Adaptive Control

Chapter 3:
Parameter adaptation algorithms – deterministic environment

Chapter 3:
Parameter adaptation algorithms – deterministic environment

Abstract Parameter adaption algorithms are the key step for building an adaptive control system. An extensive coverage of the subject is provided in this chapter. Both synthesis and analysis of the parameter adaptation algorithms in a deterministic environment will be considered. Stability and convergence issues will be emphasized.

Parametric adaptation algorithms (PAA)

- Indirect adaptive control uses PAA in the plant model estimator
- Direct adaptive control uses PAA for controllers' parameter estimation
- Adaptive control with multiple models needs also PAA
- Self-tuning control using identification in closed loop needs also PAA

Parametric adaptation algorithm (PAA)

Parameter vector = contains all the parameters of the model

$$\begin{bmatrix} \text{New parameters} \\ \text{estimation} \\ \text{(vector)} \end{bmatrix} = \begin{bmatrix} \text{Old parameters} \\ \text{estimation} \\ \text{(vector)} \end{bmatrix} +$$

$$\begin{bmatrix} \text{Adaptation} \\ \text{Gain} \\ \text{(matrix)} \end{bmatrix} \times \begin{bmatrix} \text{Measurement} \\ \text{function} \\ \text{(vector)} \end{bmatrix} \times \begin{bmatrix} \text{Error prediction} \\ \text{function} \\ \text{(scalar)} \end{bmatrix}$$

↑ Regressor vector

We will develop the PAA in the context of plant model estimation

Plant Model

$$G(q^{-1}) = \frac{q^{-d}B(q^{-1})}{A(q^{-1})} = \frac{q^{-d-1}B^*(q^{-1})}{A(q^{-1})}$$

$$A(q^{-1}) = 1 + a_1 q^{-1} + \ldots + a_{n_A} q^{-n_A} = 1 + q^{-1} A^*(q^{-1})$$

$$B(q^{-1}) = b_1 q^{-1} + \ldots + b_{n_B} q^{-n_B} = q^{-1} B^*(q^{-1})$$

$$y(t+1) = -A^*(q^{-1})y(t) + B^*(q^{-1})u(t-d) = \theta^T \phi(t)$$

$$\theta^T = \left[a_1, \ldots a_{n_A}, b_1, \ldots, b_{n_B}\right]$$ ← Parameter vector

$$\phi(t)^T = \left[-y(t) \ldots - y(t - n_A + 1), u(t-d) \ldots u(t - d - n_B + 1)\right]$$

↖ Measurement vector

Algorithms for parameter estimation

Discrete time plant model (unknown parameters))

$$y(t+1) = -a_1 y(t) + b_1 u(t) = \theta^T \phi(t)$$

Measurement vector

$\theta^T = [a_1, b_1]$ ← Parameter vector ; $\phi(t)^T = [-y(t), u(t)]$

Adjustable prediction model (à priori)

$$\hat{y}^o(t+1) = \hat{y}(t+1 | \hat{\theta}(t)) = -\hat{a}_1(t) y(t) + \hat{b}_1(t) u(t) = \hat{\theta}(t)^T \phi(t)$$

$\theta(t)^T = [\hat{a}_1(t), \hat{b}_1(t)]$ ← Vector of adjustable parameters

Prediction error (à priori)

$$\varepsilon^o(t+1) = y(t+1) - \hat{y}^o(t+1) = \varepsilon^o(t+1, \hat{\theta}(t))$$

Criterion to be minimized (objective):

$$J(t+1) = \left[\varepsilon^o(t+1)\right]^2 = \left[\varepsilon^o(t+1, \hat{\theta}(t))\right]^2$$

?

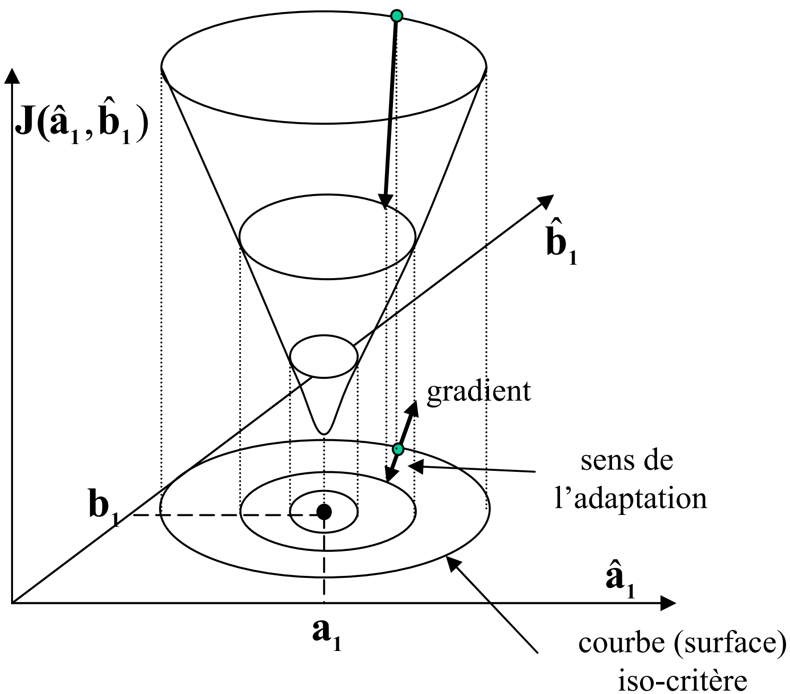
Parameter adaptation algorithm

$$\hat{\theta}(t+1) = \hat{\theta}(t) + \Delta\hat{\theta}(t+1) = \hat{\theta}(t) + f\left(\hat{\theta}(t), \phi(t), \varepsilon^o(t+1)\right)$$

PAA – Gradient algorithm

Criterion to be minimized (objective):

$$\min_{\hat{\theta}(t)} J(t+1) = \left[\varepsilon^o(t+1)\right]^2$$



gradient of the criterion

$$\hat{\theta}(t+1) = \hat{\theta}(t) - F\frac{\partial J(t+1)}{\partial \hat{\theta}(t)}$$

$F = \alpha I \quad (\alpha > 0)$ **(I = unit matrix)**

$$\frac{1}{2}\frac{\partial J(t+1)}{\partial \hat{\theta}(t)} = \frac{\partial \varepsilon^o(t+1)}{\partial \hat{\theta}(t)}\varepsilon^o(t+1)$$

$$\varepsilon^o(t+1) = y(t+1) - \hat{y}^o(t+1) = y(t+1) - \hat{\theta}(t)^T\phi(t) \quad \Rightarrow \quad \frac{\partial \varepsilon^o(t+1)}{\partial \hat{\theta}(t)} = -\phi(t)$$

$$\boxed{\hat{\theta}(t+1) = \hat{\theta}(t) + F\phi(t)\varepsilon^o(t+1)}$$

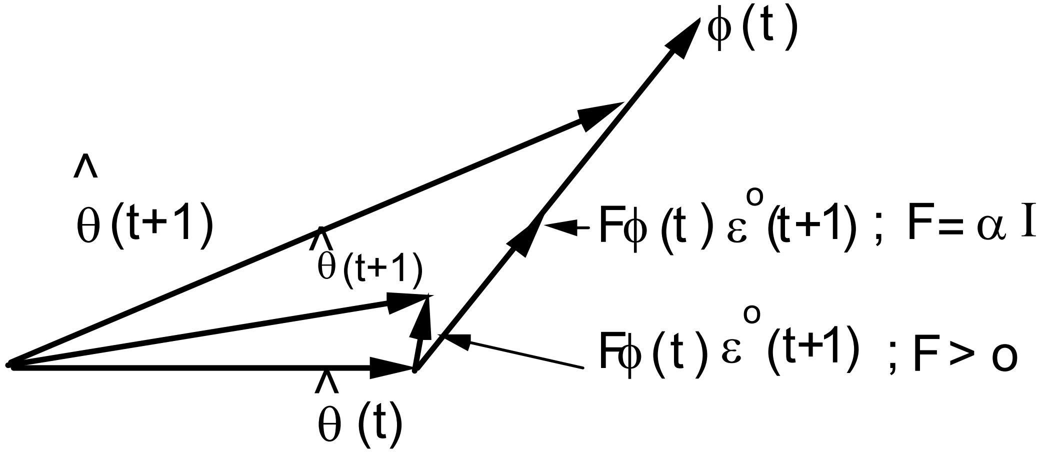
PAA – Gradient algorithm

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

Adaptation gain

$$\begin{cases} F = \alpha I & (\alpha > 0) \\ F > 0 & \textit{Positive definite matrix} \end{cases}$$

Geometrical interpretation →



Attention: Instability risk if *F (α)* is large !!

(see book Landau-Zito, pg. 213 – 214 for details)

PAA – Improved gradient algorithm

a posteriori output of the adjustable predictor

$$\hat{y}(t+1) = \hat{y}(t+1 | \hat{\theta}(t+1)) = -\hat{a}_1(t+1)y(t) + \hat{b}_1(t+1)u(t) = \hat{\theta}(t+1)^T \phi(t)$$

Prediction error (a posteriori): $\varepsilon(t+1) = y(t+1) - \hat{y}(t+1)$

Criterion to be minimized (objective):

$$\min_{\hat{\theta}(t+1)} J(t+1) = [\varepsilon(t+1)]^2$$

Gradient technique:

$$\hat{\theta}(t+1) = \hat{\theta}(t) - F \frac{\partial J(t+1)}{\partial \hat{\theta}(t)}$$

$$\frac{1}{2} \frac{\partial J(t+1)}{\partial \hat{\theta}(t+1)} = \frac{\partial \varepsilon(t+1)}{\partial \hat{\theta}(t+1)} \varepsilon(t+1)$$

$$\varepsilon(t+1) = y(t+1) - \hat{y}(t+1) = y(t+1) - \hat{\theta}(t+1)^T \phi(t) \quad \Rightarrow \quad \frac{\partial \varepsilon(t+1)}{\partial \hat{\theta}(t+1)} = -\phi(t)$$

$$\hat{\theta}(t+1) = \hat{\theta}(t) + F\phi(t)\varepsilon(t+1)$$

For implementation one should express: $\varepsilon(t+1) = f(\hat{\theta}(t), \phi(t), \varepsilon^0(t+1))$

PAA – Improved gradient algorithm

$$\varepsilon(t+1) = \underbrace{y(t+1) - \hat{\theta}(t)^T \phi(t)}_{\varepsilon^0(t+1)} - \left[\hat{\theta}(t+1) - \hat{\theta}(t)\right]^T \phi(t)$$

$$\hat{\theta}(t+1) = \hat{\theta}(t) + F\phi(t)\varepsilon(t+1) \longrightarrow \hat{\theta}(t+1) - \hat{\theta}(t) = F\phi(t)\varepsilon(t+1)$$

$$\varepsilon(t+1) = \varepsilon^0(t+1) - \phi(t)^T F\phi(t)\varepsilon(t+1) \longrightarrow \varepsilon(t+1) = \frac{\varepsilon^o(t+1)}{1+\phi(t)^T F\phi(t)}$$

$$\hat{\theta}(t+1) = \hat{\theta}(t) + \frac{F\phi(t)\varepsilon^o(t+1)}{1+\phi(t)^T F\phi(t)}$$

Stable for any *F > 0*

Implementation:

1. Before *t+1* one has: $u(t), u(t-1), .. y(t), y(t-1), \phi(t), \hat{\theta}(t), F$
2. Before *t+1* one computes : $F\phi(t)/(1+\phi(t)^T F\phi(t)),\ \hat{y}^0(t+1) = \hat{\theta}^T \phi(t)$
3. At *t+1* acquisition of *y(t+1)* and *u(t+1) is sent*
4. Running of the AAP
   (computation of : $\varepsilon^0(t+1), \hat{\theta}(t+1)$)

General form of the parameter adaptation algorithms

$$\hat{\theta}(t+1) = \hat{\theta}(t) + F(t+1)\Phi(t)\varepsilon^o(t+1)$$

$$F^{-1}(t+1) = \lambda_1(t)F^{-1}(t) + \lambda_2(t)\Phi(t)\Phi(t)^T$$

$$0 < \lambda_1(t) \le 1 \; ; \; 0 \le \lambda_2(t) < 2 \; ; \; F(0) > 0$$

Matrix inversion lemma

$$F(t+1) = \frac{1}{\lambda_1(t)}\left[F(t) - \frac{F(t)\phi(t)\phi(t)^T F(t)}{\frac{\lambda_1(t)}{\lambda_2(t)} + \phi(t)^T F(t)\phi(t)} \right]$$

$$\left(F^{-1} + \phi\phi^T\right)^{-1} = F - \frac{F\phi\phi^T F}{1 + \phi^T F\phi}$$

$$\varepsilon^o(t+1) = y(t+1) - \hat{\theta}(t)^T \phi(t)$$

$\Phi(t) - \textit{regressor vector}$

$\varepsilon^o(t+1) =$ "*a priori*" *adaptation error*

F(t) is a time varying adaptation gain (positive definite matrix).

In "self-tuning" regime *F(t)* ⟹ *0* as t increases

In "adaptive regime" *F(t)* should remain > *0*

Sequence of operations in the PAA (details)

Between t and $t+1$ $(\lambda_1(t) = \lambda_2(t) = 1)$

$$F(t+1) = F(t) - \frac{F(t)\Phi(t)\Phi(t)^T F(t)}{1+\Phi(t)^T F(t)\Phi(t)}$$

$$F(t+1)\Phi(t) = \frac{F(t)\Phi(t)}{1+\Phi(t)^T F(t)\Phi(t)}$$

$$y^\circ(t+1) = \hat{\theta}^T \phi(t)$$

At $t+1$: acquisition of $y(t+1)$

After $t+1$

$$\varepsilon^o(t+1) = y(t+1) - \hat{\theta}(t)^T \phi(t)$$

$$\hat{\theta}(t+1) = \hat{\theta}(t) + F(t+1)\Phi(t)\varepsilon^o(t+1)$$

Choice of the adaptation gain *F(t)*

- *To optimize the performances of the PAA it is useful to tune the time profile of the adaptation gain matrix*
- *To understand the influence of the tuning parameters, it is more convenient to use the equation of the "inverse" of the adaptation gain (°°)*
- *To tune the time profile of the adaptation gain in (°°), 2 parameters are introduced* $(\lambda_1(t), \lambda_2(t))$

General form:

$$F(t+1)^{-1} = \lambda_1(t)F(t)^{-1} + \lambda_2(t)\phi(t)\phi(t)^T$$
$$0 < \lambda_1(t) \le 1 \quad ; \quad 0 \le \lambda_2(t) < 2 \quad ; \quad F(0) > 0 \qquad (°°)$$

Which gives (using matrix inversion lemma):

$$F(t+1) = \frac{1}{\lambda_1(t)}\left[F(t) - \frac{F(t)\phi(t)\phi(t)^T F(t)}{\frac{\lambda_1(t)}{\lambda_2(t)} + \phi(t)^T F(t)\phi(t)}\right]$$

Choice of the adaptation gain *F(t)*

Forme générale:

$$F(t+1)^{-1} = \lambda_1(t)F(t)^{-1} + \lambda_2(t)\phi(t)\phi(t)^T$$
$$0 < \lambda_1(t) \le 1 \quad ; \quad 0 \le \lambda_2(t) < 2 \quad ; \quad F(0) > 0$$

A.1 Decreasing gain (RLS): $\lambda_1(t) = \lambda_1 = 1 \quad ; \quad \lambda_2(t) = 1$

$t \nearrow F(t)^{-1} \nearrow F(t) \searrow$

Parameter estimation (constant parameters). To be used in self-tuning regime

Forgetting factor

A.2 Fixed forgetting factor: $\lambda_1(t) = \lambda_1 \quad ; \quad 0 < \lambda_1 < 1 \quad ; \quad \lambda_2(t) = \lambda_2 = 1$

Typical values for λ_1: $\lambda_1 = 0.95,\ldots,0.99$

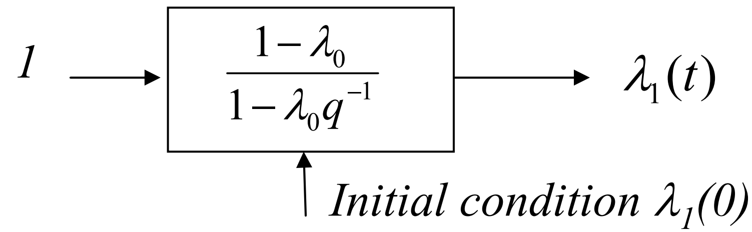
Minimized criterion: $$J(t) = \sum_{i=1}^{t} \lambda_1^{(t-i)} \left[y(i) - \hat{\theta}(t)^T \phi(i-1) \right]^2$$

One gives more weight in the criterion the last prediction error (weigtht=1).
The weight on the first prediction error is very small (weight = $\lambda^{(t-1)} << 1$ *; λ<1 and t>100)*

Parameter estimation for slowly time variyng plants
Dangerous algorithm if there is no excitation (adaptation gain goes to infinity)!

Choice of the adaptation gain *F(t)*

A.3 Variable forgetting factor: $\lambda_1(t) = \lambda_0 \lambda_1(t-1) + (1-\lambda_0)1 \;\; ; \;\; 0 < \lambda_0 < 1$



$t \to \infty ; \lambda_1 \to 1$

Asymptotically tends toward a decreasing adaptation gain

$$\lambda_2(t) = \lambda_2 = 1$$

Typical values: $\lambda_1(0) = 0.95,...,0.99 \;\; ; \;\; \lambda_0 = 0.95,...,0.99$

Minimized criterion: $J(t) = \sum_{i=1}^{t} \left[\prod_{j=1}^{t-1} \lambda_1(j-i) \right] \left[y(i) - \hat{\theta}(t)^T \phi(i-1) \right]^2$

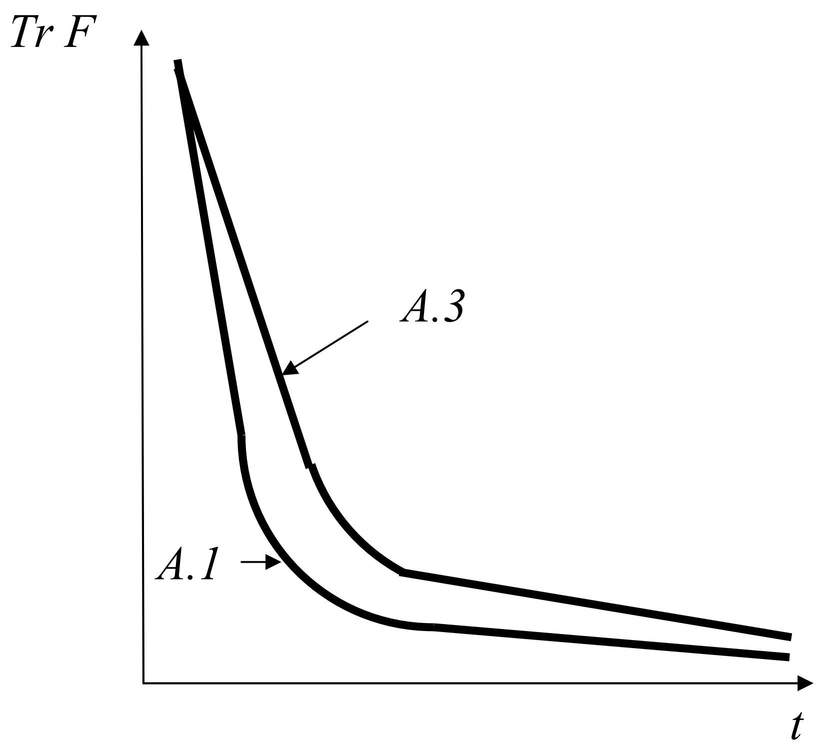
Since $\lambda_1(t)$ goes towards *1* for large *i,* one forgets intial valus

- *Parameter estimation for plants with constant parameters*
- *Use in « self-tuning regime ».*
- *Maintain a larger gain then A.1 at the beginning*
- *Gives in general better performances than A.1*

Comparison A.1/A.3

Tr F=Trace of matrix F (sum of the diagonal terms of gain matrix F)

The trace is a measure of the"gain" for the case of matrix gains



Choice of the adaption gain

A.4 Constant trace: $trF(t+1) = trF(t) = trF(0) = nGI$

$$F(0) = \begin{bmatrix} GI & & & 0 \\ & \cdot & & \\ & & \cdot & \\ 0 & & & GI \end{bmatrix}$$

n = number of parameters
GI = (0.01)0.1 to 4

$$trF(t+1) = \frac{1}{\lambda_1(t)} tr\left[F(t) - \frac{F(t)\phi(t)\phi(t)^T F(t)}{\alpha(t) + \phi(t)^T F(t)\phi(t)}\right] = trF(t)$$

One computes: $\lambda_1(t)$, $\quad$ *for* $\alpha(t) = \lambda_1(t)/\lambda_2(t)$ *fixed*

Parameter estimation for time varying systems (adaptive control regime)

A.5 Decreasing gain + constant trace

One switches from A.1 to A.4 when: $trF(t) \le nG \;\; ; \;\; G = (0.01)0.1 \text{ to } 4$

Parameter estimation of time varying systems in the absence of initial information upon the parameters (adaptive control regime)

Choice of the adaptation gain

A.6 Variable forgetting factor + constant trace

One switches from A.3 to A.4 when: $trF(t) \leq nG \quad ; \quad G = (0.01)0.1 \text{ to } 1$

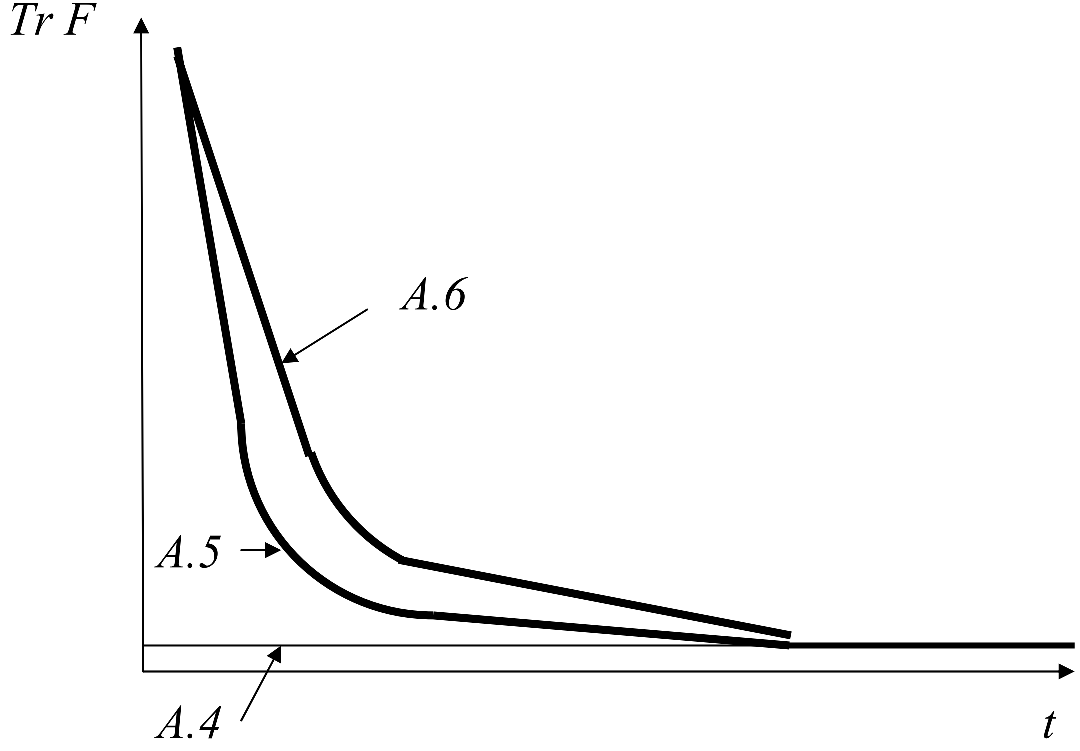
Parameter estimation of time varying systems in the absence of initial information upon the parameters (adaptive control regime)

A.7 Constant gain (improved gradient algorithm)

$$\lambda_1(t) = \lambda_1 = 1 \quad ; \quad \lambda_2(t) = \lambda_2 = 0 \quad \longrightarrow \quad F(t+1) = F(t) = F(0)$$

Estimation of systems with few parameters (≤ 3) *and low noise level.*
Simple implementation but performance inferior to A.1, A.2, A.3 and A.4 (adaptive control regime)

Comparison A.4/A.5/A.6



Choice of the initial adaptation gain *F(0)*

$$F(0) = \frac{1}{\delta} I = (GI)I$$

The adaption gain can be interpreted as a measure of the Parametric error (precision of the estimation).

Without initial information upon the parameters: $GI = 1000\,;\, \hat{\theta}(0) = 0$

Initial parameter estimation is available: $GI =<< 1\,;\, \hat{\theta}(0) = \hat{\theta}_0$

The trace of the gain matrix is a measure of the « value » of the adaptation gain

Remark:

If the trace of F(t) does not decrease significantly, in general the parameter estimation is bad.

(can happens when the excitation signals are not appropriate)

Alternative form for the PAA

$$\hat{\theta}(t+1) = \hat{\theta}(t) + F(t)\Phi(t)\varepsilon(t+1)$$

$$F(t+1) = F(t) - \frac{F(t)\Phi(t)\Phi(t)^T F(t)}{1+\Phi(t)^T F(t)\Phi(t)}$$

$$\varepsilon(t+1) = \frac{\varepsilon^0(t+1)}{1+\Phi(t)^T F(t)\Phi(t)}$$

$\varepsilon(t+1) = \varepsilon(t+1/\hat{\theta}(t+1)$ "*a posteriori*" *adaptation error*

$\varepsilon^\circ(t+1) = \varepsilon(t+1/\hat{\theta}(t))$ "*a priori*" *adaptation error*

Used for synthesis and analysis of adaptive systems

How to generate a parameter adaptation algorithm ?

Define θ^* as the "unknown" optimal value of the parameter vector
Define $\hat{\theta}(t)$ as the "adjustable" parameter vector to be tuned

One can generate a PAA if one can express a signal error $\varepsilon(t+1)$

$$\varepsilon(t+1) = H(q^{-1})\left[\theta^* - \hat{\theta}(t+1)\right]^T \Phi(t) \qquad \textbf{(x)}$$

The associated PAA is:

$$\hat{\theta}(t+1) = \hat{\theta}(t) + F(t)\Phi(t)\varepsilon(t+1)$$
$$F(t+1)^{-1} = \lambda_1(t)F(t)^{-1} + \lambda_2(t)\phi(t)\phi(t)^T$$
$$0 < \lambda_1(t) \le 1 \;\; ; \;\; 0 \le \lambda_2(t) < 2 \;\; ; \;\; F(0) > 0 \qquad \textbf{(xx)}$$
$$\varepsilon(t+1) = \frac{\varepsilon^0(t+1)}{1 + \Phi(t)^T F(t)\Phi(t)}$$

$\varepsilon(t+1) = \varepsilon(t+1/\hat{\theta}(t+1)$ "*a posteriori*" *adaptation error*
$\varepsilon^\circ(t+1) = \varepsilon(t+1/\hat{\theta}(t))$ "*a priori*" *adaptation error*

Does it always work ?

Define:

$$H'(z^{-1}) = H(z^{-1}) - \frac{\lambda}{2} \quad with \;\; 2 > \lambda \geq \max_t \lambda_2(t)$$

It works provided that:

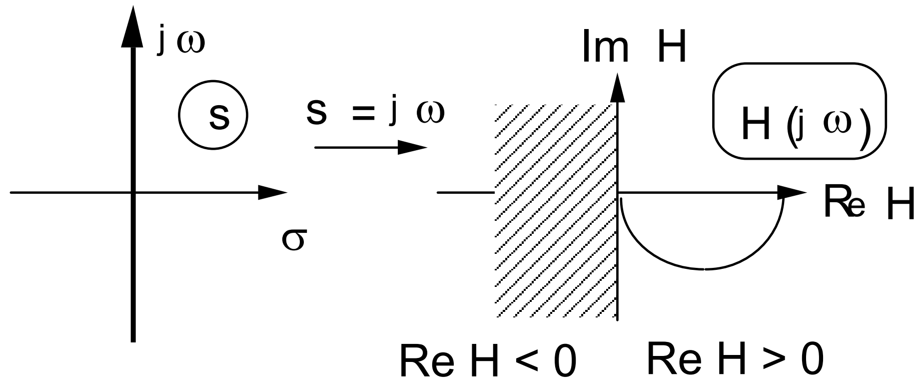
- $H'(z^{-1})$ is a *strictly positive real* discrete time transfer function
- $\Phi(t)$ is bounded

This will assure that $\varepsilon(t+1)$ and $\varepsilon^0(t+1)$ go to zero as $t \to \infty$

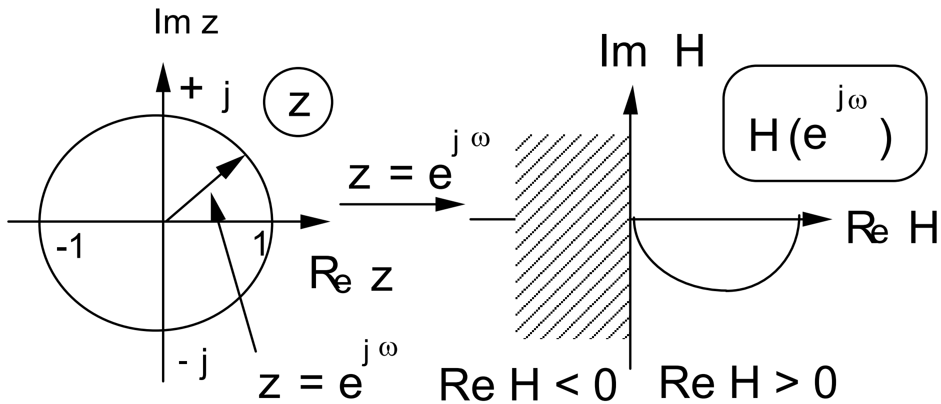
Convergence of the parameters requires in addition that *Φ(t)* is a "persistenly exciting" signal (to be discussed)

Strictly positive real transfer function (SPR)

Continuous - time



Discrete - time



- asymptotically stable
- $\operatorname{Re} H(e^{j\omega}) > 0 \; for \; all \; \left|e^{j\omega}\right| = 1, (0 < \omega < \pi)$ *(discrete-time case)*

Generation of a PAA – example

Discrete time plant model (unknown parameters))

$$y(t+1) = -a_1 y(t) + b_1(t)u(t) = \theta^T \phi(t)$$

Measurement vector

$\theta^T = [a_1, b_1]$ ← Parameter vector ; $\phi(t)^T = [-y(t), u(t)]$

Adjustable prediction model

$$\hat{y}^o(t+1) = \hat{y}(t+1|\hat{\theta}(t)) = -\hat{a}_1(t)y(t) + \hat{b}_1(t)u(t) = \hat{\theta}(t)^T \phi(t) \quad \text{a priori}$$

$$\hat{y}(t+1) = \hat{y}(t+1|\hat{\theta}(t+1)) = -\hat{a}_1(t+1)y(t) + \hat{b}_1(t+1)u(t) = \hat{\theta}(t+1)^T \phi(t) \quad \text{a posteriori}$$

$\theta(t)^T = [\hat{a}_1(t), \hat{b}_1(t)]$ ← Vector of adjustable parameters

Prediction error

$$\varepsilon^o(t+1) = y(t+1) - \hat{y}^o(t+1) = \varepsilon^o(t+1/\hat{\theta}(t))$$

$$\varepsilon(t+1) = y(t+1) - \hat{y}(t+1) = \varepsilon(t+1/\hat{\theta}(t+1))$$

But:

$$\varepsilon(t+1) = \left[\theta - \hat{\theta}(t+1)\right]^T \phi(t)$$

Is of the form (x) and we can use the PAA (xx) with $\Phi(t) = \phi(t)$

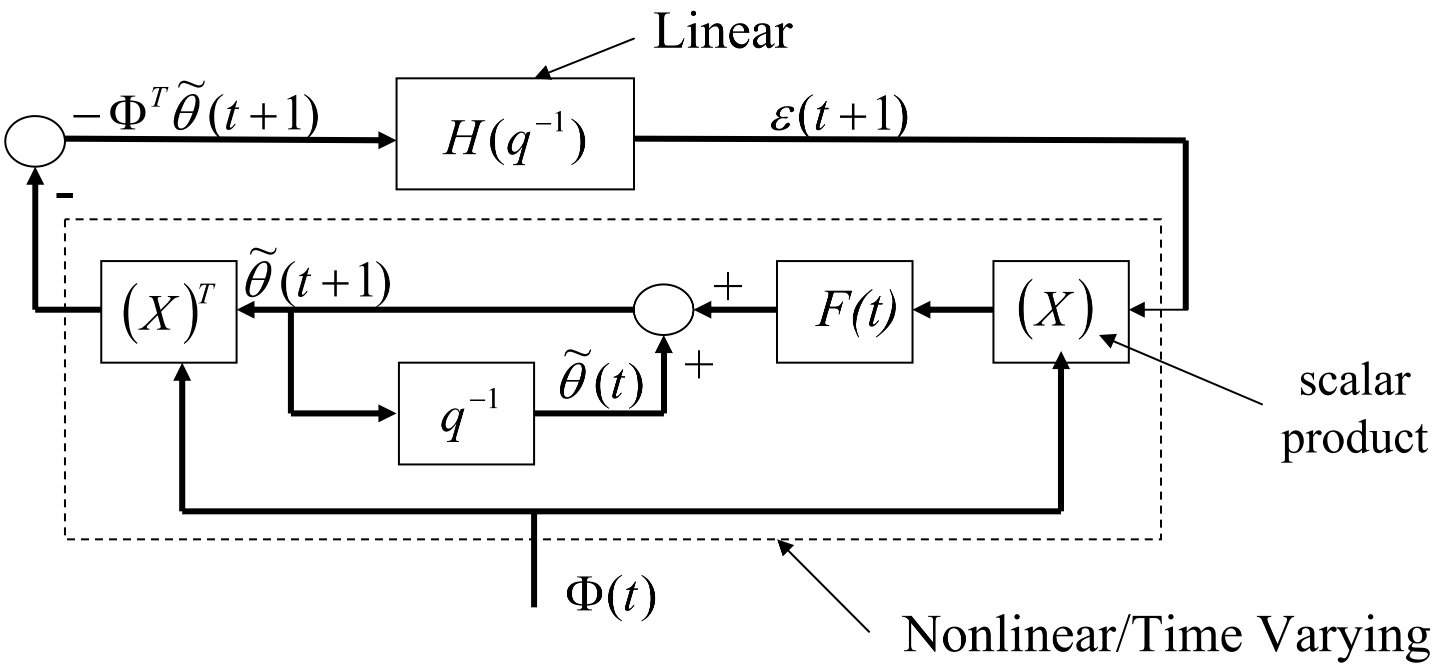
An equivalent feedback representation

Eqs (**x**) and (**xx**) allow to define an equivalent feedback system

Define the *parameter error :* $\tilde{\theta}(t) = \hat{\theta}(t) - \theta$

$$\varepsilon(t+1) = -H(q^{-1})\tilde{\theta}^{T}(t+1)\Phi(t) \qquad (\mathbf{x})$$

From (**xx**) one gets: $$\tilde{\theta}(t+1) = \tilde{\theta}(t) + F(t)\Phi(t)\varepsilon(t+1) \qquad (\mathbf{xxx})$$



The stability of the adaptive systems is studied using this equivalent representation

Convergence of estimated parameters

Convergence towards 0 (zero) of the *adaptation error* does not necessarily implies the convergence of the estimated parameter vector $\hat{\theta}$ towards the optimal value θ *!!!!*

A supplementary condition upon the input is required (*persistence of excitation*)

Persistenly exciting regressor

$\Phi(\text{t})$ is such that:

$$\left[\theta^* - \hat{\theta}\right]^T \Phi(t) = 0 \; only \; if \; \left[\theta^* - \hat{\theta}\right] = 0$$

This imply that:

- $\Phi(t)$ is different from zero
- It does not exist:

$$\Phi(t) \neq 0 \; such \; that \; it \; exists \; a \; vector \; \hat{\theta} \neq \theta^* \; for \; which \; \left[\theta^* - \hat{\theta}\right]^T \Phi(t) = 0$$

Rem.: the condition above can be converted in many cases in terms of a condition upon the excitation only.

Convergence of the parameters

« null prediction error » does not implies in all the cases « estimation of the true parameters »!!

Plant model: $$y(t+1) = -a_1 y(t) + b_1 u(t)$$

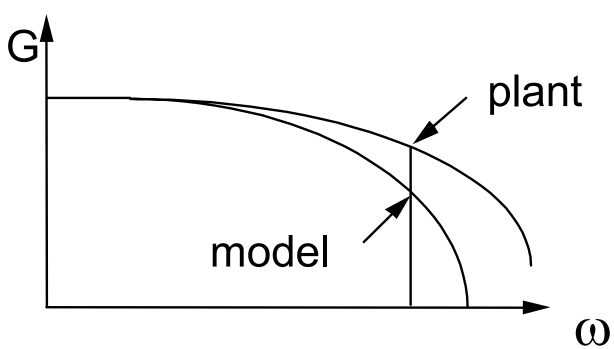
Estimated model: $$\hat{y}(t+1) = -\hat{a}_1 y(t) + \hat{b}_1 u(t)$$

$u(t) = const.$ $$\frac{b_1}{1+a_1} = \frac{\hat{b}_1}{1+\hat{a}_1}$$

The two models have the same static gain but $\hat{a}_1 \neq a_1 \,;\, \hat{b}_1 \neq b_1$

$$y(t+1) = y(t) = \frac{b_1}{1+a_1} u \quad \text{et} \quad \hat{y}(t+1) = \hat{y}(t) = \frac{\hat{b}_1}{1+\hat{a}_1} u$$

$$\varepsilon(t+1) = y(t+1) - \hat{y}(t+1) = 0 \quad \text{for } u(t) = const.;\ \hat{a}_1 \neq a_1 \,;\, \hat{b}_1 \neq b_1$$



If we would like to distinguish the two models one should apply: $u(t) = \sin(\omega t) \,;\, \omega \neq 0$

Analysis

$$\varepsilon(t+1) = y(t+1) - \hat{y}(t+1) = -[a_1 - \hat{a}_1]y(t) + [b_1 - \hat{b}_1]u(t) = 0$$

$$y(t) = \frac{b_1 q^{-1}}{1 + a_1 q^{-1}} u(t)$$

$$\varepsilon(t+1) = \left[(\hat{a}_1 - a_1) b_1 q^{-1} + (b_1 - \hat{b}_1)(1 + a_1 q^{-1}) \right] u(t) = 0$$

$$\varepsilon(t+1) = \left[(b_1 - \hat{b}_1) + q^{-1}(b_1 \hat{a}_1 - a_1 \hat{b}_1) \right] u(t) = (\alpha_0 + \alpha_1 q^{-1}) u(t) = 0 \qquad (*)$$

Solution of the recursive equation: $u(t) = z^t = e^{sT_e t}$

$$(\alpha_0 + z^{-1}\alpha_1) z^t = 0 \quad \Longrightarrow \quad z = -\frac{\alpha_1}{\alpha_0} = e^{\sigma T_e} \; ; \sigma = réel$$

$$\boxed{u(t) = e^{\sigma T_e t}}$$

$$u(t) = const \Rightarrow \sigma = 0 \Rightarrow z = 1 \Rightarrow -\alpha_1 = \alpha_0 \Rightarrow b_1 - \hat{b}_1 = a_1 \hat{b}_1 - b_1 \hat{a}_1 \Rightarrow \frac{b_1}{1 + a_1} = \frac{\hat{b}_1}{1 + \hat{a}_1}$$

Problem : find *u(t)* such that: $\varepsilon = 0 \Rightarrow \hat{a}_1 = a_1 ; \hat{b}_1 = b_1$

Answer: *u(t)* should not be a solution of (*).

An example : $u(t) = e^{j\omega T_e t} \quad or \quad e^{-j\omega T_e t} \; or \sin \omega T_e t, \omega \neq 0$

General case – choice of the excitation signal

Structure of the model to be indentified:

$$y(t) = -\sum_{i=1}^{n_A} a_i y(t-i) + \sum_{i=1}^{n_B} b_i u(t-d-i)$$

Number of the parameters to be identified: $= n_A + n_B$

Excitation signal: $u(t) = -\sum_{i=1}^{p} \sin \omega_i T_e t$

One should choose *p* such that *u(t)* can not be a solution of the recursive equation for ε whch features the parametr errors.

$$\left.\begin{array}{ll} n_A + n_B = even & p \geq \dfrac{n_A + n_B}{2} \\ n_A + n_B = odd & p \geq \dfrac{n_A + n_B + 1}{2} \end{array}\right\}$$

In order to identify correctly one should use an input « rich » in frequencies.

Standard solution: *Pseudo Random Binary Sequence (PRBS)*

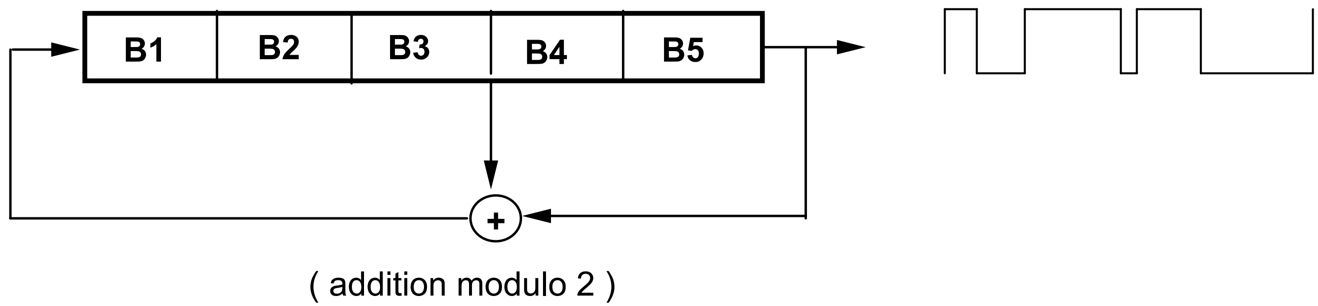
Pseudo random binary sequence (PRBS)

Sequence of rectangular pulses, modulated in width

(rich in frequencies – almost uniform spectral density from 0 to $0.5f_s$)

Generation: *shift registers connected in feedbck*

Example : generation of a PRBS of length $31=2^5-1$



Length of the sequence **: gives its periodicity.**
Random variation of the pulse width within a sequence

Characteristic parameters:

- number of cells (N)
- Maximum pulse duration ($t_{im}=Nt_e$)
- length of the sequence ($L=2^N-1$)

C++ code and .m file for generation of PRBS: see the book website

PAA – Concluding remarks

- PAA are a fundamental component of any closed loop adaptive control scheme either "direct" or "indirect"
- PAA are also used in sytem identification (recursive algorithms)
- There are several choices for the profile of the adaptation gain
- For a secure implementation the implementation of the updating algorithm for the adaption gain should use the U-D factorization (see Chapter 16)
- Convergence towards good performance does not necessarily implies that the optimal parameters have been correctly identified
- Convergence towards good performance simply implies that one has the good parameters for the specific excitation (reference signals) acting on the system
- If convergence to the optimal parameters is required, special external excitation should be applied (testing signals satisfying a persistence of excitation condition)