

# Chapitre V

## Identification des modèles de procédés : les bases

**Version 1/10.11.2002**

## **Chapitre 5. Identification des modèles de procédés : les bases**

### **5.1 Principes de l'identification des modèles de procédés**

### **5.2 Algorithmes pour l'estimation paramétrique**

#### 5.2.1 Introduction

#### 5.2.2 Algorithme du gradient

#### 5.2.3 Algorithme des moindres carrés

#### 5.2.4 Choix du gain d'adaptation

### **5.3 Choix de la séquence d'entrée pour l'identification**

#### 5.3.1 Le problème

#### 5.3.2 Séquences binaires pseudo-aléatoires (S.B.P.A.)

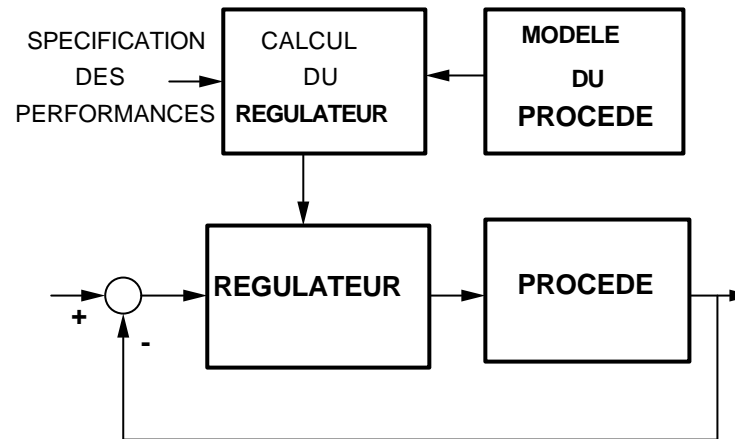
### **5.4 Effets des perturbations aléatoires sur l'identification**

### **5.5 Structure des méthodes d'identification récursives**

### **5.6 Conclusion**

### **5.7 Notes et indications bibliographiques**

# Conception et calcul d'un régulateur



Pour concevoir et ajuster un « bon » régulateur il faut :

- 1) Spécifier les performances
- 2) Connaître le *modèle* du procédé (modèle de commande)
- 3) Disposer d'une méthode appropriée de calcul

*Identification* = opération de détermination du *modèle dynamique* (de commande) d'un procédé à partir de données expérimentales

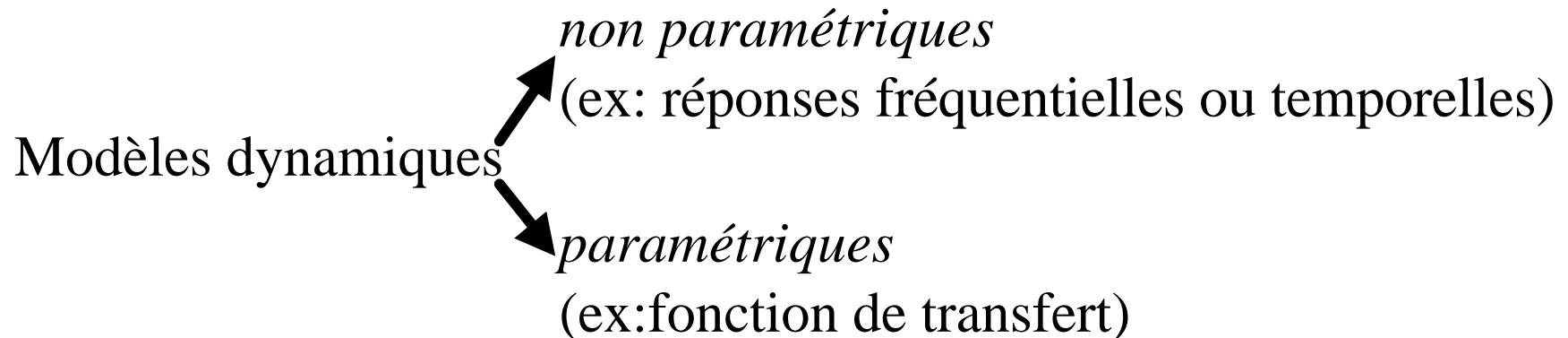
# Modélisation des procédés

## Modèles de connaissance

Utilisation: simulation et conception de procédés

## Modèles dynamiques de commande (entrée-sortie)

Utilisation : conception et calcul des régulateurs



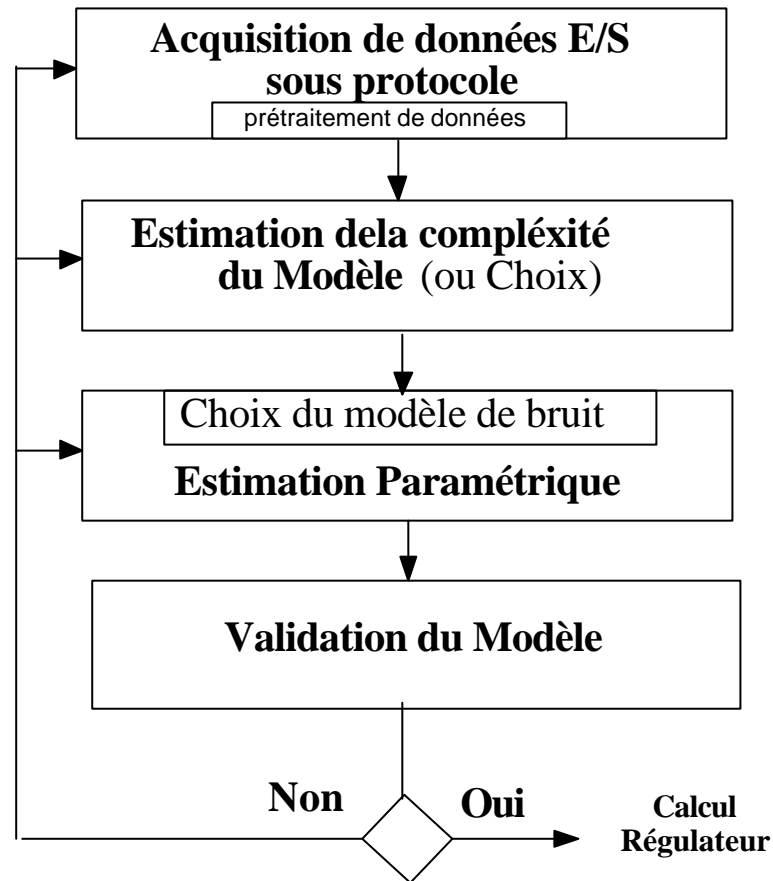
## Modèles paramétriques

*continus*

*échantillonnés*

**On s'intéresse par la suite à l'identification des modèles paramétriques échantillonnés**

# Méthodologie de l'identification

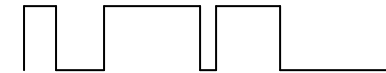


Il faut voir l'identification comme une procédure itérative dont l'objectif est l'obtention d'un modèle qui « valide »

# Identification par ordinateur numérique

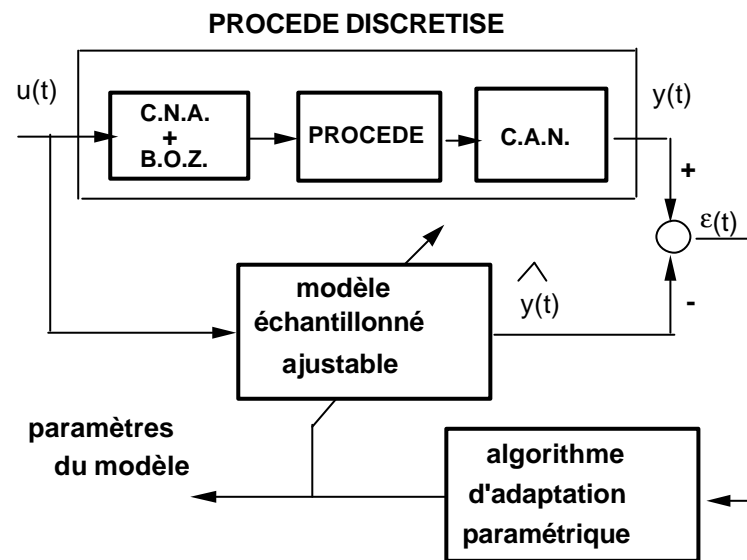
## 1) Acquisition

entrée: séquence binaire pseudo aléatoire de faible amplitude



## 2) Estimation de la complexité du modèle (algorithmes)

## 3) Estimation des paramètres du modèle (estimation récursive)



## 4) Validation du modèle identifié

Tests statistiques sur  $e(t)$  et  $\hat{y}(t)$

## Caractéristiques de l'identification par ordinateur numérique

- Signaux de tests de faible amplitude
- Excellente précision d'identification
- Critère objectifs de validation
- Possibilité de suivre les variations des paramètres du procédé
- Identification des modèles des perturbations

### Autres applications (liste non exhaustive)

- Modélisation des bruits « capteurs » (en vue de suppression)
- Détection et mesure des modes de vibrations (fréquence et amortissement)
- Analyse spectrale des signaux
- Détection préventive des pannes

# Identification (estimation) paramétrique réursive

## *Algorithme d'adaptation paramétrique (AAP)*

Vecteur des paramètres = contient l'ensemble des paramètres à identifier

$$\begin{aligned}
 \begin{bmatrix} \text{Nouvelle estimation} \\ \text{des paramètres} \\ \text{(vecteur)} \end{bmatrix} &= \begin{bmatrix} \text{Estimation précédente} \\ \text{des paramètres} \\ \text{(vecteur)} \end{bmatrix} + \\
 \begin{bmatrix} \text{Gain} \\ \text{d'adaptation} \\ \text{(matrice)} \end{bmatrix} \times \begin{bmatrix} \text{Fonction} \\ \text{des mesures} \\ \text{(vecteur)} \end{bmatrix} \times \begin{bmatrix} \text{Fonction} \\ \text{de l'erreur de prédiction} \\ \text{(scalaire)} \end{bmatrix}
 \end{aligned}$$



Vecteur des  
observations



## Identification (estimation) paramétrique

### *Estimation paramétrique non réursive:*

Traitement des entrées/sorties par paquets obtenus sur un horizon de temps.  
Pas d'estimation des paramètres pendant l'acquisition ou la lecture du fichier

### *Estimation paramétrique réursive:*

Traitement d'une paire d'entrées/sorties à chaque pas d'échantillonnage.  
(pendant l'acquisition (temps réel) ou lors de la lecture d'un fichier de données).  
Estimation des paramètres à chaque pas d'échantillonnage

### *Avantages de l'estimation paramétrique réursive*

- Obtention d'une estimation du modèle à fur et à mesure que le procédé évolue
- Compression importante des données
- Nécessite moins de puissance de calcul et de mémoire
- Mise en œuvre facile sur microprocesseurs
- Identification temps réel si nécessaire
- Possibilité de poursuite des paramètres variables dans le temps

## Algorithmes pour l'estimation paramétrique

Modèle discrétisé du procédé (paramètres inconnus)

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

$$\mathbf{q}^T = [a_1, b_1] \leftarrow \text{Vecteur des paramètres} ; \quad \mathbf{f}(t)^T = [-y(t), u(t)]$$

↙ Vecteur des mesures

Modèle ajustable de prédiction (à priori)

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

$$\hat{\mathbf{q}}(t)^T = [\hat{a}_1(t), \hat{b}_1(t)] \leftarrow \text{Vecteur des paramètres ajustables}$$

Erreur de prédiction (à priori)

$$\mathbf{e}^o(t+1) = y(t+1) - \hat{y}^o(t+1) = \mathbf{e}^o(t+1, \hat{\mathbf{q}}(t))$$

Critère à minimiser (objectif):

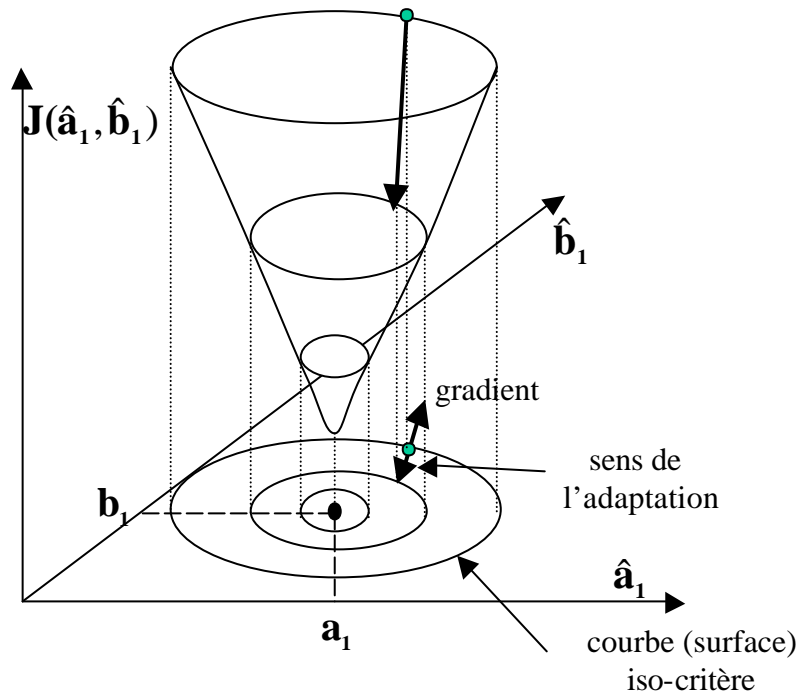
$$J(t+1) = \left[ \mathbf{e}^o(t+1) \right]^2 = \left[ \mathbf{e}^o(t+1, \hat{\mathbf{q}}(t)) \right]^2 \quad ?$$

Algorithme d'adaptation paramétrique

$$\hat{\mathbf{q}}(t+1) = \hat{\mathbf{q}}(t) + \Delta \hat{\mathbf{q}}(t+1) = \hat{\mathbf{q}}(t) + f(\hat{\mathbf{q}}(t), \mathbf{f}(t), \mathbf{e}^o(t+1))$$

# AAP – Algorithme du gradient

Critère à minimiser (objectif):



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

gradient  
du critère

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

$$F = aI \quad (a > 0) \quad (I = \text{matrice unité})$$

$$\frac{1}{2} \frac{\partial J(t+1)}{\partial \hat{\mathbf{q}}(t)} = \frac{\partial \mathbf{e}^o(t+1)}{\partial \hat{\mathbf{q}}(t)} \mathbf{e}^o(t+1)$$

$$\mathbf{e}^o(t+1) = y(t+1) - \hat{y}^o(t+1) = y(t+1) - \hat{\mathbf{q}}(t)^T \mathbf{f}(t) \quad \longrightarrow$$

$$\frac{\partial \mathbf{e}^o(t+1)}{\partial \hat{\mathbf{q}}(t)} = -\mathbf{f}(t)$$

$$\hat{\mathbf{q}}(t+1) = \hat{\mathbf{q}}(t) + F \mathbf{f}(t) \mathbf{e}^o(t+1)$$

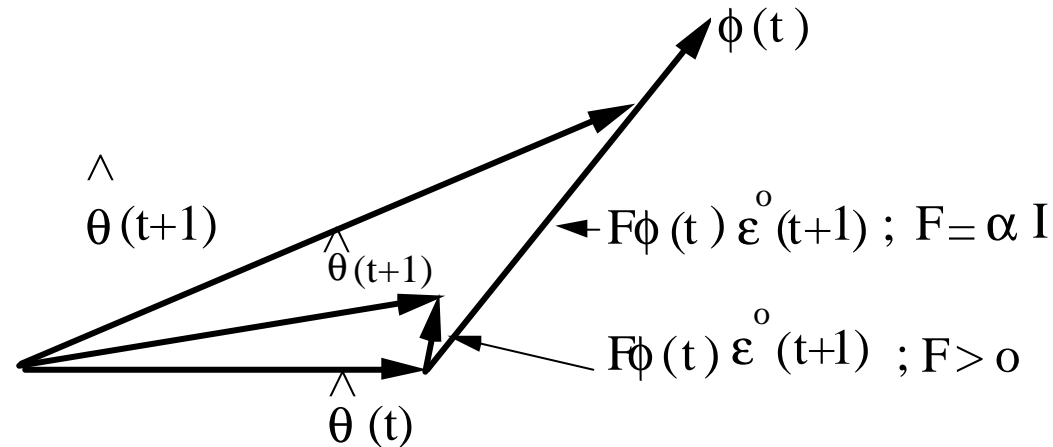
## AAP – Algorithme du gradient

$$\hat{\mathbf{q}}(t+1) = \hat{\mathbf{q}}(t) + F \mathbf{f}(t) \varepsilon^o(t+1)$$

↑  
Gain d'adaptation

$$\begin{cases} F = aI & (a > 0) \\ F > 0 & \text{Matrice définie positive} \end{cases}$$

Interprétation  
géométrique →



*Attention:* Risques d'instabilité si  $F$  ( $a$ ) est grand !!  
(voir livre pg. 266 – 267 pour les détails)

## AAP – Algorithme du gradient amélioré

Sortie *à posteriori* du prédicteur ajustable

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

Erreur de prédiction (à posteriori):  $\mathbf{e}(t+1) = y(t+1) - \hat{y}(t+1)$

Critère à minimiser (objectif):  $\min_{\hat{\mathbf{q}}(t+1)} J(t+1) = [\mathbf{e}(t+1)]^2$

Technique du gradient:  $\hat{\mathbf{q}}(t+1) = \hat{\mathbf{q}}(t) - F \frac{\partial J(t+1)}{\partial \hat{\mathbf{q}}(t)}$

$$\frac{1}{2} \frac{\partial J(t+1)}{\partial \hat{\mathbf{q}}(t+1)} = \frac{\partial \mathbf{e}(t+1)}{\partial \hat{\mathbf{q}}(t+1)} \mathbf{e}(t+1)$$

$$\mathbf{e}(t+1) = y(t+1) - \hat{y}(t+1) = y(t+1) - \hat{\mathbf{q}}(t+1)^T \mathbf{f}(t) \quad \longrightarrow \quad \frac{\partial \mathbf{e}(t+1)}{\partial \hat{\mathbf{q}}(t+1)} = -\mathbf{f}(t)$$

$$\hat{\mathbf{q}}(t+1) = \hat{\mathbf{q}}(t) + F \mathbf{f}(t) \mathbf{e}(t+1)$$

Pour la mise en œuvre il faut exprimer:  $\mathbf{e}(t+1) = f(\hat{\mathbf{q}}(t), \mathbf{f}(t), \mathbf{e}^0(t+1))$

## AAP – Algorithme du gradient amélioré

$$\mathbf{e}(t+1) = y(t+1) - \underbrace{\hat{\mathbf{q}}(t)^T \mathbf{f}(t)}_{\mathbf{e}^0(t+1)} - \left[ \hat{\mathbf{q}}(t+1) - \hat{\mathbf{q}}(t) \right]^T \mathbf{f}(t)$$

$$\hat{\mathbf{q}}(t+1) = \hat{\mathbf{q}}(t) + F\mathbf{f}(t)\mathbf{e}(t+1) \longrightarrow \hat{\mathbf{q}}(t+1) - \hat{\mathbf{q}}(t) = F\mathbf{f}(t)\mathbf{e}(t+1)$$

$$\mathbf{e}(t+1) = \mathbf{e}^0(t+1) - \mathbf{f}(t)F\mathbf{f}(t)\mathbf{e}(t+1) \longrightarrow \mathbf{e}(t+1) = \frac{\mathbf{e}^0(t+1)}{1 + \mathbf{f}(t)^T F\mathbf{f}(t)}$$

$$\hat{\mathbf{q}}(t+1) = \hat{\mathbf{q}}(t) + \frac{F\mathbf{f}(t)\mathbf{e}^0(t+1)}{1 + \mathbf{f}(t)^T F\mathbf{f}(t)}$$

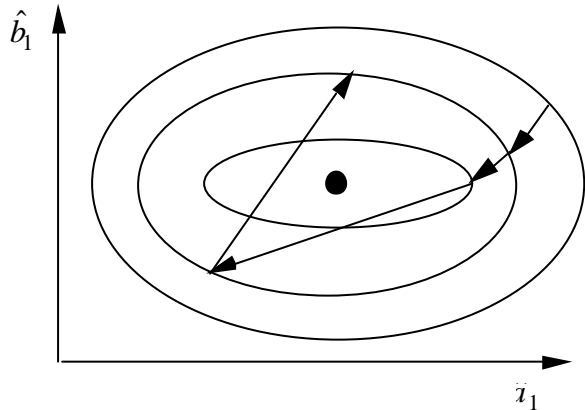
**Stable quel que soit  $F > 0$**

### *Mise en œuvre:*

1. Avant  $t+1$  on dispose de :  $u(t), u(t-1), \dots, y(t), y(t-1), \mathbf{f}(t), \hat{\mathbf{q}}(t), F$
2. Avant  $t+1$  on calcule :  $F\mathbf{f}(t)/(1 + \mathbf{f}(t)^T F\mathbf{f}(t)), \hat{y}^0(t+1)$
3. A  $t+1$  on fait l'acquisition de  $y(t+1)$  et on envoie  $u(t+1)$
4. On met en œuvre l'algorithme d'adaptation paramétrique (calcul de :  $\mathbf{e}^0(t+1), \hat{\mathbf{q}}(t+1)$ )

## Algorithmes des moindres carrés

La minimisation à chaque pas de  $e^2(t+1)$  n'entraîne pas nécessairement la minimisation de  $\sum e^2(i+1)$  sur un horizon de  $t$  pas.



Pour un bon résultat il faudrait (intuitivement):

- grand gain d'adaptation au début
- petit gain d'adaptation à la fin

*L'algorithme des moindres carrés a ces propriétés.*

Modèle discrétisé du procédé (paramètres inconnus)

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

$$\mathbf{q}^T = [a_1, b_1] \leftarrow \text{Vecteur des paramètres}$$

$$\mathbf{f}(t)^T = [-y(t), u(t)]$$

↙ Vecteur des mesures

Prédicteur ajustable *à priori*

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

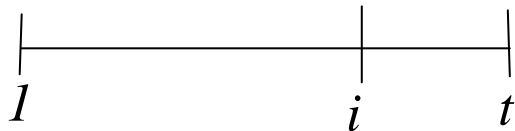
$$\mathbf{q}(t)^T = [\hat{a}_1(t), \hat{b}_1(t)] \leftarrow \text{Vecteur des paramètres ajustables}$$

## Algorithmes des moindres carrés

Critère des moindres carrés (objectif) :

$$\min_{\hat{\mathbf{q}}(t)} J(t) = \frac{1}{t} \sum_{i=1}^t \left[ y(i) - \hat{\mathbf{q}}(t)^T \mathbf{f}(i-1) \right]^2 = \frac{1}{t} \sum_{i=1}^t e^2(i, \hat{\mathbf{q}}(t))$$

$$\hat{\mathbf{q}}(t)^T \mathbf{f}(i-1) = -\hat{a}_1(t)y(i-1) + \hat{b}_1(t)u(i-1) = \hat{y}(i|\hat{\mathbf{q}}(t))$$



Prédiction de la sortie à l'instant  $i$  ( $i \ll t$ ) basée sur l'estimation des paramètres à l'instant  $t$

Solution:  $\frac{\partial J(t)}{\partial \hat{\mathbf{q}}(t)} = 0$

(voir détails livre pg.270-271)

$$\hat{\mathbf{q}}(t) = F(t) \sum_{i=1}^t y(i) \mathbf{f}(i-1)$$

$$F(t)^{-1} = \sum_{i=1}^t \mathbf{f}(i-1) \mathbf{f}(i-1)^T$$

**Algorithme non récursif !**

Voir fonction: *nrls.sci(.m)* sur le site web du livre



## Moindres carrés récursifs

$\hat{\mathbf{q}}(t) \rightarrow \hat{\mathbf{q}}(t+1)$  sans refaire tous les calculs

$$\hat{\mathbf{q}}(t) = F(t) \sum_{i=1}^t y(i) \mathbf{f}(i-1) \quad ; \quad F(t)^{-1} = \sum_{i=1}^t \mathbf{f}(i-1) \mathbf{f}(i-1)^T$$

$$\hat{\mathbf{q}}(t+1) = F(t+1) \sum_{i=1}^{t+1} y(i) \mathbf{f}(i-1) = \hat{\mathbf{q}}(t) + \Delta \hat{\mathbf{q}}(t+1) \quad \leftarrow ?$$

$$F(t+1)^{-1} = \sum_{i=1}^{t+1} \mathbf{f}(i-1) \mathbf{f}(i-1)^T = F(t)^{-1} + \mathbf{f}(t) \mathbf{f}(t)^T \quad (*)$$

↓ Calcul (pg.271-273)

$$\hat{\mathbf{q}}(t+1) = \hat{\mathbf{q}}(t) + F(t+1) \mathbf{f}(t) \mathbf{e}^o(t+1) \quad (**)$$

Version I

$$F(t+1) = F(t) - \frac{F(t) \mathbf{f}(t) \mathbf{f}(t)^T F(t)}{1 + \mathbf{f}(t)^T F(t) \mathbf{f}(t)} \quad (***)$$

$$\mathbf{e}^o(t+1) = y(t+1) - \hat{\mathbf{q}}(t)^T \mathbf{f}(t)$$

Le passage de (\*) à (\*\*\*) se fait à l'aide du « Lemme d'inversion matricielle »

Voir fonction: *rls.sci(m)* sur le site web du livre

## Moindres carrés récursifs

En remplaçant  $F(t+1)$  dans (\*\*) par (\*\*\*) on obtient:

$$F(t+1)\mathbf{f}(t)\mathbf{e}^o(t+1) = F(t)\mathbf{f}(t) \frac{\mathbf{e}^o(t+1)}{1 + \mathbf{f}(t)^T F(t)\mathbf{f}(t)}$$

Et d'autre part :

$$\mathbf{e}(t+1) = y(t+1) - \hat{\mathbf{q}}(t+1)\mathbf{f}(t) = \frac{\mathbf{e}^o(t+1)}{1 + \mathbf{f}(t)^T F(t)\mathbf{f}(t)}$$

Version II  
(pour l'analyse)

$$\hat{\mathbf{q}}(t+1) = \hat{\mathbf{q}}(t) + F(t)\mathbf{f}(t)\mathbf{e}(t+1)$$

$$F(t+1)^{-1} = F(t)^{-1} + \mathbf{f}(t)\mathbf{f}(t)^T$$

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

$$\mathbf{e}(t+1) = \frac{y(t+1) - \hat{\mathbf{q}}(t)^T \mathbf{f}(t)}{1 + \mathbf{f}(t)^T F(t)\mathbf{f}(t)} = \frac{\mathbf{e}^o(t+1)}{1 + \mathbf{f}(t)^T F(t)\mathbf{f}(t)}$$

## Moindres carrés récurrents

Gain d'adaptation initial:  $F(0) = \frac{1}{d} I = (GI)I$  ;  $0 < d \ll 1$

Valeur typique:  $GI = 1000$       *Algorithme stable quel que soit  $F(0)$*

*C'est un algorithme à gain décroissant !!*

Considérons le cas d'un seul paramètre.  $F$  et  $f$  sont des scalaires.  
Dans ce cas:

$$F(t+1) = \frac{F(t)}{1 + f(t)^2 F(t)} \leq F(t)$$

Rem.: ne peut pas suivre les variations des paramètres

## Moindres carrés récurrents

L'algorithme se généralise pour n'importe quelle dimension.

Modèle à identifier:

$$y(t) = \frac{q^{-d} B(q^{-1})}{A(q^{-1})} u(t)$$

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

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

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

$$\mathbf{q}^T = [a_1, \dots, a_{n_A}, b_1, \dots, b_{n_B}]; \quad \mathbf{f}(t)^T = [-y(t) \dots -y(t-n_A+1), u(t-d) \dots u(t-d-n_B+1)]$$

Prédicteur ajustable *à priori*:

$$\hat{y}^o(t+1) = -\sum_{i=1}^{n_A} \hat{a}_i(t) y(t+1-i) + \sum_{i=1}^{n_B} \hat{b}_i(t) u(t-d-i+1) = \hat{\mathbf{q}}(t)^T \mathbf{f}(t)$$

$$\hat{\mathbf{q}}(t)^T = [\hat{a}_1(t), \dots, \hat{a}_{n_A}(t), \hat{b}_1(t), \dots, \hat{b}_{n_B}(t)]$$

Erreur de prédiction *à priori*:

$$\mathbf{e}^o(t+1) = y(t+1) - \hat{y}^o(t+1) = y(t+1) - \hat{\mathbf{q}}(t)^T \mathbf{f}(t)$$

*On utilise l'algorithme MCR donné transparent 17*

## Choix du gain d'adaptation $F(t)$

Forme générale:

$$F(t+1)^{-1} = I_1(t)F(t)^{-1} + I_2(t)\mathbf{f}(t)\mathbf{f}(t)^T$$

$$0 < I_1(t) \leq 1 \quad ; \quad 0 \leq I_2(t) < 2 \quad ; \quad F(0) > 0$$

$$F(t+1) = \frac{1}{I_1(t)} \left[ F(t) - \frac{F(t)\mathbf{f}(t)\mathbf{f}(t)^T F(t)}{\frac{I_1(t)}{I_2(t)} + \mathbf{f}(t)^T F(t)\mathbf{f}(t)} \right]$$

A.1 Gain décroissant (MCR):  $I_1(t) = I_1 = 1 \quad ; \quad I_2(t) = 1$   $t \nearrow F(t)^{-1} \nearrow F(t) \searrow$

*Identification des systèmes stationnaires (paramètres constants)*

A.2 Facteur d'oubli fixe:  $I_1(t) = I_1 \quad ; \quad 0 < I_1 < 1 \quad ; \quad I_2(t) = I_2 = 1$

Valeurs typiques pour  $\lambda_1$ :  $I_1 = 0.95, \dots, 0.99$

Critère minimisé:  $J(t) = \sum_{i=1}^t I_1^{(t-i)} \left[ y(i) - \hat{\mathbf{q}}(t)^T \mathbf{f}(i-1) \right]^2$

*Identification des systèmes lentement variables*

## Choix du gain d'adaptation

A.3 Facteur d'oubli variable:  $I_1(t) = I_0 I_1(t-1) + 1 - I_0$  ;  $0 < I_0 < 1$

$$I_2(t) = I_2 = 1$$

Valeurs typiques:  $I_1(0) = 0.95, \dots, 0.99$  ;  $I_0 = 0.95, \dots, 0.99$

Critère minimisé: 
$$J(t) = \sum_{i=1}^t \left[ \prod_{j=1}^{t-1} I_1(j-i) \right] \left[ y(i) - \hat{\mathbf{q}}(t)^T \mathbf{f}(i-1) \right]^2$$

Comme  $I_1(t)$  tend vers 1 pour  $i$  grand, on oublie que les données initiales

*Recommandé pour l'identification des systèmes stationnaires.*

*Offre en général des meilleures performances que A.1*

## Choix du gain d'adaptation

A.4 Trace constante:

$$\text{tr}F(t+1) = \text{tr}F(t) = \text{tr}F(0) = nGI$$

$$F(0) = \begin{bmatrix} GI & & 0 \\ & \cdot & \\ 0 & & GI \end{bmatrix} \quad \begin{array}{l} n = \text{nombre de paramètres} \\ GI = (0.01)0.1 \text{ à } 4 \end{array}$$

$$\text{tr}F(t+1) = \frac{1}{\mathbf{I}_1(t)} \text{tr} \left[ F(t) - \frac{F(t)\mathbf{f}(t)\mathbf{f}(t)^T F(t)}{\mathbf{a}(t) + \mathbf{f}(t)^T F(t)\mathbf{f}(t)} \right] = \text{tr}F(t)$$



On calcule:  $\mathbf{I}_1(t)$ , pour  $\mathbf{a}(t) = \mathbf{I}_1(t) / \mathbf{I}_2(t)$  fixé

*Identification des systèmes à paramètres variables dans le temps*

A.5 Gain décroissant + trace constante

On commute de A.1 à A.4 quand:  $\text{tr}F(t) \leq nG$  ;  $G = (0.01)0.1 \text{ à } 4$

*Identification des systèmes à paramètres variables dans le temps en absence d'information initiale sur les paramètres*

## Choix du gain d'adaptation

### A.6 Facteur d'oubli variable + trace constante

On commute de A.1 à A.4 quand:  $trF(t) \leq nG$  ;  $G = (0.01)0.1 \text{ à } 4$

*Identification des systèmes à paramètres variables dans le temps en absence d'information initiale sur les paramètres*

### A.7 Gain constant (algorithme du gradient amélioré)

$$\mathbf{I}_1(t) = \mathbf{I}_1 = 1 \quad ; \quad \mathbf{I}_2(t) = \mathbf{I}_2 = 0 \quad \longrightarrow \quad F(t+1) = F(t) = F(0)$$

*Identification des systèmes à peu de paramètres ( $\leq 3$ ) et niveau de bruit réduit.  
Mise en œuvre simple mais performances inférieures à A.1, A.2, A.3, A.4*



## Choix du gain d'adaptation initial $F(0)$

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

Le gain d'adaptation peut être interprété comme une mesure de l'erreur paramétrique (précision d'estimation).

Pas d'information initiales sur les paramètres :  $GI = 1000$  ;  $\hat{\mathbf{q}}(0) = 0$

Estimation initiale des paramètres disponible :  $GI \lll 1$  ;  $\hat{\mathbf{q}}(0) = \hat{\mathbf{q}}_0$

*La trace de la matrice de gains est une mesure de la « valeur » du gain d'adaptation*

Remarque:

*Si la trace de  $F(t)$  ne décroît pas d'une façon significative l'estimation paramétrique est en général mauvaise*

(peut se produire quand les signaux d'excitation ne sont pas appropriés)

## Choix de la séquence d'entrée pour l'identification

« erreur de prédiction nulle » n'implique pas dans tous les cas  
« estimation des vrais paramètres » !!

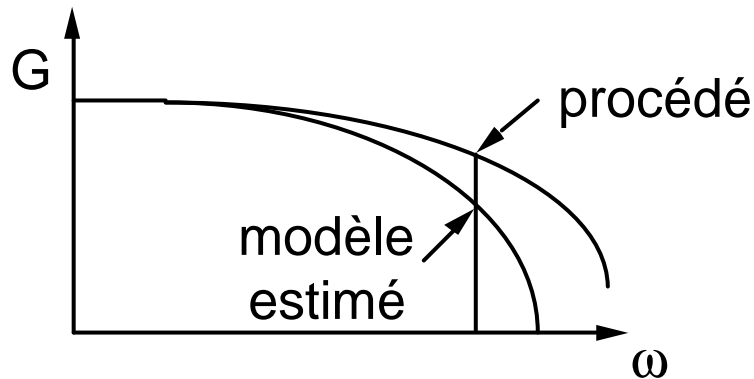
Modèle procédé:  $y(t+1) = -a_1 y(t) + b_1 u(t)$

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

$u(t) = const.$   $\frac{b_1}{1+a_1} = \frac{\hat{b}_1}{1+\hat{a}_1}$   $\leftarrow$  Les deux modèles ont le même gain statique mais  $\hat{a}_1 \neq a_1 ; \hat{b}_1 \neq b_1$

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

$e(t+1) = y(t+1) - \hat{y}(t+1) = 0$  pour  $u(t) = const ; \hat{a}_1 \neq a_1 ; \hat{b}_1 \neq b_1$



*Si on veut distinguer les deux modèles il faut appliquer:  $u(t) = \sin(\omega t) ; \omega \neq 0$*

## Analyse

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

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

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

$$\mathbf{e}(t+1) = \left[ (b_1 - \hat{b}_1) + q^{-1}(b_1 \hat{a}_1 - a_1 \hat{b}_1) \right] u(t) = (\mathbf{a}_0 + \mathbf{a}_1 q^{-1})u(t) = 0 \quad (*)$$

Solution équation récurrente:  $u(t) = z^t = e^{sT_e t}$

$$(\mathbf{a}_0 + z^{-1} \mathbf{a}_1) z^t = 0 \quad \longrightarrow \quad z = -\frac{\mathbf{a}_1}{\mathbf{a}_0} = e^{sT_e} ; \mathbf{s} = \text{réel}$$

$$u(t) = e^{sT_e t}$$

$$u(t) = \text{const} \Rightarrow \mathbf{s} = 0 \Rightarrow z = 1 \Rightarrow -\mathbf{a}_1 = \mathbf{a}_0 \Rightarrow b_1 - \hat{b}_1 = a_1 \hat{b}_1 - b_1 \hat{a}_1 \Rightarrow \frac{b_1}{1 + a_1} = \frac{\hat{b}_1}{1 + \hat{a}_1}$$

**Problème :** trouver  $u(t)$  tel que:  $\mathbf{e} = 0 \Rightarrow \hat{a}_1 = a_1 ; \hat{b}_1 = b_1$

**Réponse:**  $u(t)$  ne doit pas être une solution (\*).

**Pour l'exemple :**  $u(t) = e^{j\omega T_e t}$  ou  $e^{-j\omega T_e t}$  ou  $\sin \omega T_e t$ ,  $\omega \neq 0$

## Cas général – choix du signal d'excitation

Structure du modèle à identifier:

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

Nombre de paramètres à identifier:  $= n_A + n_B$

$$u(t) = -\sum_{i=1}^p \sin w_i T_e t$$

**Il faut choisir  $p$  tel que  $u(t)$  ne puisse pas être une solution de l'équation récurrente de  $e$  et faisant intervenir les écarts paramétriques.**

$$\left. \begin{array}{l} n_A + n_B = \textit{pair} \quad p \geq \frac{n_A + n_B}{2} \\ n_A + n_B = \textit{impair} \quad p \geq \frac{n_A + n_B + 1}{2} \end{array} \right\}$$

*Pour bien identifier il faut appliquer une entrée « riche » en fréquences.*

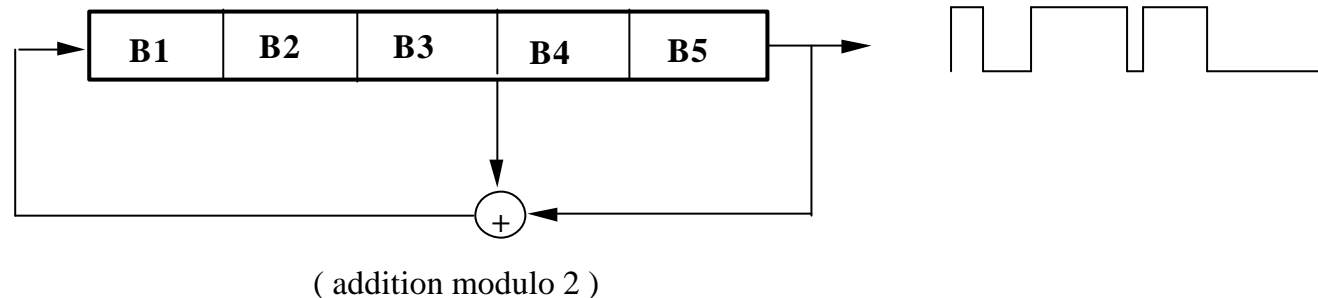
**Solution standard:** *Séquence Binaire Pseudo-Aléatoires (SBPA)*

## Séquences binaires pseudo-aléatoires (SBPA)

Succession d'impulsions rectangulaires modulées en largeur  
(très riches en fréquences – spectre uniforme de presque 0 à  $0.5f_e$ )

Génération: à l'aide de registres à décalage bouclés

Exemple : génération d'une SBPA de longueur  $31=2^5-1$



**Longueur de séquence : donne la périodicité.**

**Variation aléatoire de la largeur des impulsions à l'intérieur d'une longueur de séquence**

*Eléments caractéristiques:*

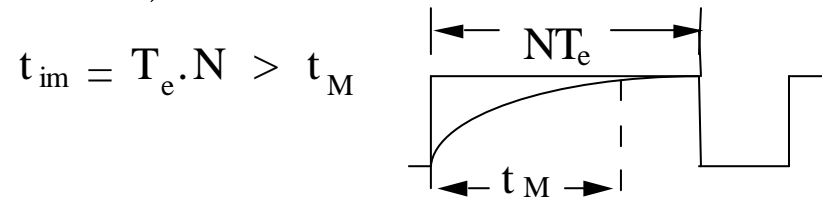
- nombres de cellules (N)
- durée maximale d'une impulsion ( $t_{im}=Nt_e$ )
- longueur de la séquence ( $L=2^N-1$ )

Programme pour la génération des SBPA: voir Annexe A.7 (pg. 533-536)

## Dimensionnement de la SBPA

Pour bien identifier le gain statique:

*Approche 1 (choix de N)*



Mais d'autre part :  $(2^N - 1)T_e = L < \text{durée de l'essai}$

$$N \rightarrow NT_e \geq t_M ; (2^N - 1)T_e \leq \text{durée de l'essai}$$

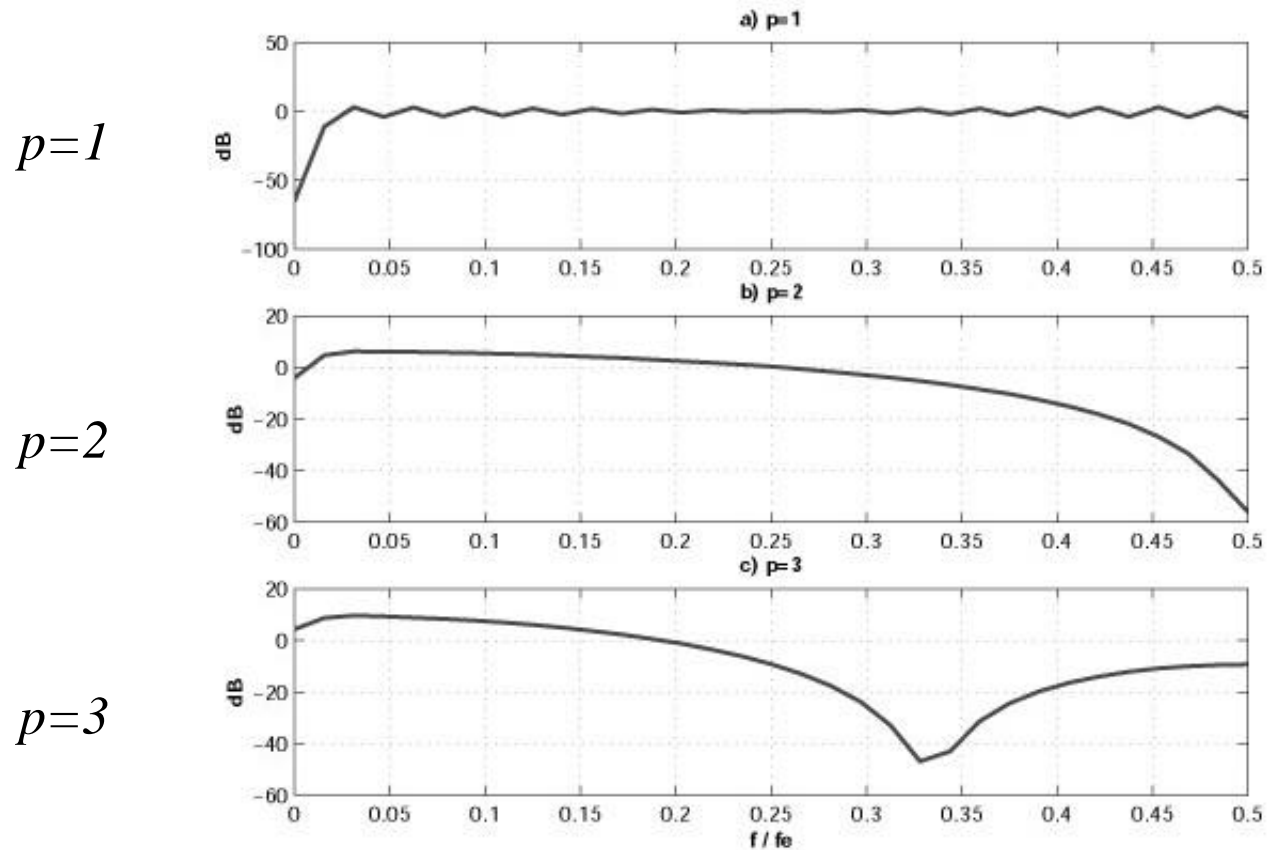
*Approche 2 (choix de N et de la fréquence d'horloge  $f_{SBPA}$ )*

$$f_{SBPA} = \frac{f_e}{p} ; p = 1, 2, 3, \dots$$

$$N, p \rightarrow pNT_e > t_M ; p(2^N - 1)T_e \leq \text{durée de l'essai}$$

**L'approche 2 permet d'obtenir une valeur plus grande de l'impulsion de durée maximale pour la même durée de l'essai (ex : prendre N+1 et respectivement p=2 et N)**

## Densité spectrale de la SBPA



L'utilisation d'un diviseur de fréquence pour  $f_{SBPA}$  augmente la densité spectrale en basses fréquences et la réduit en hautes fréquences.

## Choix de l'amplitude de la SBPA

- Amplitude faible, mais supérieure au niveau du bruit résiduel
- Rapport signal/bruit faible requiert l'allongement de l'essai
- L'augmentation excessive de l'amplitude n'est pas souhaitable (des phénomènes non-linéaires liés au procédé peuvent apparaître)

**Valeurs typiques :**

**0.5% à 10% de la valeur de la commande statique (point de fonctionnement)**



# Effets des perturbations aléatoires sur l'estimation des paramètres

La sortie mesurée des procédés est en général *bruitée* (perturbations, bruits de mesure)

*Voir Chapitre IV pour la modélisation des perturbations aléatoires*

*Effet des perturbations aléatoires sur les « moindres carrés »*

Sortie du modèle du procédé en présence de bruit:  $y(t+1) = \mathbf{q}^T \mathbf{f}(t) + w(t)$

$w(t)$  représente l'effet du bruit (stationnaire, moyenne nulle, variance finie, indép. de  $u(t)$ )

Solution des moindres carrés:  $\hat{\mathbf{q}}(N) = \left[ \sum_{t=1}^N \mathbf{f}(t-1)\mathbf{f}(t-1)^T \right]^{-1} \left[ \sum_{t=1}^N \mathbf{f}(t-1)y(t) \right]$

division et multiplication par N

$$\hat{\mathbf{q}}(N) = \mathbf{q} + \left[ \sum_{t=1}^N \mathbf{f}(t-1)\mathbf{f}(t-1)^T \right]^{-1} \left[ \sum_{t=1}^N \mathbf{f}(t-1)w(t) \right]$$

**Erreur (biais)**

$$\hat{\mathbf{q}}(N) = \mathbf{q} + \left[ \frac{1}{N} \sum_{t=1}^N \mathbf{f}(t-1)\mathbf{f}(t-1)^T \right]^{-1} \left[ \frac{1}{N} \sum_{t=1}^N \mathbf{f}(t-1)w(t) \right]$$

**Condition d'estimation asymptotiquement non biaisée**

$$\lim_{N \rightarrow \infty} \left[ \frac{1}{N} \sum_{i=1}^{N-1} \mathbf{f}(t-1)w(t) \right] = E\{\mathbf{f}(t-1)w(t)\} = 0 \quad (*)$$

## Estimation non biaisée en présence de bruit

*Il faut que  $\mathbf{f}(t-1)$  et  $w(t)$  soient non corrélés*

Pour les « moindres carrés » ceci implique  $w(t) = e(t)$  (bruit blanc).

Dans tous les autres cas : estimation biaisée (voir détails, livre pg. 290)

*Supposons :  $\hat{\mathbf{q}} = \mathbf{q}$  et on souhaite que l'algorithme laisse cette valeur inchangée*

$$\hat{y}(t+1|\mathbf{q}) = \mathbf{q}^T \mathbf{f}(t) \quad \longrightarrow \quad \mathbf{e}(t+1|\mathbf{q}) = y(t+1) - \hat{y}(t+1|\mathbf{q}) = w(t+1)$$

Condition nécessaire à satisfaire pour l'estimation non biaisée:

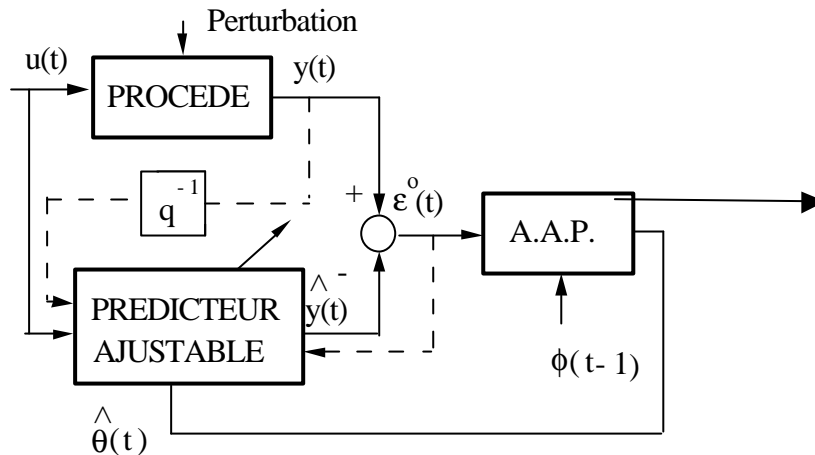
$$(*) \quad \longrightarrow \quad \lim_{N \rightarrow \infty} \left[ \frac{1}{N} \sum_{i=1}^{N-1} \mathbf{f}(t-1, \mathbf{q}) \mathbf{e}(t, \mathbf{q}) \right] = E\{\mathbf{f}(t-1, \mathbf{q}) \mathbf{e}(t, \mathbf{q})\} = 0$$

Pour éliminer le biais:  $E\{\mathbf{f}(t) \mathbf{e}(t+1)\} = 0$  pour  $\hat{\mathbf{q}} \equiv \mathbf{q}$

Modification de l'algorithme d'estimation pour obtenir:

$\mathbf{e}(t+1)$  est un bruit blanc pour  $\hat{\mathbf{q}} = \mathbf{q}$   
ou:  
 $\mathbf{f}(t)$  et  $\mathbf{e}(t+1)$  sont non corrélés (ou indépendants) pour  $\hat{\mathbf{q}} = \mathbf{q}$

## Structure des méthodes d'identification récursives



$$\hat{\mathbf{q}}(t+1) = \hat{\mathbf{q}}(t) + F(t)\Phi(t)\mathbf{e}(t+1)$$

$$F^{-1}(t+1) = \mathbf{I}_1(t)F(t) + \mathbf{I}_2\Phi(t)\Phi(t)^T$$

$$0 < \mathbf{I}_1(t) \leq 1 ; 0 \leq \mathbf{I}_2(t) < 2 ; F(0) > 0$$

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

*Éléments caractéristiques:*

- structure du prédicteur
- nature des composantes du vecteur d'observations ( $F$ )
- dimension des vecteurs des paramètres ajustables et de  $F \hat{\mathbf{q}}$
- génération de l'erreur de prédiction ( $\mathbf{e}$ )
- même structure pour l'algorithme d'adaptation paramétrique

**Types de méthodes d'identification:**

- I) Basées sur le blanchissement de l'erreur de prédiction ( $\mathbf{e}$ )
- II) Basées sur la décorrélation du vecteur  $F$  et de  $\mathbf{e}$

## Méthodes d'identification

- Il y a différentes structures « procédé + perturbation »
- Il n'y a pas une méthode d'identification unique utilisable pour toutes les structures

*Conséquence:*

En pratique il faut un système interactif pour faire de l'identification

Il doit fournir:

- différentes structures « procédé + perturbation »
- différentes méthodes d'identification
- des méthodes de validation des modèles d'identification
- un système d'acquisition et traitement des données E/S
- des outils d'analyse et de visualisation graphique

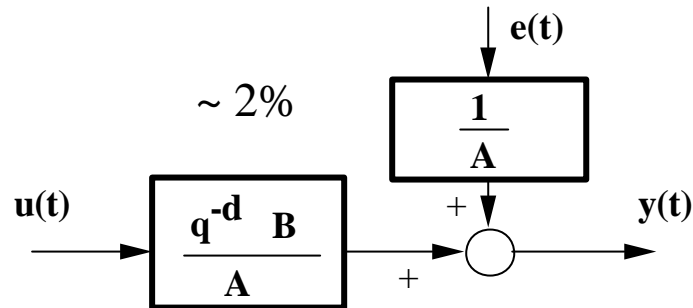
*Exemple: WinPIM (Adaptech). Voir Annexe A.5 et site web Adaptech (pg. d'accueil)*

Des routines pour l'identification (Scilab et Matlab) sont téléchargeables à partir du site web du livre ([www-lag.ensieg.inpg.fr/landau/bookIC](http://www-lag.ensieg.inpg.fr/landau/bookIC))

*Le transparent suivant donne un aperçu des principales structures « procédé + perturbation », de leur taux d'utilisation et des méthodes d'identifications appropriées*

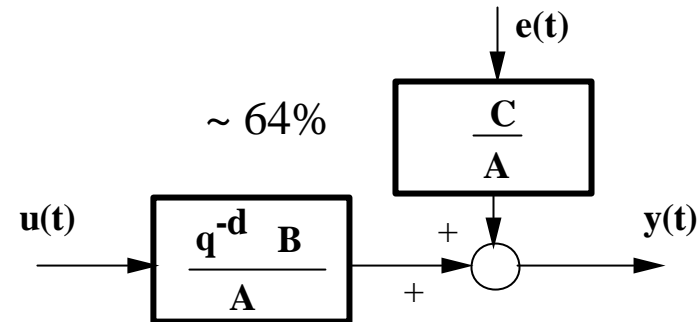
# Structures « procédé + perturbation » et méthodes d'identification

$$S1: A(q^{-1})y(t) = q^{-d}B(q^{-1})u(t) + e(t)$$



Moindres carrés récurrents (MCR)

$$S3: A(q^{-1})y(t) = q^{-d}B(q^{-1})u(t) + C(q^{-1})e(t)$$

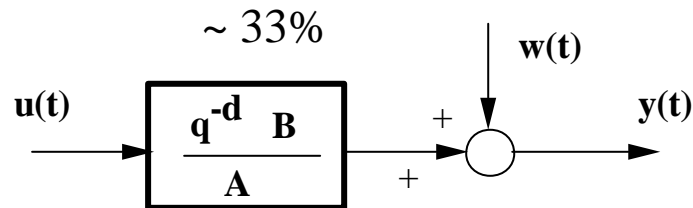


Moindres carrés étendus (MCE)

ES avec modèle de prédiction étendu (ESMPE)

Maximum de vraisemblance récursif (MVR)

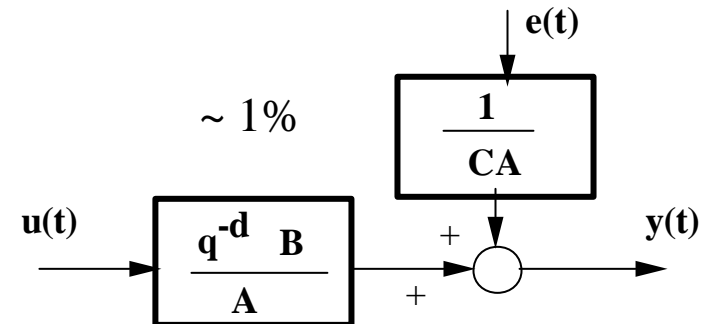
$$S2: A(q^{-1})y(t) = q^{-d}B(q^{-1})u(t) + A(q^{-1})w(t)$$



Variable instrumentale...(VIMA)

Erreur de sortie (ESCF, ESFO, ESFAO)

$$S4: A(q^{-1})y(t) = q^{-d}B(q^{-1})u(t) + [1/C(q^{-1})]e(t)$$



Moindres carrés généralisés (MCG)

## Quelques remarques récapitulatives

- L'identification comporte 4 étapes (*acquisition E/S, estimation complexité modèle, estimation des paramètres, validation*)
- L'estimation des paramètres peut se faire avec des algorithmes récursifs ou non récursifs
- L'unicité des paramètres identifiés dépend du choix de l'entrée
- Entrée standard pour l'identification :  
*Séquence binaire pseudo-aléatoire (SBPA)*
- Les perturbations et bruits sur la sortie peuvent provoquer des erreurs d'estimation (*biais*)
- Il n'y a pas une structure unique « procédé + perturbation » pour décrire toutes les situations rencontrés en pratique
- Il n'y a pas de méthode unique d'identification donnant des bons résultats pour toutes les structures « procédé + perturbation »
- Il faut disposer d'un système interactif pour l'identification