ME233 **ADAPTIVE CONTROL**

Parameter Identification and Adaptive Control

Introduction

We have studied the analysis and design of linear control systems assuming that we have a dynamic model of the controlled plant. While we have addressed the question of modeling uncertainties, we have assumed that the structure and (nominal) parameter values are known. In practice, however, this is not necessarily a good assumption. In the remaining part of ME 233, we will consider the analysis and design of control systems when the parameter values are either not known in advance or time varying. The basic assumption is that the structure of the model is known.

When the parameters are not known, they must be identified by off-line or on-line computation. "Off-line" means that identification is performed as an independent activity. "On-line" means that identification and control are progressing simultaneously. Results of off-line identification can be utilized in the off-line design of the controller. In some cases, off-line parameter identification and off-line controller design is either impractical or impossible: e.g. cases where the plant parameters vary during operation. In such cases, the controller must be adaptive in real time: i.e. the controller must be adaptive.

As an introductory example, we consider a thermal mixing process sketched below. In this system, the three way valve adjusts the mixing ratio of hot water and cold water, and the valve placed after the mixing point adjust s the total flow rate. We assume that the flow control loop has been provided to maintain the flow rate at a desired level. One goal is to control the temperature of water in the tank by adjusting the mixing ratio.

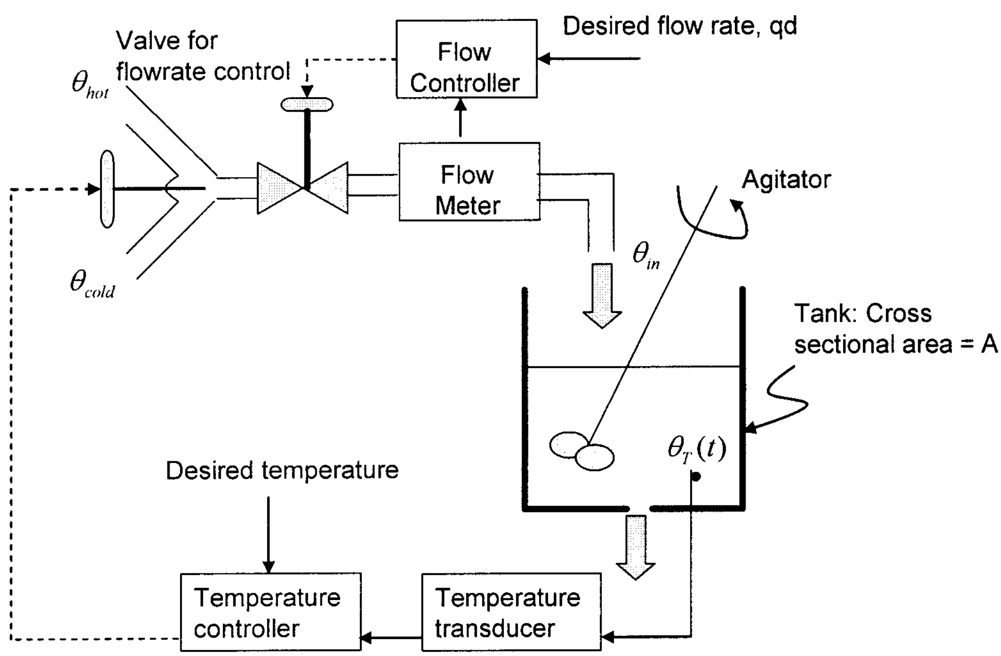

Fig. PIAC-1 Mixing Tank System

The differential equation model for the tank is

$$d(V(t)\,\theta_T(t))/dt = q_{in}(t)\,\theta_{in}(t) - q_{out}(t)\,\theta_T(t) \qquad \text{(PIAC-1)}$$
$$dV(t)/dt = q_{in}(t) - q_{out}(t) \qquad \text{(PIAC-2)}$$

where V is the volume of water in the tank, θ_T and θ_{in} are the temperature of water in the tank and that of the incoming fluid, respectively, and q_{in} and q_{out} are the flow rates. The flow rate of the outgoing fluid from the tank depends on the water level in the tank and valve characteristics: i.e.

$$q_{out}(t) = f_{valve}(V/A) \qquad \text{(PIAC-3)}$$

where A is the cross sectional area of the tank. It is assumed that q_{in} is maintained close to the desired flow rate, q_d, by a fast responding flow control loop. The temperature of the incoming fluid is related to the controlling input, i.e. the position of the three way valve, by

$$\theta_{in}(\mathrm{t}) = k\, u(t-t_d) \qquad \text{(PIAC-4)}$$

where u is the controlling input and the delay time t_d depends on the desired flow rate and the pipe length. From Eqs. (PIAC-1) and (PIAC-4), it can be seen that the dynamics between u(t) and the tank water temperature highly depend on the desired flow rate. For a simplified case where the flow rate is fixed, the dynamic equation is

$$d\theta_T(t)/dt = -\,(q_d/V_s)\,\theta_T(t) + (k\, q_d/V_s)\, u(t-t_d) \qquad \text{(PIAC-5)}$$

where $V_s = A\, f_{valve}^{-1}(q_d)$. The dependence of the time constant on q_d is clear in this expression. It is obvious that a conventional fixed gain feedback controller for the regulation of θ_T makes the closed loop performance dependent on the desired flow rate. Performance can be uniform if the controller can adjust itself to varying plant dynamics: i.e. the controller is adaptive.

Adaptation capability is provided by the adaptation mechanism in the adaptive controller. The adaptation mechanism contains the parameter adaptation algorithm (PAA). The most popular PAA is the least squares. The goal or role of the adaptation mechanism is to maintain the performance of the control system at an acceptable (desired) level under variation of the plant parameters. At this stage, we introduce several concepts related to adaptive control.

Fig. PIAC-2 shows the basic elements and their interactions in the adaptive system. Notice that the adaptive system in the figure has a closed loop in terms of a performance measure. This is a very unique aspect of **adaptive** systems. The adjustable system includes parameters that the adaptation mechanism can adjust; this implies that the dynamics of the adjustable system is controlled by the adaptation mechanism. In our study, we are primarily interested in a special kind of adaptive systems called the Model Reference Adaptive Systems (MRAS). MRAS is illustrated in Fig. PIAC-3. In MRAS, the desired performance is specified in terms of the input-output behavior of the reference model in the figure. The reference model and adjustable system are excited by the same input. The output error between reference model and adjustable system is a good measure for the difference between the desired and actual performances.

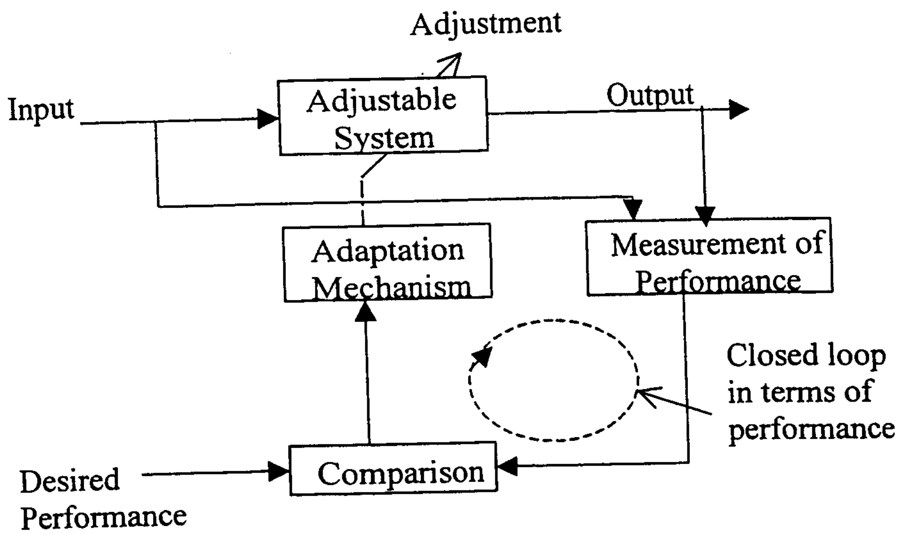

Fig. PIAC-2 Basic Elements and their Interactions in Adaptive Systems

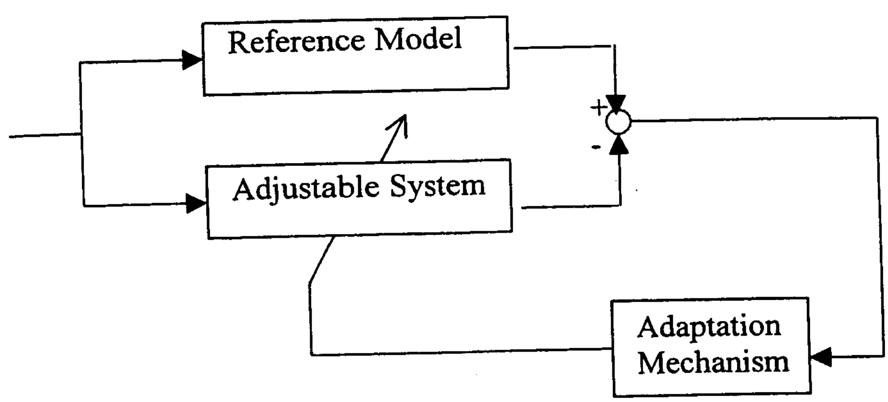

Fig. PIAC-3 Model Reference Adaptive System (MRAS)

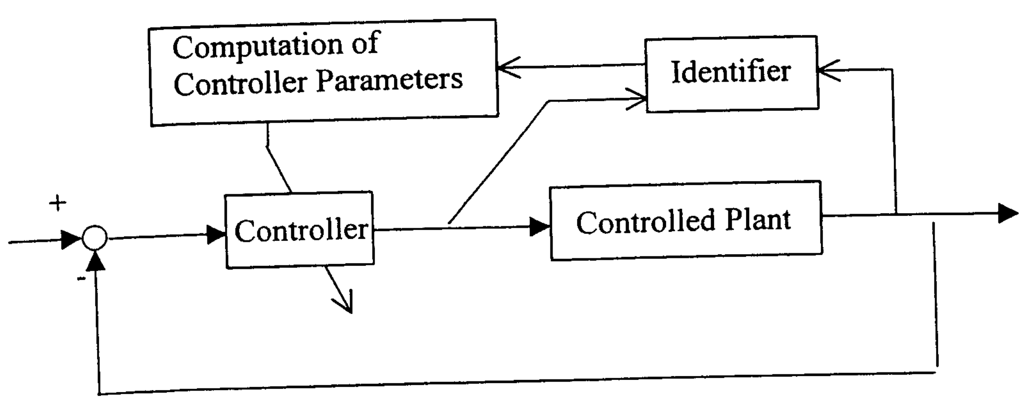

Fig. PIAC-4 Self Tuning Control

The MRAS in Fig. PIAC-3 can represent either parameter identification or adaptive (model following) control. The parameter identification problem refers to a problem of identifying or estimating unknown parameters of a system, the structure of which is assumed to be known. Since the structure is known, we may use a mathematical model, which has the identical structure as the plant, and adjust model parameters so that the input-output behavior of the model and that of the plant become close to each other. Note that the plant acts as a reference model in identification.

In the case of adaptive control, the adjustable system represents the controlled plant and the adjustable controller.

Controller parameters, such as PID control gains, are adjustable in adaptive control. When the controller parameters are adjusted by PAA in a direct manner, the adaptive controller is called direct. Instead of adjusting the controller parameters by PAA directly, they can be calculated based on the estimates of plant parameters. The adaptive controller of this kind is called indirect. The self tuning control system illustrated in Fig. PIAC-4 represents a popular indirect adaptive control approach.

Parameter Adaptation Algorithm and Identification

Consider a single-input, single-output (SISO) system described by

$$y(k+1) = -\sum_{i=1}^{n} a_i y(k+1-i) + \sum_{j=0}^{m} b_j u(k-j) = \theta^T \phi(k) \qquad \text{(PIAC-6)}$$

where

$$\theta^T = [a_1, \ \ldots, \ a_n, \ b_0, \ \ldots, \ b_m] \qquad \text{(PIAC-7)}$$

$$\phi^T(k) = [-y(k), \ \ldots, \ -y(k+1-n), \ u(k), \ \ldots, \ u(k-m)] \qquad \text{(PIAC-8)}$$

Notice that the system transfer function can be written as

$$G_p(z^{-1}) = \frac{z^{-1}B(z^{-1})}{A(z^{-1})} \qquad \text{(PIAC-9)}$$

where

$$B(z^{-1}) = b_0 + b_1 z^{-1} + \cdots + b_m z^{-m} \qquad \text{(PIAC-10)}$$

and

$$A(z^{-1}) = 1 + z^{-1}A^*(z^{-1}) = 1 + a_1 z^{-1} + \cdots + a_n z^{-n}$$
$$A^*(z^{-1}) = a_1 + a_2 z^{-1} + \cdots + a_n z^{-n+1} \qquad \text{(PIAC-11)}$$

Let us suppose that we have a data set $\{u(l), y(l) \,|\, -(n-1) \leq l \leq k\}$ avaialble but that we do not know the values of a_i's and b_j's. If we have an estimate of the parameter vector, θ, we may use the available data set and the estimate to predict $y(i)$, $i \leq k$, to be

$$\hat{y}(i) = \hat{\theta}^T\phi(i-1) \qquad \text{(PIAC-12)}$$

and judge the accuracy of the estimate from the prediction error

$$\varepsilon(i) = y(i) - \hat{y}(i) \qquad \text{(PIAC-13)}$$

The least squares estimate minimizes the summation of the squared prediction error,

$$J = \sum_{i=1}^{k} [y(i) - \hat{\theta}^T(k)\phi(i-1)]^2 \qquad \text{(PIAC-14)}$$

The least squares estimate can be found by setting the partial derivative of J with respect to θ to zero.

Noting that

$$\frac{\partial J}{\partial \hat{\theta}} = -2\sum_{i=1}^{k} [y(i) - \hat{\theta}^T(k)\phi(i-1)]\phi(i-1) = -2\sum_{i=1}^{k} [\phi(i-1)y(i) - \phi(i-1)\phi^T(i-1)\hat{\theta}(k)] \qquad \text{(PIAC-15)}$$

we find

$$\hat{\theta}(k) = F(k)\sum_{i=1}^{k} \phi(i-1)y(i) \qquad \text{(PIAC-16)}$$

where

$$F(k) = \left[\sum_{i=1}^{k} \phi(i-1)\phi^T(i-1)\right]^{-1} \qquad \text{(PIAC-17)*}$$

The least squares formula (PIAC-16) and (PIAC-17) is called the batch formula since it processes the available data set all at once. Let us rewrite the least squares formula into a recursive form. When we move from the time instance k to the next time instance k+1, $\theta(k+1)$ in the recursive formula is obtained from $\theta(k)$ and a correction term, that is

$$\hat{\theta}(k+1) = \hat{\theta}(k) + [correction\_term] \qquad \text{(PIAC-18)}$$

To find the recursive formula, we first note

$$F^{-1}(k+1) = F^{-1}(k) + \phi(k)\phi^T(k) \qquad \text{(PIAC-19)}$$

and

$$\sum_{i=1}^{k} \phi(i-1)y(i) = F^{-1}(k)\hat{\theta}(k) \qquad \text{(PIAC-20)}$$

*If the inverse matrix in (PIAC-17) does not exist, use the pseudoinverse, $F(k) = \left[\sum_{i=1}^{k}\phi(i-1)\phi^T(i-1)\right]^{\#}$. $X = A^{\#}$ denotes the pseudoinverse of A and satisfies the following properties: 1) X has the same dimension as A^T, 2) $AXA = A$ and $XAX = A$, and 3) $AX = X^TA^T$.

From Eqs. (PIAC-19) and (PIAC-20),

$$\begin{aligned}\hat{\theta}(k+1) &= F(k+1)\left[\sum_{i=1}^{k}\phi(i-1)y(i)+\phi(k)y(k+1)\right] \\ &= F(k+1)[F^{-1}(k)\hat{\theta}(k)+\phi(k)y(k+1)] \\ &= F(k+1)[\{F^{-1}(k+1)-\phi(k)\phi^T(k)\}\hat{\theta}(k)+\phi(k)y(k+1)] \\ &= \hat{\theta}(k)+F(k+1)\phi(k)[y(k+1)-\hat{\theta}^T(k)\phi(k)]\end{aligned} \tag{PIAC-21}$$

In Eq. (PIAC-21), $\theta^T(k)\phi(k)$ is the *a priori* predicted output based on the parameter estimate vector at time k. Writing the a priori and a posteriori predicted output as

$$\textit{a priori}: \quad \hat{y}^o(k+1) = \hat{\theta}^T(k)\phi(k) \tag{PIAC-22a}$$

$$\textit{a posteriori}: \quad \hat{y}(k+1) = \hat{\theta}^T(k+1)\phi(k) \tag{PIAC-22b}$$

respectively, and the *a priori* and *a posteriori* prediction error as

$$\textit{a priori}: \quad \varepsilon^o(k+1) = y(k+1)-\hat{y}^o(k+1) \tag{PIAC-23a}$$

$$\textit{a posteriori}: \quad \varepsilon(k+1) = y(k+1)-\hat{y}(k+1) \tag{PIAC-23b}$$

respectively, we can express Eq. (PIAC-21) as

$$\hat{\theta}(k+1) = \hat{\theta}(k)+F(k+1)\phi(k)\varepsilon^o(k+1) \tag{PIAC-24}$$

Eq. (PIAC-24) is the parameter adaptation algorithm in the recursive form.

Eq. (PIAC-19) updates $\mathbf{F}^{-1}(k)$ in the recursive form. However, it is not convenient to update $\mathbf{F}^{-1}(k)$ since Eq. (PIAC-24) requires $\mathbf{F}(k+1)$. The recursive formula for $\mathbf{F}(k)$ can be obtained by applying the matrix inversion lemma[1] to Eq. (PIAC-19), and it is

$$F(k+1) = F(k)-\frac{F(k)\phi(k)\phi^T(k)F(k)}{1+\phi^T(k)F(k)\phi(k)} \tag{PIAC-25}$$

Notice that the recursive formula (PIAC-25) does not involve any matrix inversion, which is advantageous in real time applications of the least squares method.

Multiplication of $\phi(k)$ to Eq. (PIAC-25) from right yields

$$F(k+1)\phi(k) = \frac{F(k)\phi(k)}{1+\phi^T(k)F(k)\phi(k)} \tag{PIAC-26}$$

By using Eq. (PIAC-26), Eq. (PIAC-24) can be also expressed as

---

[1]For an $n\times n$ nonsingular matrix P, $m\times m$ nonsingular matrix R and $n\times m$ rectangular matrix H, $(P^{-1}+H^TR^{-1}H)^{-1}= P-PH^T(R+HPH^T)^{-1}HP$.

(PIAC-25) can be obtained from (PIAC-19) as follows. Multiply (PIAC-19) $F(k+1)$ from left and $F(k)$ from right to obtain $F(k) = F(k+1)+F(k+1)\phi(k)\phi^T(k)F(k)$ (*). Multiply (*) $\phi(k)$ from right and do a little algebra to obtain $F(k+1)\phi(k) = F(k)\phi(k)/([1+\phi^T(k)F(k)\phi(k)]$. Substitute this expression to (*) to obtain (PIAC-25).

$$\hat{\theta}(k+1) = \hat{\theta}(k) + \frac{F(k)\phi(k)}{1+\phi^T(k)F(k)\phi(k)}\varepsilon^o(k+1) \qquad \text{(PIAC-27)}$$

From Eqs. (PIAC-23a), (PIAC-23b) and (PIAC-27), we obtain the relation between the *a priori* error and the *a posteriori* error,

$$\varepsilon(k+1) = \frac{\varepsilon^o(k+1)}{1+\phi^T(k)F(k)\phi(k)} \qquad \text{(PIAC-28)}$$

Notice that Eq. (PIAC-28) implies $|\varepsilon(k+1)| \leq |\varepsilon^o(k+1)|$. Eqs. (PIAC-27) and (PIAC-28) mean that the parameter adaptation algorithm (PAA) may be expressed as

$$\hat{\theta}(k+1) = \hat{\theta}(k) + F(k)\phi(k)\varepsilon(k+1) \qquad \text{(PIAC-29)}$$

This form of PAA is not implementable but is suited for stability analysis.

The recursive least square algorithm can be initialized either by applying the batch formula once after collecting enough input and output data or by setting $\theta(0) = 0$ and $F(0) = \sigma I$ where σ is a large number. The latter approach can be justified by noting that for large σ's the infuence of $F^{-1}(0)$ is small in Eq. (PIAC-19).

As can be seen from Eq. (PIAC-19), the inverse of the adapta tion gain F(k) is normally increasing. This implies that the adaptation gain F(k) is decreasing, which is a consequence of the performance index (PIAC-14). In deriving the least square formula, the parameters a_i's and b_j's were assumed to be time invariant. However, the reason for real time identification is often the time varying nature of those parameters. In such applications, the time decreasing adaptation gain is not suited since it means that adaptation becomes weaker and weaker. One way to fix this problem is to let the least square identifier forget the past by introducing a <u>forgetting factor</u>, λ, in the performance index: i.e. we use instead of (PIAC-15)

$$J = \sum_{i=1}^{k} \lambda^{k-i}[y(i) - \hat{\theta}^T(k)\phi(i-1)]^2, \qquad 0 < \lambda \leq 1 \qquad \text{(PIAC-30)}$$

The recursive least square estimate with the forgetting factor is still given by Eq. (PIAC-24). However, the adaptation gain is given by

$$F^{-1}(k+1) = \lambda F^{-1}(k) + \phi(k)\phi^T(k) \qquad \text{(PIAC-31)}$$

or

$$F(k+1) = \frac{1}{\lambda}\left[F(k) - \frac{F(k)\phi(k)\phi^T(k)F(k)}{\lambda + \phi^T(k)F(k)\phi(k)}\right] \qquad \text{(PIAC-32)}$$

A more general formula[2] for F(k) is

---

[2]I. D. Landau, System Identification and Control Design, Prentice Hall, 1990.

$$F^{-1}(k+1) = \lambda_1(k)F^{-1}(k)+\lambda_2(k)\phi(k)\phi^T(k) \tag{PIAC-33}$$

where

$$F(0) > 0, \quad 0 < \lambda_1(k) \leq 1 \quad \textit{and} \quad 0 \leq \lambda_2(k) < 2 \tag{PIAC-34}$$

In actual implementation, F(k) is updated by the following formula which is obtained by applying the matrix inversion lemma to (PIAC-33).

$$F(k+1) = \frac{1}{\lambda_1(k)}\left[F(k)-\frac{F(k)\phi(k)\phi^T(k)F(k)}{\lambda_1(k)/\lambda_2(k) + \phi^T(k)F(k)\phi(k)}\right] \tag{PIAC-35}$$

This adaptation gain is used with the parameter updating law (PIAC-27) to assure the stability. However, it should be noted that (PIAC-27) is equivalent to (PIAC-24) only for $\lambda_1(k) = \lambda_2(k) = 1$. By selecting $\lambda_1(k)$ and $\lambda_2(k)$ properly, various kinds of adaptation gains can be obtained:

i. $\lambda_1(k) = 1$ and $\lambda_2(k) = 0$ for the canstant adaptation gain

ii. $\lambda_1(k) = \lambda_2(k) = 1$ for the least square gain

iii. $\lambda_1(k) = \ < 1$ and $\lambda_2(k) = 1$ for the least square gain with forgetting factor.

Furthermore, by taking the advantage of $\lambda_1(k)$ and $\lambda_2(k)$ both being time varying, we can set $\lambda_1(k)/\lambda_2(k) = \alpha =$ const. (typical value of α is between 0.5 and 1) and adjusting $\lambda_1(k)$, it is possible to control the trace of F(k) to a desired value in updating F(k) by Eq. (PIAC-35).

Another simplified algorithm is the projection algorithm (see Goodwin and Sin, Astrom and Wittenmark):

$$\hat{\theta}(k+1) = \hat{\theta}(k)+\frac{\gamma\phi(k)}{\alpha+\phi^T(k)\phi(k)}[y(k+1)-\hat{\theta}^T(k)\phi(k)]$$

where $\alpha \geq 0$ and $0 < \gamma < 2$.

When we use recursive algorithms, we would like to be assured that algoirhtms are stable. In this case, stability implies that the adaptation error signal converges to zero.