

Identification des systèmes dynamiques

FST Settat-Master ATSII

Y.ROCHDI

youssefrochdi@yahoo.fr

Identification des systèmes dynamiques

- Objectif: présenter les concepts (théoriques) et démarches générales (pratiques) permettant l'identification des systèmes dynamiques.
- Notions introduites: Modèles, identification paramétrique, non paramétrique, consistance, ...

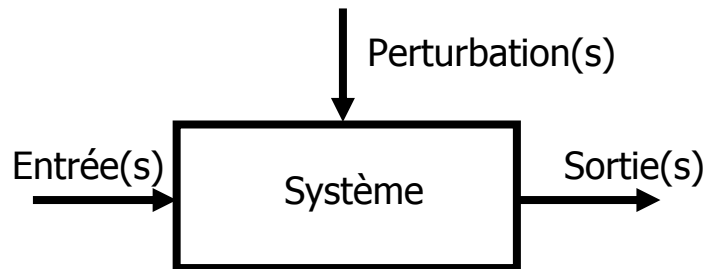
Identification des systèmes dynamiques

Chap 1 Introduction

1.1 Systèmes

- **Système dynamique**= un objet dans lequel différentes variables interagissent et produisent des signaux observables (sorties)
- **Un système est affecté par des stimulus externes:**
 - Signaux manipulables → Entrées.
 - Signaux non manipulables → Perturbations.

1.1 Systèmes



1.1 Système

■ Exemples:

- Système de chauffage d'un local: entrée= puissance électrique fournie par les résistances chauffantes / sortie: température du local / perturbations: échanges thermiques avec le milieu extérieur via les fenêtres, les portes, l'apport thermique des personnes...
- Contrôle de gouvernail d'un avion: entrée force hydraulique/ perturbations: le vent, la neige .../sortie: position du gouvernail.

1.2 Modèles –définition(1)

- Un modèle d'un système traduit les relations entre les différentes variables de ce système.
- Différentes manières de décrire ces relations → différentes classes de modèle.
- Ces classes diffèrent par le processus d'obtention du modèle et par la manière d'exploiter ce modèle.

1.2 Modèles –définition(2)

- Modèle mental: aucune formulation mathématique; pour conduire une voiture: manipulation du volant/ mémoire musculaire(direction assistée/non assistée).
Obtenu de manière heuristique.
- Modèle graphique: description du système par des tables numériques/ graphes et abaque; cas des systèmes linéaires (réponses imp/ind/fréq).
Obtenu par des essais/observations/heuristique

1.2 Modèles –définition(3)

- ❑ Modèle mathématique: description de système en termes d'expressions mathématiques comme des équations différentielles/ aux différences, le type d'expressions utilisées donne des qualificatifs au modèle: linéaire /non linéaire, continu/discret, déterministe/stochastique...

Obtenu par formulation mathématiques des relations entre les différentes variables.

1.2 Modèles -Intérêts

■ Un modèle est utilisé :

- ❑ Dans la simulation; Exple simulateur d'avions.
- ❑ Dans la prédiction des sorties; Exple :prévision d'évolution des actions dans le marché boursier/prévision météorologique/commande prédictive.
- ❑ Pour la synthèse des régulateurs; Exple: contrôle de procédés industriels.

1.2 Modèles –Conception

- Un modèle mathématique est obtenu via deux approches :
 - Par modélisation: modèle de connaissances/ modèle boîte grise (Grey-box).
 - Par identification: modèle de comportement entrées-sorties (commande)/modèle boîte noire (Black-box) .

1.2.1 Modèles de connaissances

- Le modèle de connaissances:

Exploitant les connaissances a priori sur le système, les lois du domaine et la décomposition en des sous systèmes simples, on déduit les relations mathématiques. Aucune expérimentation n'est nécessaire.

Approche utilisée pour obtenir un modèle mathématique/graphique.

Exemple: Contrôle de la vitesse d'un ventilateur entraîné par une MCC commandé par un Hacheur série...

1.2.2 Modèles de comportement

■ Le modèle de comportement:

Exploitant uniquement les mesures prélevées sur les entrées/sorties et éventuellement les expériences précédentes. Les connaissances a priori ne sont pas nécessaires (trop complexes ou indisponibles) en général.

Approche utilisée pour obtenir un modèle mathématique/graphique/mental.

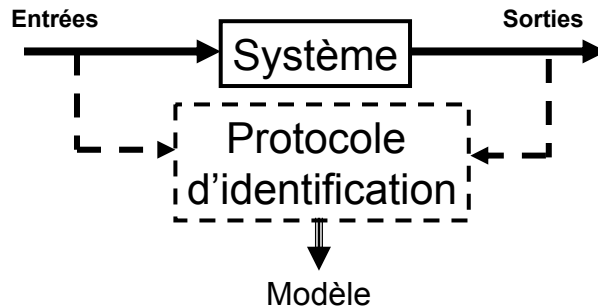
Exemple: Système de climatisation d'un grand centre commercial / conduite d'une bicyclette.

1.2.3 Modèles -Comparaison

Modèle	Connaissances	Comportements
Effort d'obtention	Plus élevé (systèmes complexes/spécialité)	Moins élevé (spécialité non nécessaire)
Précision	Plus précis et complet	Moins précis et moins complet
Interprétation physique	(Souvent) possible	Pas toujours possible
Exploitation	Moins efficace (si modèle trop complexe)	Plus efficace

1.3 Identification –Définition(1)

- L'identification est l'opération expérimentale qui consiste à déterminer un modèle à partir du comportement entrée/sortie du système.



1.3 Identification –Définition(2)

- Le modèle obtenu doit reproduire au mieux le comportement du système, dans toutes les conditions utiles du fonctionnement du système.

Exemple: un système linéaire 3^{ème} ordre avec un pôle dominant, peut être modélisé par un premier ordre en basse fréquence; le système peut être instable alors que le modèle est toujours stable !! Ce ci s'explique par le fait que le modèle ne tient pas comptes des autres pôles.

1.4 Identification – Validation

- Comment savoir qu'un modèle obtenu est « Assez bon » ?

Par rapport

- Aux données observées et utilisées pour l'identification
 - Aux connaissances a priori (retard, ordre, ...)
 - au cadre d'utilisation (bande de fréquences, niveaux d'entrées, rapport signal/bruit...)
- procédures de validation du modèle.

1.5 Identification – Type(1)

- Le modèle peut être:

- Paramétrique: modèle caractérisé par un ensemble fini de paramètres

Fonction de transfert $G(p)=B(p)/A(p)$

Représentation d'état $dx/dt= Ax+ Bu$

$$y=Cx+Du$$

Equations aux différences: $\sum a_i y(t-i)= \sum b_i u(t-i)$

- Non Paramétrique: Réponse indicielle /impulsionnelle ($g(t)$) / fréquentielle ($G(j\omega)$).

1.5 Identification –Type(2)

■ Deux types d'identification:

- Paramétrique: procédure en ligne (temps réel), se prête mieux à une commande adaptative; Peut s'effectuer alors que le système est en exploitation. La plus utilisée.
- non paramétrique: souvent procédure hors ligne, nécessite des essais particuliers, le système doit être hors exploitation. La moins utilisée.

1.6 Identification –Procédure(1)

■ L'identification se base sur 3 entités fondamentales:

- Un ensemble de données (mesures E/S).
- Un ensemble de modèles candidats: une classe de modèles.
- Un algorithme d'identification qui permet d'obtenir le meilleur modèle dans une classe à partir des données disponibles.

1.7 Identification –Procédure(2)

■ Un ensemble de données (mesures E/S).

Les données doivent contenir suffisamment d'informations concernant le système (richesse de l'excitation). Le choix (quand c'est possible) des conditions d'expérimentation est crucial:

le type de signaux d'entrées utilisés (amplitudes, durées, spectre fréquentiel...), la cadence des mesures ...

1.7 Identification –Procédure(3)

■ Une Classe de modèles

Plusieurs classes de modèles existent: linéaires/non linéaires, déterministes/stochastiques...

La classe de modèle limite l'espace de recherche en spécifiant une forme des relations entre les E/S:

Par exemple l'ordre du modèle, le point d'action des perturbations (en entrée ou en sortie), la nature des perturbations, ...

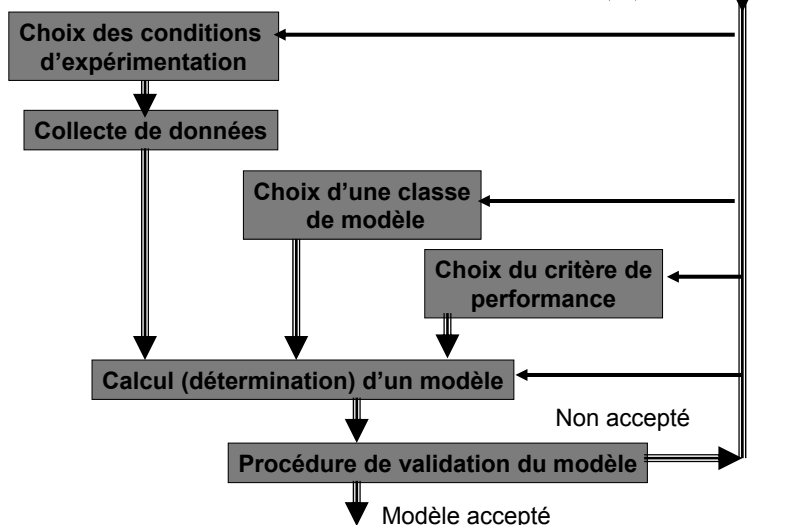
Le choix de cette classe n'est pas tjrs évident, et nécessite quelque connaissances a priori sur le système, sinon procéder par essai/erreur

1.7 Identification –Procédure(4)

■ Un algorithme d'identification

Méthode qui se base sur la classe de modèle choisie et les données collectées pour trouver le meilleur modèle de cette classe au sens d'un critère de performance choisi d'avance: par exemple minimiser l'erreur de sortie (différence entre la sortie du modèle et celle du système), ...

1.7 Identification –Procédure(5)



1.8 Étude d'un cas typique(1)

■ Le modèle d'un système discret linéaire invariant

$$y(t) + a_1 y(t-1) + \dots + a_n y(t-n) = b_1 u(t-1) + b_2 u(t-2) + \dots + b_m u(t-m) \quad (1.1)$$

t représente le temps discret, $t=kT_e$, la période d'échantillonnage est supposée $T_e=1$ (pour simplifier les notations).

Sous forme compacte, ou forme de régression:

1.8 Étude d'un cas typique(2)

■ Forme de régression linéaire

$$y(t) = \varphi^T \theta \quad (1.2)$$

$$\varphi^T = [-y(t-1) \quad \dots \quad -y(t-n) \quad u(t-1) \quad \dots \quad u(t-m)] \quad (1.3a)$$

$$\theta = [a_1 \quad \dots \quad a_n \quad b_1 \quad \dots \quad b_m]^T \quad (1.3b)$$

Vecteur de paramètre

**Vecteur de
régression/observations
Régresseur**

1.8 Étude d'un cas typique(3)

- Si θ est connu on peut prédire à l'instant $(t-1)$ $y(t)$ à partir des valeurs passées de y et u .

$$\hat{y}(t|\theta, t-1) = \varphi^T \theta \quad (1.4)$$

- Si θ est inconnu, comment le déterminer ?

Si on dispose d'une collection de données:

$$Z^N = [u(1), \quad y(1), \quad \dots, \quad u(N), \quad y(N)] \quad (1.5)$$

L'estimé de θ peut être calculé par la méthode des moindres carrés (LS: Least squares).../..

1.8 Étude d'un cas typique(4)

$$\min_{\theta} V^N(\theta, Z^N) \quad (1.6)$$

$$V^N(\theta, Z^N) = \frac{1}{N} \sum_{t=1}^N (y(t) - \hat{y}(t|\theta, t-1))^2 \quad (1.7)$$

θ sera l'argument qui minimise, au sens des moindres carrés, la différence entre la sortie prédite (modèle) et la sortie mesurée (système). L'estimé du vecteur de paramètre

$$\hat{\theta}_N = \arg(\min_{\theta} V^N(\theta, Z^N)) \quad (1.8)$$

On démontre ...

$$\hat{\theta}_N = \left[\sum_{t=1}^N \varphi(t) \varphi^T(t) \right]^{-1} \sum_{t=1}^N \varphi(t) y(t) \quad (1.9)$$

1.8 Étude d'un cas typique(5)

La résolution de cette équation peut être faite par programmation

$$\hat{\theta}_N = \left[\sum_{t=1}^N \varphi(t) \varphi^T(t) \right]^{-1} \sum_{t=1}^N \varphi(t) y(t) \quad (1.9)$$

Un exemple simple

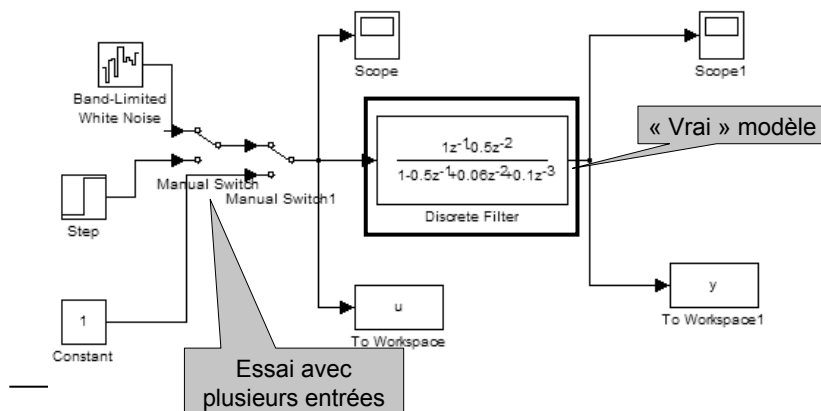
$$y(t) + ay(t-1) = bu(t-1)$$

$$\begin{bmatrix} \hat{a}_N \\ \hat{b}_N \end{bmatrix} = \begin{bmatrix} \sum_{t=1}^N y^2(t-1) & -\sum_{t=1}^N y(t-1)u(t-1) \\ -\sum_{t=1}^N y(t-1)u(t-1) & \sum_{t=1}^N u^2(t-1) \end{bmatrix}^{-1} \times \begin{bmatrix} -\sum_{t=1}^N y(t)y(t-1) \\ \sum_{t=1}^N y(t)u(t-1) \end{bmatrix}$$

$$y(0)=0$$

1.8 Étude d'un cas typique(6)

La résolution de l'équation (1.9) peut être faite par calculateur (toolbox Matlab)



1.8 Étude d'un cas typique(7)

```
>> data = iddata(y.signals.values,u.signals.values,0.1)
```

Time domain data set with 101 samples.
Sampling interval: 0.1

Construction d'un
objet data

Outputs Unit (if specified)
y1

Inputs Unit (if specified)
u1

Estimation du
modèle

```
>> m=arx(data, [3 2 1])
```

Discrete-time IDPOLY model: $A(q)y(t) = B(q)u(t) + e(t)$

$A(q) = 1 - 0.5 q^{-1} + 0.06 q^{-2} + 0.1 q^{-3}$

$B(q) = q^{-1} - 0.5 q^{-2}$

Indicateur de la
qualité de l'estimé
obtenu

Estimated using ARX from data set data
Loss function 6.45746e-032 and FPE 7.13011e-032
Sampling interval: 0.1

```
>>
```

1.8 Étude d'un cas typique(8)

A travers cet exemple simple, et en faisant plusieurs essais on voit l'influence:

- De la structure du modèle: ordres, retard.
- Du type du signal d'excitation choisi.
- Du nombre de données utilisées.

Cette influence sera plus accentuée pour des modèles plus complexes et surtout en présence de perturbations non mesurables.

Identification des systèmes

Chap 2 Identification paramétrique

2.1 Introduction(1)

- On s'intéresse aux systèmes mono entrée/sortie (SISO), linéaires, discrets et invariants.
- L'entrée de commande est $u(t)$, la sortie est $y(t)$ et le système est sous l'influence d'une perturbation $e(t)$.

2.1 Introduction(2)

- L'identification paramétrique se base sur l'estimation d'un certain nombre d'un paramètres d'un modèle choisi au préalable.
- Les connaissances a priori sur le système à identifier peuvent guider ce choix, sinon on procède par essai erreur.

2.2 Structures de modèle(1)

- Il existe différentes structures de modèle.
- La différence réside principalement dans la modélisation de la perturbation et la présence ou non d'une entrée externe
- Comme l'identification se fait en général par ordinateur, les mesures prélevées sont des échantillons, et le modèle est discret même si le système est continu

2.2 Structures de modèle(2)

- AR Model: auto regressive

$$A(q^{-1})y(t) = e(t)$$

- ARX Model: auto regressive with eXogenous input

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

- ARMAX Model: auto regressive, moving average with eXogenous input

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

- Output error Model

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

- Box-Jenkins Model

$$y(t) = \frac{B(q^{-1})}{A(q^{-1})}u(t) + \frac{C(q^{-1})}{D(q^{-1})}e(t)$$

2.2 Structures de modèle(3)

- t représente le temps discret.
- L'opérateur q^{-1} introduit un retard d'une période d'échantillonnage.
- Les coefficients des polynômes en q^{-1} sont les paramètres à estimer.
- $e(t)$ est une perturbation de type bruit blanc (processus stochastique non prédictible) de moyenne nulle.
- Les polynômes C,D permettent de décrire des perturbations à moyenne non nulle (moyenne mobile).

$$q^{-1}x(t) = x(t-1)$$

$$A(q^{-1}) = 1 + \sum_{i=1}^{na} a_i q^{-i}$$

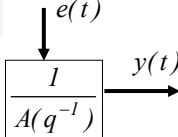
$$B(q^{-1}) = \sum_{i=1}^{nb} b_i q^{-i+1-nk}$$

$$C(q^{-1}) = 1 + \sum_{i=1}^{nc} c_i q^{-i}$$

$$D(q^{-1}) = 1 + \sum_{i=1}^{nd} d_i q^{-i}$$

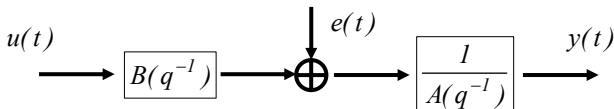
2.3.1 Modèle AR

- Ce modèle se caractérise par l'absence d'une entrée de commande → il est plutôt adapté à la caractérisation de signaux et non pas de systèmes.
- Le nombre de paramètres à estimer $= na$.

$$y(t) = a_1 y(t-1) + \dots + a_{na} y(t-na) + e(t)$$


2.3.2 Modèle ARX

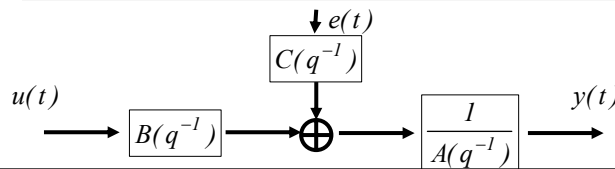
- Ce modèle suppose l'existence d'une entrée de commande, et un modèle de bruit blanc à moyenne nulle (fixe).
- Le nombre de paramètres à estimer $= na+nb$ et le retard nk

$$y(t) = a_1 y(t-1) + \dots + a_{na} y(t-na) + b_1 u(t-nk) + \dots + b_{nb} u(t-nk-nb+1) + e(t)$$


2.3.3 Modèle ARMAX

- Ce modèle suppose l'existence d'une entrée de commande, et un modèle de bruit blanc filtré, à moyenne mobile (moving average).
- Le nombre de paramètres à estimer $=na+nb+nc$ et le retard nk

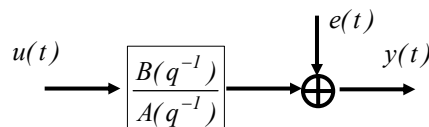
$$y(t) = a_1 y(t-1) + \dots + a_{na} y(t-na) + b_1 u(t-nk) + \dots + b_{nb} u(t-nk-nb+1) + e(t) + c_1 e(t-1) + \dots + c_{nc} e(t-nc)$$



2.3.4 Modèle à erreur de sortie

- Ce modèle suppose l'existence d'une entrée de commande, et un modèle de bruit blanc agissant directement sur la sortie
- Le nombre de paramètres à estimer $=na+nb$ et le retard nk

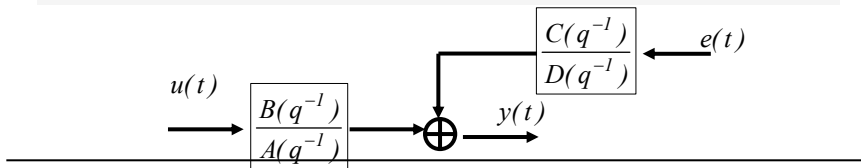
$$y(t) = a_1 y(t-1) + \dots + a_{na} y(t-na) + b_1 u(t-nk) + \dots + b_{nb} u(t-nk-nb+1) - e(t) - a_1 e(t-1) - \dots - a_{na} e(t-na)$$



2.3.5 Modèle Box-Jenkins

- Ce modèle suppose l'existence d'une entrée de commande, et un modèle de bruit filtré. La dynamique du bruit peut être différente de celle de l'entrée.
- Le nombre de paramètres à estimer $=na+nb+nc+nd$ et le retard nk

si $A(q^{-1}) = D(q^{-1})$. $y(t) = a_1 y(t-1) + \dots + a_{na} y(t-na) + b_1 u(t-nk) + \dots + b_{nb} u(t-nk-nb+1) + e(t) + c_1 e(t-1) + \dots + c_{nc} e(t-nc)$



Model Structure	Discrete-Time Form	Noise Model
ARX	$A(q)y(t) = \sum_{i=1}^{nu} B_i(q)u_i(t-nk_i) + e(t)$	The noise model is $\frac{1}{A}$ and the noise is coupled to the dynamics model. ARX does not let you model noise and dynamics independently. Use ARX to have a simple model at good signal-to-noise ratios.
ARMAX	$A(q)y(t) = \sum_{i=1}^{nu} B_i(q)u_i(t-nk_i) + C(q)e(t)$	Extends the ARX structure by providing more flexibility for modeling noise using the C parameters (a Moving Average of white noise). Use ARMAX when the dominating disturbances enter at the input. Such disturbances are called <i>load disturbances</i> .
Box-Jenkins (BJ)	$y(t) = \sum_{i=1}^{nu} \frac{B_i(q)}{F_i(q)} u_i(t-nk_i) + \frac{C(q)}{D(q)} e(t)$	Provides completely independent parameterization for the dynamics and the noise using rational polynomial functions. Use BJ models when the noise does not enter at the input, but is primary a measurement disturbance. This structure provides additional flexibility for modeling noise.
Output-Error (OE)	$y(t) = \sum_{i=1}^{nu} \frac{B_i(q)}{F_i(q)} u_i(t-nk_i) + e(t)$	Use when you want to parameterize dynamics, but do not want to estimate a noise model. Note The white noise source $e(t)$ to $H=1$ in the general equation.

2.3.7 Modèle général

- Le modèle du système peut être écrit sous forme générale:

$$y(t) = G(q^{-1})u(t) + H(q^{-1})e(t)$$

- Deux cas:
 - H et G ont des pôles communs, l'entrée de commande et la perturbation ont des dynamiques communes.
 - H et G n'ont pas de pôles communs, les dynamiques de l'entrée de commande et celle de la perturbation sont découplées (plus réaliste).

2.3.8 Modèle à erreur d'équation(1)

- Modèle à erreur d'équation :

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

- $\varepsilon(t)$: erreur d'équation, dépend de la perturbation de $e(t)$.
- $\varepsilon(t)$ comme $e(t)$ est un processus aléatoire; son expression dépend du point d'action de la perturbation $e(t)$.
- $\varepsilon(t)$ est une version filtrée d'un bruit blanc $e(t)$ de moyenne et variance

$$E(e(t)) = 0; \text{Var}(e(t)) = \sigma^2$$

2.3.8 Modèle à erreur d'équation(2)

- AR /ARX Model:

$$\varepsilon(t) = e(t)$$

$$G(q^{-1}) = \frac{B(q^{-1})}{A(q^{-1})}; H(q^{-1}) = \frac{1}{A(q^{-1})}$$

- ARMAX Model:

$$\varepsilon(t) = C(q^{-1})e(t)$$

$$G(q^{-1}) = \frac{B(q^{-1})}{A(q^{-1})}; H(q^{-1}) = \frac{C(q^{-1})}{A(q^{-1})}$$

2.3.9 Modèle à erreur de sortie(1)

- Modèle à erreur de sortie :

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

- $\varepsilon(t)$: erreur d'équation, dépend de la perturbation de $e(t)$. $\varepsilon(t)$ comme $e(t)$ est un processus aléatoire; son expression dépend du point d'action de la perturbation $e(t)$.
- $\varepsilon(t)$ est une version filtrée d'un bruit blanc $e(t)$ de moyenne et variance

$$E(e(t)) = 0; Var(e(t)) = \sigma^2$$

2.3.9 Modèle à erreur de sortie(2)

- Output error Model:

$$\varepsilon(t) = A(q^{-1})e(t)$$

$$G(q^{-1}) = \frac{B(q^{-1})}{A(q^{-1})}; H(q^{-1}) = I$$

- Box-Jenkins Model

$$\varepsilon(t) = A(q^{-1}) \frac{C(q^{-1})}{D(q^{-1})} e(t)$$

$$G(q^{-1}) = \frac{B(q^{-1})}{A(q^{-1})}; H(q^{-1}) = \frac{C(q^{-1})}{D(q^{-1})}$$

2.4 Forme de régression et Prédiction optimale

- Définition: On appelle prédicteur optimal de la sortie $y(t)$, sur la base des informations disponibles à l'instant $(t-1)$, le nombre réel, noté

$$\hat{y}(t|\theta, t-1) = f(\theta, y(t-1), y(t-2), \dots, u(t-1), u(t-2), \dots)$$

qui minimise la variance

$$E((y(t) - x)^2) = g(x)$$

Soit

$$\hat{y}(t|\theta, t-1) = \arg \left[\min_x (g(x)) \right]$$

2.4 Forme de régression et Prédiction optimale

- Définition: La forme de régression exprime la sortie $y(t)$ en fonction d'un vecteur d'observations (régresseur) et du vecteur de paramètres. Ce dernier est estimé par un algorithme d'identification via cette forme de régression

$$y(t) = f(\theta, \Phi^T, \dots)$$

Exemple: modèle ARX sans perturbations

$$y(t) = \Phi^T \theta$$

Un algo d'identification

$$\hat{\theta}_N = \left[\sum_{i=1}^N \Phi(t) \Phi^T(t) \right]^{-1} \sum_{t=1}^N \Phi(t) y(t)$$

2.4.1 Modèle ARX $A(q^{-1})y(t) = B(q^{-1})u(t) + e(t)$

On démontre que le prédicateur optimal pour ce modèle est (voir annexe1):

$$\begin{aligned} \hat{y}(t | \theta, t-1) &= \Phi^T(t) \theta \\ &= (1 - A(q^{-1}))y(t) + B(q^{-1})u(t) \end{aligned}$$

Forme de régression:

$$y(t) = \Phi^T(t) \theta + e(t)$$

$$\begin{aligned} \Phi^T(t) &= [-y(t-1) \quad \dots \quad -y(t-na) \quad u(t-nk) \quad \dots \quad u(t-nb-nk+1)] \\ \theta &= [a_1 \quad \dots \quad a_{na} \quad b_1 \quad \dots \quad b_{nb}]^T \end{aligned}$$

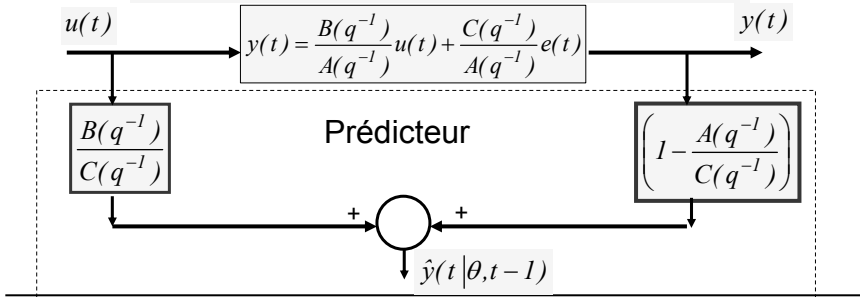
Le vecteur de paramètre apparaît de manière linéaire.

2.4.2 Modèle ARMAX (1)

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

On démontre que le prédicteur optimal pour ce modèle est (voir annexe2):

$$\hat{y}(t|\theta, t-1) = \left(1 - \frac{A(q^{-1})}{C(q^{-1})}\right)y(t) + \frac{B(q^{-1})}{C(q^{-1})}u(t)$$



2.4.2 Modèle ARMAX(2)

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

Calcul pratique du prédicteur optimal:

$$C(q^{-1})\hat{y}(t|\theta, t-1) = (C(q^{-1}) - A(q^{-1}))y(t) + B(q^{-1})u(t)$$

$$\hat{y}(t|\theta, t-1) = (1 - C(q^{-1}))\hat{y}(t|\theta, t-1) + (C(q^{-1}) - A(q^{-1}))y(t) + B(q^{-1})u(t)$$

$$\begin{aligned} \hat{y}(t|\theta, t-1) = & c_1\hat{y}(t-1|\theta, t-2) + \dots + c_{nc}\hat{y}(t-nc|\theta, t-nc-1) + \\ & c_1y(t-1) + \dots + c_{nc}y(t-nc) - a_1y(t-1) - \dots - a_{na}y(t-na) + \\ & b_1u(t-nk) + \dots + b_{nb}u(t-nb-nk+1) \end{aligned}$$

2.4.2 Modèle ARMAX(3)

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

Effets des conditions initiales?

$$\hat{y}(-1|\theta, -2), \hat{y}(-1|\theta, -3), \dots, \hat{y}(-nc|\theta, -nc-1),$$

Comme $\frac{1}{C(q^{-1})}$ est asymptotiquement stable l'effet des conditions initial s'annule.

Deux prédictions avec deux conditions initiales différentes

$$C(q^{-1})(\hat{y}_1(t|\theta, t-1) - \hat{y}_2(t|\theta, t-1)) = 0$$

$$(\hat{y}_1(t|\theta, t-1) - \hat{y}_2(t|\theta, t-1)) \xrightarrow{\text{expon}} 0$$

2.4.2 Modèle ARMAX(4)

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

Le prédicateur s'écrit

$$\hat{y}(t|\theta, t-1) = (1 - C(q^{-1}))\hat{y}(t|\theta, t-1) + (C(q^{-1}) - 1 + 1 - A(q^{-1}))y(t) + B(q^{-1})u(t)$$

Soit l'erreur de prédiction

$$\delta(t, \theta) = y(t) - \hat{y}(t|\theta, t-1)$$

On obtient

avec

$$\hat{y}(t|\theta, t-1) = \Phi^T(t, \theta) \cdot \theta$$

$$\theta = [a_1 \dots a_{na} \ b_1 \dots b_{nb} \ c_1 \dots c_{nc}]^T$$

$$\Phi^T(t, \theta) = [-y(t-1) \dots -y(t-na) \ u(t-nk) \dots u(t-nk-nb+1) \dots \delta(t-1, \theta) \dots \delta(t-nc, \theta)]^T$$

2.4.2 Modèle ARMAX(5)

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

D'où la forme de régression

$$y(t) = \Phi^T(t, \theta) \cdot \theta + \delta(t, \theta)$$

$$\theta = [a_1 \dots a_{na} \ b_1 \dots b_{nb} \ c_1 \dots c_{nc}]^T$$

$$\Phi^T(t, \theta) = [-y(t-1) \dots -y(t-na) \ u(t-nk) \dots u(t-nk-nb+1) \ \delta(t-1, \theta) \dots \delta(t-nc, \theta)]^T$$

Remarque:

$$\delta(t, \theta) = y(t) - \hat{y}(t | \theta, t-1) = e(t)$$

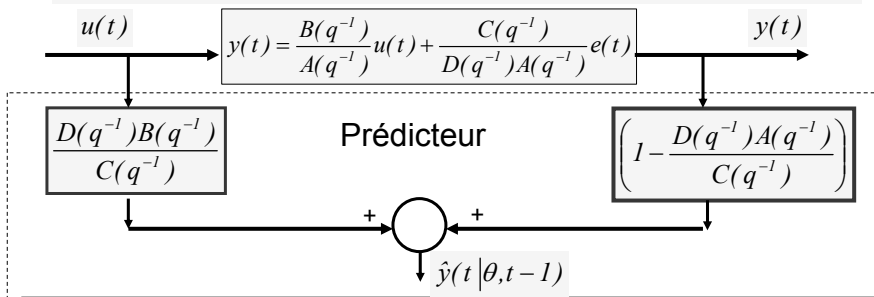
$$y(t) = \Phi^T(t, \theta) \cdot \theta + e(t) \quad \text{pseudo_linéaire}$$

2.4.2 Modèle ARMAX (6)

$$A(q^{-1})y(t) = B(q^{-1})u(t) + \frac{C(q^{-1})}{D(q^{-1})}e(t)$$

Montrer que le prédicteur optimal pour ce modèle est (voir annexe3):

$$\hat{y}(t | \theta, t-1) = \left(1 - \frac{D(q^{-1})A(q^{-1})}{C(q^{-1})} \right) y(t) + \frac{D(q^{-1})B(q^{-1})}{C(q^{-1})} u(t)$$



2.4.3 Modèle OE (1)

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

On démontre que le prédicteur optimal pour ce modèle est (voir annexe4):

$$\hat{y}(t | \theta, t-1) = \frac{B(q^{-1})}{A(q^{-1})}u(t)$$

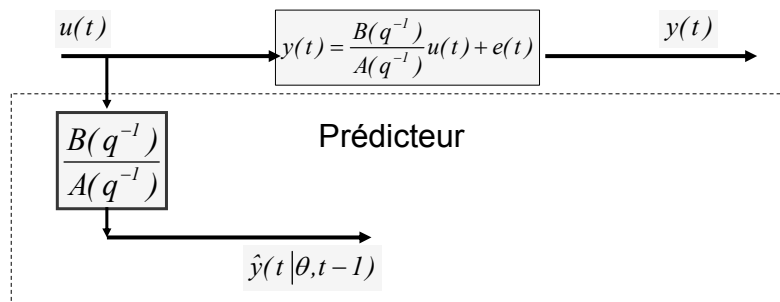
Calcul pratique

$$\hat{y}(t | \theta, t-1) = (1 - A(q^{-1}))\hat{y}(t | \theta, t-1) + B(q^{-1})u(t)$$

$$\hat{y}(t | \theta, t-1) = a_1\hat{y}(t-1 | \theta, t-2) + \dots + a_{na}\hat{y}(t-na | \theta, t-na-1) + b_1u(t-nk) + \dots + b_{nb}u(t-nk-nb+1)$$

2.4.3 Modèle OE (2)

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



2.4.3 Modèle OE (3)

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

Puisque

$$\hat{y}(t | \theta, t-1) = \frac{B(q^{-1})}{A(q^{-1})} u(t)$$

D'où

$$y(t) = \hat{y}(t | \theta, t-1) + e(t)$$

La forme de régression pseudo-linéaire s'écrit

$$y(t) = \Phi^T(t, \theta) \theta + e(t)$$

$$\theta = [a_1 \dots a_{na} \quad b_1 \dots b_{nb}]^T$$

$$\Phi^T(t, \theta) = [-\hat{y}(t-1 | \theta, t-2) \dots -\hat{y}(t-na | \theta, t-na-1) \\ u(t-nk) \dots u(t-nk-nb+1)]^T$$

2.4.4 Modèle Général (1)

$$A(q^{-1})y(t) = \frac{B(q^{-1})}{F(q^{-1})} u(t) + \frac{C(q^{-1})}{D(q^{-1})} e(t)$$

On démontre que le prédicteur optimal pour ce modèle est (voir annexe5):

$$\hat{y}(t | \theta, t-1) = \frac{D(q^{-1})B(q^{-1})}{C(q^{-1})F(q^{-1})} u(t) + \left(1 - \frac{D(q^{-1})A(q^{-1})}{C(q^{-1})} \right) y(t)$$

Sachant que

$$A(q^{-1}) = 1 + \sum_{i=1}^{na} a_i q^{-i}; F(q^{-1}) = 1 + \sum_{i=1}^{nf} f_i q^{-i}; C(q^{-1}) = 1 + \sum_{i=1}^{nc} c_i q^{-i}; \\ B(q^{-1}) = \sum_{i=1}^{nb} b_i q^{-nk-i+1}; D(q^{-1}) = 1 + \sum_{i=1}^{nd} d_i q^{-i}$$

Le vecteur de paramètre est

$$\theta = [a_1 \quad \dots \quad a_{na} \quad b_1 \quad \dots \quad b_{nb} \quad c_1 \quad \dots \quad c_{nc} \quad d_1 \quad \dots \quad d_{nd} \quad f_1 \quad \dots \quad f_{nf}]$$

2.4.4 Modèle Général (2)

$$A(q^{-1})y(t) = \frac{B(q^{-1})}{F(q^{-1})}u(t) + \frac{C(q^{-1})}{D(q^{-1})}e(t)$$

En utilisant les variables intermédiaires

$$w(t, \theta) = \frac{B(q^{-1})}{F(q^{-1})}u(t)$$

$$v(t, \theta) = A(q^{-1})y(t) - w(t, \theta)$$

On démontre qu'on obtient la forme de régression suivante (annexe 6):

2.4.4 Modèle Général (3)

$$A(q^{-1})y(t) = \frac{B(q^{-1})}{F(q^{-1})}u(t) + \frac{C(q^{-1})}{D(q^{-1})}e(t)$$

$$y(t) = \Phi^T(t, \theta) \cdot \theta + \delta(t, \theta) \quad \text{forme pseudo-linéaire}$$

Avec

$$\theta = [a_1 \quad \dots \quad a_{na} \quad b_1 \quad \dots \quad b_{nb} \quad c_1 \quad \dots \quad c_{nc} \quad d_1 \quad \dots \quad d_{nd} \quad f_1 \quad \dots \quad f_{nf}]$$

$$\begin{aligned} \Phi^T = & [-y(t-1) \quad \dots \quad -y(t-na) \quad u(t-nk) \quad \dots \quad u(t-nk-nb+1) \\ & \delta(t-1, \theta) \quad \dots \quad \delta(t-nc, \theta) \quad -v(t-1, \theta) \quad \dots \quad -v(t-nd, \theta) \\ & -w(t-1, \theta) \quad \dots \quad -w(t-nf, \theta)] \end{aligned}$$

L'erreur de prédiction

$$\delta(t, \theta) = y(t) - \hat{y}(t | \theta, t-1) = e(t)$$

2.4.5 Remarques générales(1)

- Tous les cas précédents, l'erreur de prédiction est donnée par:

$$\delta(t, \theta) = y(t) - \hat{y}(t | \theta, t-1) = e(t)$$

Cette propriété sera utilisée ultérieurement dans les procédures de validation du modèle estimé (test de blancheur, test d'indépendance).

2.4.5 Remarques générales(2)

- Dans les formes de régression pseudo-linéaires, les algorithmes d'identification suggèrent de remplacer à l'instant t, le vecteur θ par $\hat{\theta}(t-1)$

Sans cette substitution, il est impossible de calculer à l'instant t le vecteur d'observation $\Phi^T(t, \theta)$

2.5 Boîtes à outils pratiques

Il existe de nombreux outils informatiques qui permettent l'identification des systèmes:

- Matlab, Labview, Scilab...
- Objectif:
 - connaître et savoir utiliser qq uns de ces outils.
 - Profiter de l'outil informatique pour automatiser et faciliter la procédure d'identification.

2.5.1 Déroulement général d'une procédure d'identification

- Les étapes typiques d'une procédure d'identification sous Matlab:
 - Préparation des données: Importation / visualisation des données/pré-traitement (valeur moy, filtrage, dérivées...)
 - Estimation et validation du modèle: essai de différentes structures de modèles, avec différents ordres et délais.
 - Post-traitement et transformation du modèle: discret/linéaire; réduction d'ordre ...

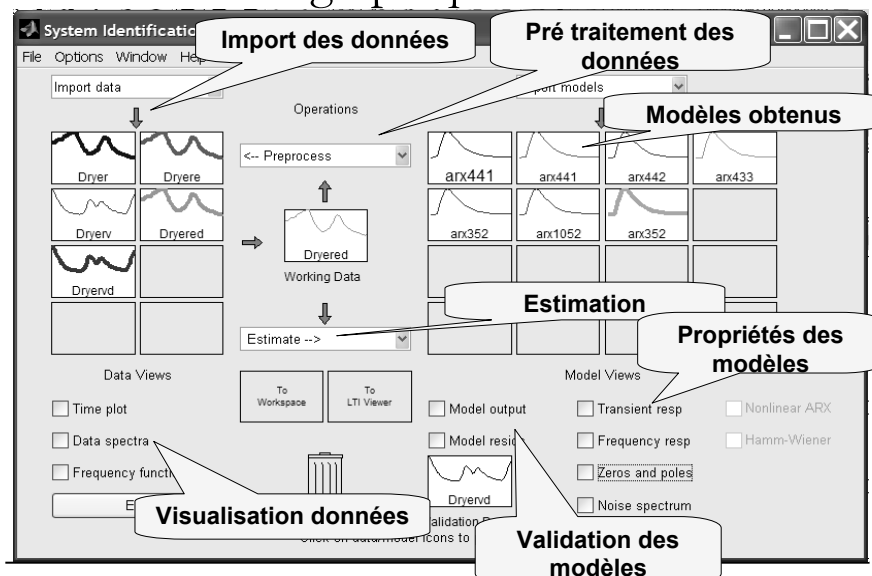
2.5.2 Toolbox system identification- Matlab™

Matlab permet l'identification des systèmes à l'aide:

- D'une boîte à outils graphiques.
- Ou d'un ensemble de commandes /fonctions.

Ces outils couvrent l'identification paramétrique/ non paramétrique des systèmes linéaires/ non linéaires, discrets/ continus, dans les domaines temporel /fréquentiel, et supportent plusieurs structures de modèles (équations aux différences, équation d'état...)

2.5.3 Interface graphique d'utilisateur GUI



Identification des systèmes dynamiques

Chap 3 Identification paramétrique récursive en l'absence de perturbations

3.1 Récursivité

L'identification paramétrique par un algorithme récursif permet d'estimer en ligne le vecteur des paramètres = à chaque instant d'échantillonnage le vecteur de paramètres est estimé en fonction du vecteur précédemment trouvé.

L'identification récursive permet donc:

- ❑ De mieux estimer des systèmes variants dans le temps.
- ❑ De ne pas impliquer des calculs avec un grand nombre de données.

3.1 Position du problème et Objectif(1)

On s'intéresse dans un premier temps à l'identification d'un système de type ARX sans perturbation.

$$A(q^{-1})y(t) = B(q^{-1})u(t) \quad \text{Avec} \quad A(q^{-1}) = 1 + a_1q^{-1} + \dots + a_nq^{-n}$$
$$B(q^{-1}) = b_1q^{-1} + \dots + a_nq^{-n}$$
$$n \geq \max(n_a, n_b) \text{ connu}$$

Si $n > \max(n_a, n_b)$ alors $a_i = 0$ et $b_j = 0$ pour $i = n_a + 1, \dots, n$
et $j = n_b + 1, \dots, n$

Si le retard $n_k > 1$ alors $b_j = 0$ pour $j = 1, \dots, n_k - 1$

La structure du modèle est donc connue, au prix d'une surparamétrisation du modèle.

3.1 Position du problème et Objectif(2)

L'objectif est d'estimer en ligne le vecteur de paramètre inconnu

$$\theta = [a_1 \quad \dots \quad a_n \quad b_1 \quad \dots \quad b_n]^T$$

en utilisant des algorithmes récursifs d'identification.

3.2 Algorithme de gradient(1)

Forme de régression (chap 2) $y(t) = \Phi^T(t)\theta$

Le prédicteur optimal (chap 2)

$$\begin{aligned}\hat{y}(t|\theta, t-1) &= \Phi^T(t)\theta \\ &= (I - A(q^{-1}))y(t) + B(q^{-1})u(t)\end{aligned}$$

Comme θ est inconnu, on suggère comme prédicteur

$$\hat{y}(t|\theta, t-1) = \Phi^T(t)\hat{\theta}(t-1)$$

avec par déf $\hat{\theta}(t)$ l'estimé de θ en utilisant les données $u(i)$ et $y(i)$ pour $i = 0, \dots, t$

3.2 Algorithme de gradient(2)

On définit l'erreur de prédiction a priori par

$$\delta(t) = y(t) - \Phi^T(t)\hat{\theta}(t-1)$$

On définit l'erreur de prédiction a posteriori par

$$\delta_p(t) = y(t) - \Phi^T(t)\hat{\theta}(t)$$

La méthode du gradient considère que le meilleur estimateur pour θ est le vecteur x , qui minimise le critère:

$$J(x) = \|x - \hat{\theta}(t-1)\|^2 + \lambda(y(t) - \Phi^T(t)x)^2$$

Ce vecteur sera noté $\hat{\theta}(t)$

3.2 Algorithme de gradient(3)

Dans le critère:

$$J(x) = \|x - \hat{\theta}(t-1)\|^2 + \lambda(y(t) - \Phi^T(t)x)^2$$

λ Module le compromis entre:

- Meilleure prédiction a posteriori
- Convergence rapide de $\hat{\theta}(t)$, soit la proximité de $\hat{\theta}(t-1)$

On démontre(voir annexe7) que le vecteur recherché se calcule de la manière suivante:

3.2 Algorithme de gradient(2)

Algorithme du gradient

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{\lambda \Phi(t) \delta(t)}{1 + \lambda \Phi^T(t) \Phi(t)}$$

$$\delta(t) = y(t) - \Phi^T(t) \hat{\theta}(t-1); \text{ avec } \hat{\theta}(0) = 0$$

Autre forme

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{\Phi(t) \delta(t)}{\alpha + \Phi^T(t) \Phi(t)}$$

$$\delta(t) = y(t) - \Phi^T(t) \hat{\theta}(t-1); \alpha = 1/\lambda; \text{ avec } \hat{\theta}(0)$$

TP: programmation de cet algorithme

3.2 Algorithme de gradient(3)

Propriétés de l'algorithme du gradient

$\|\hat{\theta}(t) - \theta\|$ est non croissante

$$\lim_{t \rightarrow \infty} \frac{\delta(t)}{\sqrt{\alpha + \Phi^T(t)\Phi(t)}} = 0$$

si $\|\Phi(t)\|$ est bornée alors $\lim_{t \rightarrow \infty} \delta(t) = 0$

$$\lim_{t \rightarrow \infty} \delta_p(t) = \lim_{t \rightarrow \infty} (y(t) - \Phi^T(t)\hat{\theta}(t)) = 0$$

Le modèle induit par $\hat{\theta}(t)$ est bien représentatif du système réel.

Mais ce ci n'implique pas forcément que $\hat{\theta}(t) \xrightarrow{t \rightarrow \infty} \theta$

3.2 Algorithme de gradient(4)

En effet, l'algorithme résoud récursivement

$$y(t) = \Phi^T(t)X \text{ soit } \Phi^T(t)\theta = \Phi^T(t)X$$

l'équation $\Phi^T(t)(\theta - X) = 0$ n'implique pas forcément

$X = \theta$, mais seulement que $(\theta - X)$ appartient au sous espace orthogonal au sous espace engendré par le vecteur $\Phi(t)$.

Donc si $\Phi(t)$ balaye tout l'espace \mathbb{R}^{2n} alors le vecteur $(\theta - X) = 0$ soit $X = \theta$

3.2 Algorithme de gradient(5)

Pour assurer la convergence des estimés des paramètres vers leur vraies valeurs:

Il faut que $\Phi(t)$ balaye tout l'espace \mathbb{R}^{2n}

On dit que $\Phi(t)$ doit avoir la propriété d'excitation persistante EP.

Déf : $\Phi(t)$ possède la propriété d'excitation persistante EP si

$$\exists N \text{ et } \alpha > 0 \text{ tels que } \sum_{i=1}^N \Phi(t+i) \Phi^T(t+i) \geq \alpha I$$

$$\text{ou autrement } \forall W \in \mathbb{R}^{2n} \quad \sum_{i=1}^N (\Phi^T(t+i)W)^2 \geq \alpha \|W\|^2$$

Sous cette condition $\hat{\theta}(t) \rightarrow \theta$ quand $t \rightarrow \infty$

3.2 Algorithme de gradient(5)

Remarquons que

$$\Phi^T(t) = H(q^{-1})u(t)$$

Avec

$$H(q^{-1}) = \begin{bmatrix} -q^{-1} \frac{B(q^{-1})}{A(q^{-1})} & \dots & -q^{-n+1} \frac{B(q^{-1})}{A(q^{-1})} & q^{-1} & \dots & q^{-n} \end{bmatrix}$$

C'est la nature de $u(t)$ qui procure à $\Phi(t)$ la propriété EP (voir plus loin pour le choix de $u(t)$).

3.2 Algorithme des MC récursif(1)

La méthode des MC récursive considère que le meilleur estimateur pour θ est le vecteur x , qui minimise le critère:

$$J(x) = \sum_{i=1}^t (y(i) - \Phi^T(i)x)^2$$

La solution non récursive à ce problème est:

$$\hat{\theta}(t) = \left(\sum_{i=1}^t \Phi(i)\Phi^T(i) \right)^{-1} \sum_{i=1}^t \Phi(i)y(i)$$

3.2 Algorithme des MC récursif(2)

On démontre que la solution récursive de ce problème est donnée par (voir annexe 8):

$$\hat{\theta}(t) = \hat{\theta}(t-1) + P(t)\Phi(t)\delta(t)$$

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

$$\delta(t) = y(t) - \Phi^T(t)\hat{\theta}(t-1)$$

Initialisation $\hat{\theta}(0) = \theta_0$ si on a des infos sur le système
ou $\hat{\theta}(0) = 0$

$$P(0) = \alpha I_d \quad \text{avec } \alpha \gg 1 \text{ (1000, 10000...)}$$

3.2 Algorithme des MC récursif(3)

Autre forme des MC récursives

$$\begin{aligned}\hat{\theta}(t) &= \hat{\theta}(t-1) + \frac{P(t-1)\Phi(t)\delta(t)}{1 + \Phi^T(t)P(t-1)\Phi(t)} \\ P(t) &= P(t-1) - \frac{P(t-1)\Phi(t)\Phi^T(t)P(t-1)}{1 + \Phi^T(t)P(t-1)\Phi(t)} \\ \delta(t) &= y(t) - \Phi^T(t)\hat{\theta}(t-1)\end{aligned}$$

TP programmation de cet algorithme

3.2 Algorithme des MC récursif(3)

Propriétés des MC récursives

$$\begin{aligned}\|\hat{\theta}(t) - \theta\| &\leq K \|\hat{\theta}(0) - \theta\| \\ \frac{\delta(t)}{\sqrt{1 + \Phi^T(t)\Phi(t)}} &\text{ est de carré intégrable}\end{aligned}$$

$$\|\hat{\theta}(t) - \hat{\theta}(t-1)\| \text{ est de carré intégrable}$$

Le modèle induit par $\hat{\theta}(t)$ est bien représentatif du système réel.

Mais ce ci n'implique pas forcément que $\hat{\theta}(t) \xrightarrow{t \rightarrow \infty} \theta$.

$\hat{\theta}(t) \xrightarrow{t \rightarrow \infty} \theta$, nécessité d'avoir la propriété d'EP pour $\Phi(t)$.

Identification des systèmes dynamiques

Chap 4 Identification paramétrique récursive en présence de perturbations

4.1 Position du problème et Objectif

On s'intéresse à l'identification d'un système en présence de perturbations.

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

$$A(q^{-1}) = 1 + a_1 q^{-1} + \dots + a_n q^{-n}$$

$$B(q^{-1}) = b_1 q^{-1} + \dots + a_n q^{-n}$$

$$n \geq \max(n_a, n_b) \text{ connu}$$

Le terme perturbateur est $\varepsilon(t)$

Cette équation est assez générale, en effet si :

- $\varepsilon(t) = e(t)$ (bruit blanc) alors Modèle ARX
- $\varepsilon(t) = C(q^{-1})e(t)$ alors Modèle ARMAX
- $\varepsilon(t) = A(q^{-1})e(t)$ alors Modèle OE
- $\varepsilon(t) = \frac{C(q^{-1})}{D(q^{-1})}e(t)$ alors Modèle Box_Jenkins

4.2 Modèle ARX perturbé

$$A(q^{-1})y(t) = B(q^{-1})u(t) + e(t) \quad e(t) \text{ (bruit blanc)}$$

Forme de régression linéaire: $y(t) = \Phi(t)\theta + e(t)$

Algo MC récursif donne

$$\hat{\theta}(t) = \hat{\theta}(t-1) + P(t)\Phi(t)\delta(t)$$

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

$$\delta(t) = y(t) - \Phi^T(t)\hat{\theta}(t-1)$$

sous la condition d'EP $\lim_{t \rightarrow \infty} \hat{\theta}(t) = \theta$

4.2 Modèle ARMAX(1)

$$A(q^{-1})y(t) = B(q^{-1})u(t) + C(q^{-1})e(t) \quad e(t) \text{ (bruit blanc)}$$

Forme de régression pseudo-linéaire (voir chap2):

$$y(t) = \Phi(t, \theta)\theta + e(t)$$

Le prédicteur optimal au sens des MC (chap2)

$$\hat{y}(t | \theta, t-1) = (I - C(q^{-1}))\hat{y}(t | \theta, t-1) + (C(q^{-1}) - A(q^{-1}))y(t) + B(q^{-1})u(t)$$

Ces deux équations n'ont aucun intérêt pratique puisque θ n'est pas connu.

4.2 Modèle ARMAX(2)

Les MC récursives suggèrent de remplacer dans la forme de régression pseudo-linéaire et dans le prédicteur optimal

$$\theta \text{ par } \hat{\theta}(t-1)$$

Soient

$$y(t) = \Phi^T(t, \hat{\theta}(t-1))\theta + e(t)$$

$$\hat{y}(t | \theta, t-1) \text{ par } \hat{y}(t | \hat{\theta}(t-1), t-1)$$

Les MC récursives prennent alors la forme:

4.2 Modèle ARMAX(3)

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{P(t-1)\Phi(t, \hat{\theta}(t-1))\delta(t, \hat{\theta}(t-1))}{1 + \Phi^T(t, \hat{\theta}(t-1))P(t-1)\Phi(t, \hat{\theta}(t-1))}$$

$$P(t) = P(t-1) - \frac{P(t-1)\Phi(t, \hat{\theta}(t-1))\Phi^T(t, \hat{\theta}(t-1))P(t-1)}{1 + \Phi^T(t, \hat{\theta}(t-1))P(t-1)\Phi(t, \hat{\theta}(t-1))}$$

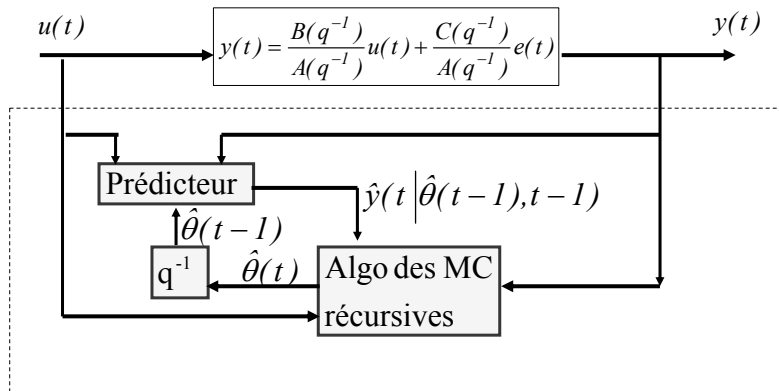
$$\delta(t, \hat{\theta}(t-1)) = y(t) - \Phi^T(t, \hat{\theta}(t-1))\hat{\theta}(t-1)$$

$$\Phi^T(t, \hat{\theta}(t-1)) =$$

$$[-y(t-1) \dots -y(t-n) \ u(t-1) \dots u(t-n)]$$

$$\delta(t-1, \hat{\theta}(t-2)) \dots \delta(t-n, \hat{\theta}(t-n-1))]^T$$

4.2 Modèle ARMAX(4)



TP programmation de cet algorithme

4.3 Modèle OE(1)

$$A(q^{-1})y(t) = B(q^{-1})u(t) + \frac{1}{A(q^{-1})}e(t) \quad e(t) \text{ (bruit blanc)}$$

Forme de régression pseudo-linéaire (voir chap2):

$$y(t) = \Phi(t, \theta)\theta + e(t)$$

Le prédicteur optimal au sens des MC (chap2)

$$\hat{y}(t | \theta, t-1) = (1 - A(q^{-1}))\hat{y}(t | \theta, t-1) + B(q^{-1})u(t)$$

Ces deux équations n'ont aucun intérêts pratiques
puisque θ n'est pas connu.

4.3 Modèle OE(2)

Les MC récursives suggèrent de remplacer dans la forme de régression pseudo-linéaire et dans le prédicteur optimal

$$\theta \text{ par } \hat{\theta}(t-1)$$

Soient

$$y(t) = \Phi^T(t, \hat{\theta}(t-1))\theta + e(t)$$

$$\hat{y}(t | \theta, t-1) \text{ par } \hat{y}(t | \hat{\theta}(t-1), t-1)$$

Les MC récursives prennent alors la forme:

4.3 Modèle OE(3)

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{P(t-1)\Phi(t, \hat{\theta}(t-1))\delta(t, \hat{\theta}(t-1))}{1 + \Phi^T(t, \hat{\theta}(t-1))P(t-1)\Phi(t, \hat{\theta}(t-1))}$$

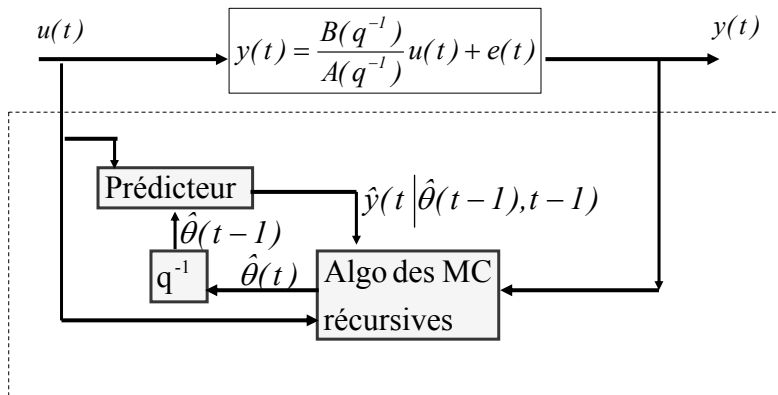
$$P(t) = P(t-1) - \frac{P(t-1)\Phi(t)\Phi^T(t, \hat{\theta}(t-1))P(t-1)}{1 + \Phi^T(t, \hat{\theta}(t-1))P(t-1)\Phi(t, \hat{\theta}(t-1))}$$

$$\delta(t, \hat{\theta}(t-1)) = y(t) - \Phi^T(t, \hat{\theta}(t-1))\hat{\theta}(t-1)$$

$$\Phi^T(t, \hat{\theta}(t-1)) =$$

$$[-\hat{y}(t-1 | \hat{\theta}(t-2), t-2) \dots - \hat{y}(t-n | \hat{\theta}(t-n-1), t-n-1) \\ u(t-1) \dots u(t-n)]^T$$

4.4 Modèle OE(4)



TP programmation de cet algorithme