I Parameter Adaptation Algorithm Review

0. Reference Model (Plant):

$$\frac{Y(z)}{U(z)} = \frac{z^{-1}B(z^{-1})}{A(z^{-1})} = \frac{z^{-1}(b_0 + b_1 z^{-1} + \cdots + b_m z^{-m})}{1 + a_1 z^{-1} + \cdots + a_n z^{-n}}, \tag{1}$$

Written as a difference equation:

$$\begin{aligned} y(k+1) &= -a_1 y(k) - \cdots - a_n y(k-n+1) + b_0 u(k) + \cdots + b_m u(k-m) \\ &= \theta^T \phi(k) \end{aligned} \tag{2}$$

where

$$\theta^T = \begin{bmatrix} a_1 & \cdots & a_n & b_0 & \cdots & b_m \end{bmatrix}, \tag{3}$$

$$\phi^T(k) = \begin{bmatrix} -y(k) & \cdots & -y(k-n+1) & u(k) & \cdots & u(k-m) \end{bmatrix}. \tag{4}$$

1. Formulation of the Identification Problem:

Given data set $\{u(l), y(l) \mid -(n-1) \le l \le k\}$, we want to find an estimate $\hat{\theta}(k)$ of the unknown parameter vector θ by minimizing the following sum of squares.

$$J = \sum_{i=1}^{k} \left[y(i) - \hat{\theta}^T(k)\phi(i-1) \right]^2 . \tag{5}$$

2. Batch Formula:

The solution is found by making the derivative of J equal to zero:

$$\hat{\theta}(k) = \left[\sum_{i=1}^{k} \phi(i-1)\phi^T(i-1) \right]^{-1} \sum_{i=1}^{k} \left[y(i)\phi(i-1) \right]. \tag{6}$$

3. Recursive Least Squares Algorithm:

The solution given by eq. (6) can be put into a recursive form

$$\hat{\theta}(k+1) = \hat{\theta}(k) + F(k+1)\phi(k)[y(k+1) - \hat{\theta}^T(k)\phi(k)], \tag{7}$$

where

$$F(k+1) = F(k) - \frac{F(k)\phi(k)\phi^T(k)F(k)}{1 + \phi^T(k)F(k)\phi(k)}. \tag{8}$$

$F(k)$ is called the adaptation gain. This adaptation algorithm is called *recursive least squares (RLS)*.

3.1 Definitions of Some Signals

a priori predicted output: $$\hat{y}^o(k+1) = \hat{\theta}^T(k)\phi(k) \tag{9}$$

a posteriori predicted output: $$\hat{y}(k+1) = \hat{\theta}^T(k+1)\phi(k) \tag{10}$$

a priori prediction error: $$\varepsilon^o(k+1) = y(k+1) - \hat{y}^o(k+1) \tag{11}$$

a posteriori prediction error: $$\varepsilon(k+1) = y(k+1) - \hat{y}(k+1) \tag{12}$$

3.2 Some Useful Relation Between Signals for RLS

$$F(k+1)^{-1} = F(k)^{-1} + \phi(k)\phi^T(k) \tag{13}$$

$$F(k+1)\phi(k) = \frac{F(k)\phi(k)}{1+\phi^T(k)F(k)\phi(k)} \tag{14}$$

$$\varepsilon(k+1) = \frac{\varepsilon^o(k+1)}{1+\phi^T(k)F(k)\phi(k)} \tag{15}$$

From eq. (15) and the definitions of prediction errors, the recursive formula (7) can also be written as

$$\hat{\theta}(k+1) = \hat{\theta}(k) + F(k+1)\phi(k)\varepsilon^o(k+1) \tag{16}$$

$$\hat{\theta}(k+1) = \hat{\theta}(k) + F(k)\phi(k)\varepsilon(k+1) \tag{17}$$

Plugging (14) into (16), we get

$$\hat{\theta}(k+1) = \hat{\theta}(k) + \frac{F(k)\phi(k)}{1+\phi^T(k)F(k)\phi(k)}\varepsilon^o(k+1)\,. \tag{18}$$

Equations (7), (16), and (18) can all be implemented to update the parameter estimate, while eq. (17) is useful for stability analysis.

3.3 Choice of the Adaptation Gain

The recursive formula for $F(k+1)^{-1}$ is generalized by introducing two weighting sequences $\lambda_1(k)$ and $\lambda_2(k)$:

$$F(k+1)^{-1} = \lambda_1(k)F(k)^{-1} + \lambda_2(k)\phi(k)\phi^T(k)\,, \tag{19}$$

which is implemented by the following recursive formula

$$F(k+1) = \frac{1}{\lambda_1(k)}\left[F(k) - \frac{F(k)\phi(k)\phi^T(k)F(k)}{\lambda_1(k)/\lambda_2(k) + \phi^T(k)F(k)\phi(k)}\right]. \tag{20}$$

Some popular adaptation gains:

1) $\lambda_1(k) = 1$ and $\lambda_2(k) = 1$ for the RLS algorithm.

2) $\lambda_1(k) = 1$ and $\lambda_2(k) = 0$ for the constant adaptation gain ($F(k) = F > 0$).

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

For the stability consideration, no matter how the adaptation gain is updated, the recursive formula for the parameter estimate is always chosen to be

$$\hat{\theta}(k+1) = \hat{\theta}(k) + F(k)\phi(k)\varepsilon(k+1)\,. \tag{21}$$

From this equation and the definitions given by (9)-(12), we can get the same relation (15) for any adaptation gain. Then the following formula for implementation is obtained by plugging (15) into (21).

$$\hat{\theta}(k+1)=\hat{\theta}(k)+\frac{F(k)\phi(k)}{1+\phi^T(k)F(k)\phi(k)}\varepsilon^o(k+1)\,. \tag{22}$$

Note that eq. (7), (8), (14) and (16) hold only for the RLS, while eq. (15), (17) and (18) hold for any choice of the adaptation gain.

4. Stability of Parameter Adaptation Algorithms

4.1 Definitions and Theorems

1) Popov inequality
2) Definition of hyperstability
3) Definition of asymptotic hyperstability
4) Theorem of hyperstability
5) Theorem of asymptotic hyperstability
6) Definition of positive real transfer function
7) Definition of strictly positive real (SPR) transfer function

4.2 Procedure of Stability Analysis by Hyperstability Theorem

Step 1: Represent the adaptation algorithm by a feedback loop as in Fig. H-1 (equivalent feedback system).

Step 2: Verify that the feedback block satisfies the Popov inequality.

Step 3: Check that the feedforward block is SPR.

Steps 1 to 3 constitute the sufficiency portion of the asymptotic hyperstability theorem.

Step 4: Show that the output of the feedback block is bounded. Then from the definition of asymptotic hyperstability, we conclude that $\lim_{k\to\infty} x(k)=0$.

4.3 Stability of the Constant Gain PAA

Step 1: Recursive formula for stability analysis:

$$\tilde{\theta}(k+1)=\tilde{\theta}(k)+F\phi(k)\varepsilon(k+1) \tag{23}$$

$$\varepsilon(k+1)=-\tilde{\theta}^T(k+1)\phi(k) \tag{24}$$

where

$$\tilde{\theta}(k):=\hat{\theta}(k)-\theta\,. \tag{25}$$

The equivalent feedback system for this PAA is shown in Fig. PIAC-7.

Step 2: $\sum_{k=0}^{k_1}\left[\varepsilon(k+1)\phi^T(k)\tilde{\theta}(k+1)\right]\geq-\frac{1}{2}\tilde{\theta}^T(0)F^{-1}\tilde{\theta}(0),\ \text{for all } k_1$.

Step 3: $G(z)=1$ is SPR.

Step 4:

$$\sum_{k=0}^{k_1}\left[\varepsilon(k+1)\phi^T(k)\tilde{\theta}(k+1)\right]=-\sum_{k=0}^{k_1}\left[\varepsilon(k+1)\right]^2\geq-\frac{1}{2}\tilde{\theta}^T(0)F^{-1}\tilde{\theta}(0),\ \text{for all } k$$

$$\Rightarrow \lim_{k_1 \to \infty} \sum_{k=0}^{k_1} [\varepsilon(k+1)]^2 \le \frac{1}{2} \tilde{\theta}^T(0) F^{-1} \tilde{\theta}(0)$$
$$\Rightarrow \varepsilon(k+1) \text{ is bounded and } \lim_{k \to \infty} \varepsilon(k+1) = 0$$

5. Series Parallel Predictor

The adaptation algorithms we learned so far are considered as series-parallel predictors. Notice that eq. (12) can be written as

$$\begin{aligned} \varepsilon(k+1) &= y(k+1) - \left\{ \sum_{j=0}^{m} \left[\hat{b}_j(k+1) u(k-j) \right] - \sum_{i=1}^{n} \left[\hat{a}_i(k+1) y(k+1-i) \right] \right\} \\ &= y(k+1) - \left\{ \hat{B}(q^{-1}, k+1) u(k) - \hat{A}^*(q^{-1}, k+1) y(k) \right\} \end{aligned}$$

where

$$\hat{B}(q^{-1}, k+1) = \sum_{j=0}^{m} \left[\hat{b}_j(k+1) q^{-j} \right] \quad \text{and} \quad \hat{A}^*(q^{-1}, k+1) = \sum_{i=1}^{n} \left[\hat{a}_i(k+1) q^{-i} \right].$$

(q^{-1} is the one-sample-delay operator.) Then we can represent the PAA by the block diagram shown in Fig. PIAC-5.

The series-parallel PAA is asymptotically hyperstable if $0 < \lambda_1(k) \le 1$ and $0 \le \lambda_2(k) < 2$.

II Different PAA Structures

Our PAA can be categorized by the structure of the predictor. We have seen four different structures of PAA: series-parallel type, parallel type, parallel predictor with a fixed compensator, and parallel predictor with an adjustable compensator.

1. Series-Parallel Predictor

(1) Definitions of signals

$$\theta^T = \begin{bmatrix} a_1 & \cdots & a_n & b_0 & \cdots & b_m \end{bmatrix}$$
$$\phi^T(k) = \begin{bmatrix} -y(k) & \cdots & -y(k-n+1) & u(k) & \cdots & u(k-m) \end{bmatrix}$$
$$v(k+1) = \varepsilon(k+1)$$

(2) Stability condition: $0 < \lambda_1(k) \le 1$ and $0 \le \lambda_2(k) < 2$.

(3) **Advantage**: Always stable for the commonly used $\lambda_1(k)$ and $\lambda_2(k)$.

Disadvantage: The noise signal $w(k+1)$ (eq. PIAC-58) under this structure is rarely white, which results in biased parameter estimate (theorem on page PIAC-19).

2. Parallel Predictor

(1) Definitions of signals

$$\theta^T = \begin{bmatrix} a_1 & \cdots & a_n & b_0 & \cdots & b_m \end{bmatrix}$$

$$\phi^T(k) = \left[-\hat{y}(k) \quad \cdots \quad -\hat{y}(k-n+1) \quad u(k) \quad \cdots \quad u(k-m)\right]$$
$$v(k+1) = \varepsilon(k+1)$$

(2) Stability condition: $0 < \lambda_1(k) \leq 1$, $0 \leq \lambda_2(k) < 2$ and $\frac{1}{A(z^{-1})} - \frac{\lambda}{2}$ is SPR, where $\lambda = \max_k \lambda_2(k)$.

(3) **Advantage**: When noise exists, the condition for unbiased parameter convergence is usually satisfied.

Disadvantage: The stability condition is usually violated.

3. Parallel Predictor with a Fixed Compensator

(1) Definitions of signals

$$\theta^T = \left[a_1 \quad \cdots \quad a_n \quad b_0 \quad \cdots \quad b_m\right]$$
$$\phi^T(k) = \left[-\hat{y}(k) \quad \cdots \quad -\hat{y}(k-n+1) \quad u(k) \quad \cdots \quad u(k-m)\right]$$
$$v(k+1) = C(z^{-1})\varepsilon(k+1),\ C(z^{-1}) = 1 + c_1 z^{-1} + \cdots + c_n z^{-n}$$

(2) Stability condition: $0 < \lambda_1(k) \leq 1$, $0 \leq \lambda_2(k) < 2$ and $\frac{C(z^{-1})}{A(z^{-1})} - \frac{\lambda}{2}$ is SPR, where $\lambda = \max_k \lambda_2(k)$.

(3) **Advantage**: When noise exists, the condition for unbiased parameter convergence is usually satisfied. The stability condition is satisfied, if $C(z^{-1})$ is close to $A(z^{-1})$.

Disadvantage: More computation. It is difficult to determine $C(z^{-1})$, since we usually don't know $A(z^{-1})$ that we are trying to identify.

4. Parallel Predictor with an Adjustable Compensator

(1) Definitions of signals

$$\theta_e^T = \left[a_1 \quad \cdots \quad a_n \quad b_0 \quad \cdots \quad b_m \quad a_1 \quad \cdots \quad a_n\right]$$
$$\hat{\theta}_e^T(k) = \left[\hat{a}_1(k) \quad \cdots \quad \hat{a}_n(k) \quad \hat{b}_0(k) \quad \cdots \quad \hat{b}_m(k) \quad \hat{c}_1(k) \quad \cdots \quad \hat{c}_n(k)\right]$$
$$\phi_e^T(k) = \left[-\hat{y}(k) \quad \cdots \quad -\hat{y}(k-n+1) \quad u(k) \quad \cdots \quad u(k-m) \quad -\varepsilon(k) \quad \cdots \quad -\varepsilon(k-n+1)\right]$$
$$v(k+1) = \varepsilon(k+1) + \hat{c}_1(k+1)\varepsilon(k) + \cdots + \hat{c}_n(k+1)\varepsilon(k+1-n)$$

Notice that in the PAA of this predictor, $\hat{\theta}(k)$ and $\phi(k)$ are replaced by $\hat{\theta}_e(k)$ and $\phi_e(k)$

(2) Stability condition: $0 < \lambda_1(k) \leq 1$ and $0 \leq \lambda_2(k) < 2$.

(3) **Advantage**: The stability condition is always satisfied.

Disadvantage: A lot more computation. When noise exists, the convergence of parameter estimate depends on $A(z^{-1})$ (theorem on page PIAC-23).

III Popov Inequality Lemma

Lemma: Suppose that the two systems, S1 and S2, both satisfy the Popov inequality. Then, the system resulting from the parallel connection (Fig. 1) or the negative feedback connection (Fig. 2) of S1 and S2 satisfies the Popov inequality.

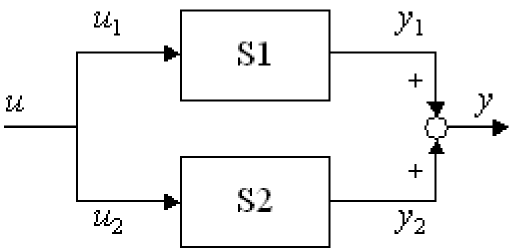

Fig. 1 Parallel connection of S1 and S2.

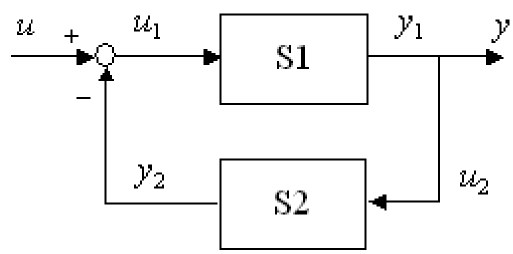

Fig. 2 Negative feedback connection of S1 and S2.

Proof: S1 and S2 satisfying the Popov inequality implies that for any $k_1 > 0$,

$$\sum_{k=0}^{k_1} u_1^T(k) y_1(k) \geq -\gamma_1^2$$
$$\sum_{k=0}^{k_1} u_2^T(k) y_2(k) \geq -\gamma_2^2$$

The proof for the lemma is based on the observation that in the parallel connection,

$$u = u_1 = u_2 \ ; \quad y = y_1 + y_2$$

and in the negative feedback connection,

$$u = u_1 + y_2 \ ; \quad y = y_1 = u_2 \ .$$

Then for the resulting system either in Fig. 1 or in Fig. 2,

$$\sum_{k=0}^{k_1} u^T(k) y(k) = \sum_{k=0}^{k_1} u_1^T(k) y_1(k) + \sum_{k=0}^{k_1} u_2^T(k) y_2(k) \geq -\gamma_1^2 - \gamma_2^2, \text{ for any } k_1 > 0 \, .$$