_t(t) = r(t) - y(t), \qquad Y_f(s) = \frac{Y(s)}{1 + 0.1 T_d s}\end{aligned} \tag{18}$$

where $u(t)$,$y(t)$,$r(t)$,$y(t)$and $y_f(t)$ are the controller output, process output, set-point and filtered derivative, respectively. The implementation of the adaptive PID is as follows:

1. *Approximate the first-order with time delay model (FOPTD) parameters by fuzzy system.*
2. *Determine the ultimate gain (K_u) and ultimate period (T_u) by the FOPTD model.*
3. *Find the PID controller parameters K_p,T_i and T_d from equation (11) and calculate u(t.)*
4. *Find the FOPTD model output $y_m(t)$ from the FOPTD Model Generator.*
5. *Calculate error between the (FOPTD) model output and the process output.*
6. *Update $c_i(t)$ by using equation(13-15) (Gradient descent algorithm).*
7. *Update the error between the set-point and the process output. Go to step (1)*

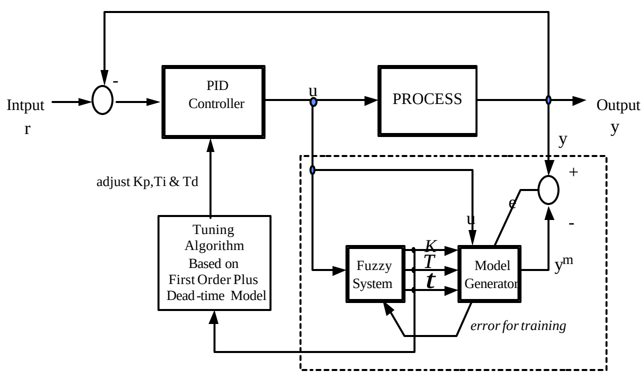

Fig. 3 On-line PID tuning using fuzzy system

4. SIMULATION RESULTS

To show the adaptive behaviour of the algorithm, let us consider three processes as:

Process I $\frac{Y(s)}{U(s)} = \frac{1.5e^{-2.5s}}{(s+1)^2}$

Process II $\frac{Y(s)}{U(s)} = \frac{1-1.4s}{(s+1)^3}$

Process III $\frac{Y(s)}{U(s)} = \frac{1.5s^{-3.0s}}{(s+1)^4}$

The first process is a second order with time delay system, the second process is a non-minimum phase system and the third process is a fourth order time delay system. First, adaptive control of Process I was simulated for t =120s after which the system was changed from Process I to Process II. For t =320s the system was switched from process II to process III. Furthermore, it should be noted that the gain in systems 1,2 and 3,2 are different (1.5 and 1.0). It is known that some adaptive controllers cannot cope with change in steady-state gain of the controlled system. However, as it is seen in Figure 5, the proposed method can successfully track the system change. Figure 4 shows the overall performance of the proposed algorithm. In the simulation, the set-point was selected to be a square wave with amplitude 0.6 and a period of 80s. A Gaussian noise with mean zero and variance of 0.001 was injected at the output of the system. We employed a fourth order Runge Kutta numerical integration algorithm for all time responses and the integration interval was selected to be 0.1s. The fuzzy system also used the same time interval for updating its parameters. The simulation proceeded as follows: the PID controller was initialized with $K_p = 1$, $T_i = 1000$, $T_d = 0.0$. The consequent values of fuzzy system were initialised with c^1=1.0, c^2=1.8, c^3=3.5. The learning rates were chosen as η_1=0.25, η_2=0.8 and η_3=0.8 respectively. Figure 4 shows the overall performance of the three controlled systems. In this figure, the set point and the output, the controller signal and the estimated parameters of gain, apparent time delay and the dominant time constant are shown in top, middle and bottom curves respectively. In all these system changes, the fuzzy system converged and the estimated parameters of the FOPDT also converged to their steady state values. The proposed method is shown to provide stable and robust control under various conditions. Tables 1, 2 and 3 show the parameters of FOPTD model approximated by several other methods such as Smith's (Smith, 1967), minimized error (Sundersan, 1978), and the corresponding ultimate gain and the ultimate period for processes I, II and III respectively. It should be noted that the parameters from all other methods except the proposed one were obtained off-line, from open loop excitation with unit step and were noise free. Furthermore, the values quoted for the proposed algorithm is based on the last measurement before each system change and not the average value.

Table 1
Process I FOPTD Model Parameters

	K	T	t	K_u	T_u
Smith Method[11]	1.5	1.65	3.00	1.06	8.38
Minimized-error[12]	1.5	1.46	3.11	0.98	8.44
Proposed method	1.5	1.33	3.19	0.936	8.48
Process I	-	-	-	1.036	8.438

Table 2
Process II FOPTD Model Parameters

	K	T	t	K_u	T_u
Smith Method[11]	1.0	1.89	2.43	1.93	7.22
Minimized-error[12]	1.0	1.67	2.55	1.74	7.35
Proposed method	1.0	1.39	2.45	1.64	6.91
Process II	-	-	-	1.54	6.83

Table 3
Process III FOPTD Model Parameters

	K	T	t	K_u	T_u
Smith Method[11]	1.5	2.49	4.86	1.03	13.39
Minimized-error[12]	1.5	2.057	5.1	0.923	13.49
Proposed method	1.5	2.66	4.75	1.07	13.33
Process III	-	-	-	0.987	13.48

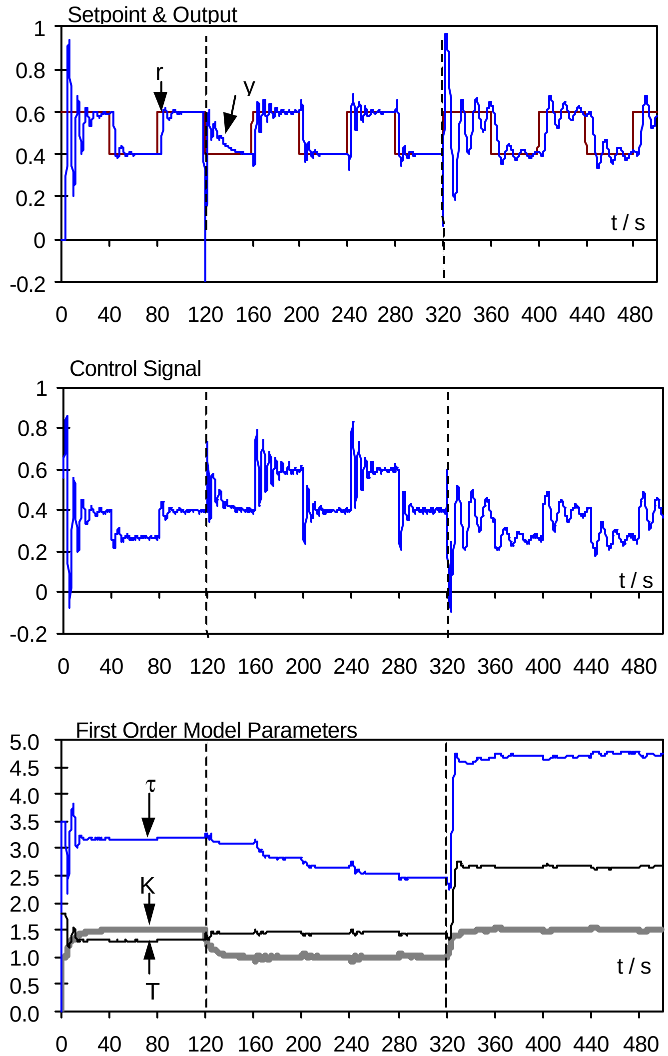

Fig 4. Simulation results of APID

5. CONCLUSIONS

In this paper, a new on-line FOPTD modelling method is proposed which is designed using fuzzy system theory. The proposed method is different from other fuzzy identification methods since it is integrated with a model generator to determine the parameters of FOPTD. The outputs of the fuzzy system are the three parameters of the FOPTD model. Combining with a PID controller, an on-line adaptive control using fuzzy system is designed and tested. The simplicity of the scheme for model-based control provides a new approach for implementing fuzzy applications for a variety of industrial control problems. Results presented clearly demonstrate the adaptive property of the proposed method.

6. ACKNOWLEDGEMENTS

The authors gratefully acknowledge the financial support of the Hong Kong Polytechnic University

REFERENCES

Ashwworth, M.J. (1982) Feedback design of systems with significant uncertainty. Lechworth, U.K. Research Studies Press.

Bhal, N. and Mecavoy, T.J. (1990) “Use of neural nets for dynamics modeling and control of chemical process systems.” Comput. Chem. Eng. 14: 573-583.

Babuska, R. and Vebruggen, H.B. (1996). “An overview of fuzzy modelling for control.” Control Eng. Practice. 4: 1593-1606.

Chen, C.L.; Chen P.C. and Chen, C.K. (1998) “Analysis and design of fuzzy control system.” Fuzzy Sets Systs. 57:125-140.

Czogala, E. and Pedrycz, W. (1981) “On identification in fuzzy systems and its applications in control problems.” Fuzzy Sets and Systems, 6: 73-83.

Gawthrop, P.J.; Nihtila, M.T. and Rad, A.B. (1989) “Recursive parameter estimation of continuous-time systems with unknown time delay.” Control Theory and Advanced Technology (C-TAT). 5(3): 227-248.

Lu, Y.Z.; He, M. and Xu, C.W. (1997) “Fuzzy modeling and expert optimisation control for industrial process.” IEEE Trans. Control Systems Technol. 5:2-12.

Narendra, K.S. and Parthasarathy, K. (1990) “Identificaion and control of dynamical systems using neural networks.” IEEE Trans. On Neural Networks, 1: 4-27.

Rad, A.B., Lo, W.L., and Tsang, K.M. (1997) “Self-tuning PID controller using Netwon Raphson search method” IEEE Trans. On Industrial Electronics. 44 (5): pp. 717-725.

Smith, J.A.; Lopez, A.M.; and Murrill, P.W. (1967) “A comparison of controller tuning rules.” Control Engineering.14, 72.

Smith, C. L. and Corripio, A.B. (1997) Principles and Practice of Automatic Process Control. John Wiley and Sons, Inc., Second Edition.

Sugeno, M. and Kang, G.T. (1986) “Structure identification of fuzzy model.” Fuzzy sets syst. 28:329-346.

Sundaresan, K.R. and Krishnaswamy, P.R. (1978) ”Estimation of Time Delay Time Constant Parameters in Time, Frequency, and Laplace Domains.” The Canadian Journal of Chemical Engineering, 56, April: 257-262.

Takagi, T. and M. Sugeno. 1985. “Fuzzy identification of systems and its applications to modeling and control.” IEEE trans. Syst. Man. Cybern. 15:116-132.

Tanaka, K. and Sugeno, M. (1992) “Stability analysis and design of fuzzy control systems.” Fuzzy Sets Systs. 45:135-156.

Wang, H.O.; Tanaka, K. and Griffin, M. (1996) “An approach to fuzzy control of nonlinear systems: stability and design issues.” IEEE Trans. Fuzzy Syst. 4(1):14-23.

Wang, L.X. (1997) A course in fuzzy systems and control. Englewood Cliffs, NJ :Prectice Hall.

Yu, C.C. (1999). Autotuning of PID Controllers. Springer, London.

Zadeh, L.A. (1965) "Fuzzy sets and systems" in: Proc. Sysmp. Systems Theory, Polytech. Inst. Brooklyn. 29-27.

Ziegler,J.G. and Nichols, N. B. (1942) Optimum Settings for Automatic controllers. Transactions on ASME, Nov: 759-768.

Appendix A1

$$\dot{\boldsymbol{X}}_m(t) = \boldsymbol{A}\boldsymbol{X}_m(t) + \boldsymbol{B}u(t-\boldsymbol{q}),$$

$$y_m(t) = \boldsymbol{C}\boldsymbol{X}_m(t) + \boldsymbol{D}u(t-\boldsymbol{q})$$

$$\boldsymbol{X}_m(t) = \boldsymbol{\ddot{O}}(t-t_0)\boldsymbol{X}_m(t_0) + \int_{t_0}^{t} \boldsymbol{\ddot{O}}(t-\boldsymbol{l})\boldsymbol{B}u(\boldsymbol{l}-\boldsymbol{q})d\boldsymbol{l}, \qquad \text{(A1)}$$

$$\boldsymbol{\ddot{O}}(t) = e^{\boldsymbol{A}t}$$

$$y_m(t) = \boldsymbol{C}\boldsymbol{\ddot{O}}(t-t_0)\boldsymbol{X}_m(t_0) + \boldsymbol{C}\int_{t_0}^{t} \boldsymbol{\ddot{O}}(t-\boldsymbol{l})\boldsymbol{B}u(\boldsymbol{l}-\boldsymbol{q})d\boldsymbol{l} + \boldsymbol{D}u(t-\boldsymbol{q})$$

Assuming that all initial states are zeros and $\boldsymbol{D=0}$, the output equation becomes:

$$y_m(t) = \int_{t_0}^{t} h(t-\boldsymbol{l},\boldsymbol{p})u(\boldsymbol{l}-\boldsymbol{q})d\boldsymbol{l}$$

$$e^{-sq}G_m(s) = e^{-sq}L[h(t,\boldsymbol{p})] = e^{-sq}\boldsymbol{C}(s\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{B} \qquad \text{(A2)}$$

$$= e^{-sq}\frac{b_0 s^n + b_1 s^{n-1} + \ldots + b_m}{s^n + a_1 s^{n-1} + \ldots + 1} = e^{-sq}\frac{B_m(s)}{A_m(s)}$$

where $\boldsymbol{q}$ is the model time delay and $h(t, \boldsymbol{p}) = L^{-1}[G_m(s)]$ is the impulse function of $G_m(s)$. The vector is defined as $\boldsymbol{p} = [a_1\ a_2 \ldots a_n\ b_0\ b_1 \ldots b_m]$. The partial derivatives of the model output with respect to the time delay and the model parameters are as follows.

$$\frac{\partial y_m}{\partial \boldsymbol{q}} = \frac{\partial}{\partial \boldsymbol{q}}\int_{t_0}^{t} h(t-\boldsymbol{l},\boldsymbol{p})u(\boldsymbol{l}-\boldsymbol{q})d\boldsymbol{l}$$

$$= \int_{t_0}^{t} \frac{\partial}{\partial \boldsymbol{q}}[h(t-\boldsymbol{l},\boldsymbol{p})u(\boldsymbol{l}-\boldsymbol{q})]d\boldsymbol{l} \qquad \text{(A3)}$$

$$= \int_{t_0}^{t} h(t-\boldsymbol{l},\boldsymbol{p})\frac{\partial u(\boldsymbol{l}-\boldsymbol{q})}{\partial \boldsymbol{q}}d\boldsymbol{l}$$

$$= \int_{t_0}^{t} [-h_m(t-\boldsymbol{l},\boldsymbol{p})\dot{u}(\boldsymbol{l}-\boldsymbol{q})]d\boldsymbol{l} = -L^{-1}[se^{-sq}G_m(s)U(s)]$$

$$\frac{\partial y_m}{\partial a_i} = \frac{\partial}{\partial a_i}\int_{t_0}^{t} h_m(t-l,\boldsymbol{p})u(l-q)dl$$

$$= \int_{t_0}^{t} \frac{\partial}{\partial a_i}[h_m(t-l,\boldsymbol{p})u(l-q)]dl \qquad \text{(A4)}$$

$$= \int_{t_0}^{t} \frac{\partial h(t-l,\boldsymbol{p})}{\partial a_i}u(l-q)dl$$

$$L[\frac{\partial h}{\partial a_i}] = \int_0^{\infty} \frac{\partial h}{\partial a_i}e^{-st}dt = \frac{\partial}{\partial a_i}\int_0^{\infty} h(t,\boldsymbol{p})e^{-st}dt$$

$$= \frac{\partial}{\partial a_i}L[h(t,\boldsymbol{p})] = \frac{\partial G_m(s)}{\partial a_i} \qquad \text{(A5)}$$

$$\Rightarrow \frac{\partial h}{\partial a_i} = L^{-1}[\frac{\partial G_m(s)}{\partial a_i}]$$

Similarly $\frac{\partial h}{\partial b_i} = L^{-1}[\frac{\partial G_m(s)}{\partial b_i}]$

$$\frac{\partial y_m}{\partial a_i} = \int_{t_0}^{t} \frac{\partial h(t-l,\boldsymbol{p})}{\partial a_i}u(l-q)dl$$

$$= L^{-1}[e^{-sq}\frac{\partial G_m(s)}{\partial a_i}U(s)] \qquad \text{(A6)}$$

$$= L^{-1}[-e^{-sq}\frac{s^{n-i}B_m(s)}{[A_m(s)]^2}U(s)]$$

$$\frac{\partial y_m}{\partial b_i} = \int_{t_0}^{t} \frac{\partial h(t-l,\boldsymbol{p})}{\partial b_i}u(l-q)dl$$

$$= L^{-1}[e^{-sq}\frac{\partial G_m(s)}{\partial b_i}U(s)] \qquad \text{(A7)}$$

$$= L^{-1}[e^{-sq}\frac{s^{m-i}}{A_m(s)}U(s)]$$

$$\frac{\partial y_m}{\partial \boldsymbol{q}} = L^{-1}[-se^{-sq}\frac{B_m(s)}{A_m(s)}U(s)] \qquad \text{(A8)}$$

For first order with time delay model $\frac{Ke^{-st}}{Ts+1}$

$$\frac{\partial y_m}{\partial K} = L^{-1}[\frac{e^{-st}}{Ts+1}U(s)] \qquad \text{(A9)}$$

$$\frac{\partial y_m}{\partial T} = L^{-1}[-\frac{Kse^{-st}}{(Ts+1)^2}U(s)] \qquad \text{(A10)}$$

$$\frac{\partial y_m}{\partial \boldsymbol{t}} = L^{-1}[-\frac{Kse^{-st}}{Ts+1}U(s)] \qquad \text{(A11)}$$

For second order with time delay model $\frac{Ke^{-st}}{a_0 s^2 + a_1 s + 1}$

$$\frac{\partial y_m}{\partial K} = L^{-1}[\frac{e^{-st}}{a_0 s^2 + a_1 s + 1}U(s)] \qquad \text{(A12)}$$

$$\frac{\partial y_m}{\partial a_0} = L^{-1}[-\frac{Ks^2 e^{-st}}{(a_0 s^2 + a_1 s + 1)^2}U(s)] \qquad \text{(A13)}$$

$$\frac{\partial y_m}{\partial a_1} = L^{-1}[-\frac{Kse^{-st}}{(a_0 s^2 + a_1 s + 1)^2}U(s)] \qquad \text{(A14)}$$

$$\frac{\partial y_m}{\partial \boldsymbol{t}} = L^{-1}[-\frac{Kse^{-st}}{a_0 s^2 + a_1 s + 1}U(s)] \qquad \text{(A15)}$$

The control signal $u(t)$ is filtered by the filter function in eq. (A9-A11) to find the partial derivatives of $y_m(t)$ with respect to various model parameters.