



République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche
Scientifique

Université Oran 2 Mohamed Ben Ahmed

Institut de Maintenance et de Sécurité Industrielle

Polycopié de cours

Identification des Processus

Dr KACIMI ABDERRAHMANE, MCB

1^{ère} année Master Génie Industriel

Avant-propos

Le cours intitulé « Identification des processus » est enseigné en master 1 de la spécialité Génie Industrielle au sein du département de maintenance en instrumentation au niveau de l'IMSI à l'université d'Oran 2 Mohamed Ben Ahmed. Il est articulé sur le canevas du module.

Ce cours est destiné aux étudiants de niveau master 1 et ingénieurs en spécialités automatique, électrotechnique et génie industrielle.

Le polycopié est devisé en cinq chapitres :

- **Le premier chapitre** contient des généralités sur l'identification des processus que l'étudiant doit connaître, tant en aspect théorique que pratique, avec son lien avec la commande des systèmes dynamiques, notamment les procédés industriels.

- **Le deuxième chapitre** traite une partie de l'identification consacrée à l'identification des procédés industriels qui se base que peu des modèles mathématiques des systèmes dynamiques au profit des connaissances et informations pratique, il s'agit de l'identification non paramétrique.

- **Le troisième chapitre** est consacré à l'identification paramétrique qui contrairement à la technique du chapitre précédent se base essentiellement des modèles mathématiques des systèmes dynamiques, combinés aux mesures obtenus par des expériences sur les systèmes physiques à identifiés. Cette technique représente actuellement la partie essentielle de l'identification des systèmes dynamiques, des algorithmes variés et aussi efficaces seront présentés dans ce chapitre.

- **Le quatrième chapitre** est dédié à la version récursive des algorithmes étudiés dans chapitre précédent, notamment l'algorithme des moindres carrés récursifs.

Mis à part le premier chapitre, chaque chapitre contient des exercices corrigés concernant son contenu, ce qui correspond aux travaux dirigés dispensé dans ce module.

Ces exercices sont élaborés de façon à aider les étudiants à assimiler les différents concepts étudiés dans ce cours.

Objectifs de l'enseignement

Ce cours vise les objectifs suivants :

- Initiation aux concepts d'identification des processus et leur lien avec commande des systèmes dynamiques.
- L'étudiant doit comprendre le rôle des techniques d'identification en ses différentes méthodes continu et discret, dans l'application industrielle.
- L'étudiant doit maîtriser ces techniques par leurs aspects théoriques et expérimentaux.

Prérequis :

- Automatique linéaire continue et échantillonnée.
- Théorie des signaux et système, continu et discret, algèbre linéaire.
- Notions d'optimisation.

Niveau : 1^{ère} année Master.

Spécialité : Génie Industriel (GI).

Enseignement : en Français.

Mots clefs : Identification, systèmes dynamiques, paramétrique, non paramétrique, protocole expérimental, signal riche en fréquence, moindres carrés, erreur de sortie.

Table des matières

Table des matières

Table des matières.....	2
Liste des figures	4
Liste des tableaux	5
Liste d'abréviation et symboles.....	6
I. Généralités.....	8
I.1 Introduction.....	9
I.2 Exposé du problème.....	9
I.3 Algorithme général d'identification	11
II. Identification non paramétrique.....	13
II.1 Introduction.....	14
II.2 Quelle entrée ?	14
II.2.1 Analyse harmonique.....	14
II.2.2 Réponse impulsionnelle.....	14
II.2.3 Réponse indicielle.....	14
II.3 Etude des réponses indicielles.....	14
II.3.1 Méthode de Broida.....	14
II.3.2 Méthode de Strecj.....	15
II.3.3 Méthode de Strecj avec intégrateur.....	16
Exercice 2.1.....	17
II.4 Méthode de corrélation	18
II.4.1 Rappels sur les signaux aléatoires	18
II.4.2 Obtention de la réponse impulsionnelle	19
II.4.3 Modalités d'application.....	20
II.4.4 Exemple.....	21
III. Identification paramétrique.....	23
III.1 Choix du signal d'excitation.....	24
III.2 Les séquences binaires pseudo aléatoires (SBPA)	24
III.2.1 Choix des paramètres d'une SBPA.....	26
Exercice 3.1.....	29
III.3 Identification basée sur l'erreur de sortie.....	29
III.4 Identification basée sur l'erreur de prédiction.....	30
III.4.1 Méthode des moindres carrés simples.....	30
Exercice 3.2.....	31
Exercice 3.3.....	32
III.4.2 Calcul du biais de l'estimateur.....	34
III.4.3 Méthode de la Matrice Instrumentale.....	36
III.5 Identification en boucle fermée.....	36
Exercice 3.4.....	38
III.6 Boîte d'outil d'identification sous matlab « identool ».....	38
III.7 Exemple d'identification par la toolbox identool	39
III.8 Cas pratique	46

Table des matières

IV.	Estimations récursives.....	47
	IV.1 Moindres carrés récursifs.....	48
	IV.2 Variable instrumentale réursive.....	51
	Exercice 4.1	51
	IV.3 Cas pratique.....	53
	Bibliographie.....	55

Table des matières

Liste des figures :

Figure I.1 : Algorithme général d'identification des systèmes.....	12
Figure II.1 : Méthode de Broida : Réponse indicielle.....	15
Figure II.2 : Méthode de Strejc : Réponse indicielle.....	16
Figure II.3 : Méthode de Strejc (système avec intégrateur).....	16
Figure II.4 : Réponse du système à un créneau unité de durée $t = 0,5s$	17
Figure II.5 : Réponse du système réel et par la réponse impulsionnelle estimée.....	22
Figure III.1 : diagramme temporel d'une séquence binaire pseudo aléatoire.....	24
Figure III.2 : Fonction d'autocorrélation d'une SBPA.....	25
Figure III.3 : Spectre d'une SBPA.....	26
Figure III.4 : Paramètres d'une SBPA.....	27
Figure III.5 : Programme simulink pour la génération d'une SBPA avec 5 bascules.....	28
Figure III.6 : Principe d'identification fondé sur l'erreur de sortie.....	29
Figure III.7 : Principe d'identification fondée sur l'erreur de prédiction.....	30
Figure III.8 : Modèle du système étudié.....	30
Figure III.9 : Système du premier ordre bruité.....	35
Figure III.10 : Fonction d'auto-corrélation des résidus d'estimation.....	35
Figure III.11 : Identification des systèmes en boucle fermée.....	37
Figure III.12 : La GUI de la toolbox identool de matlab.....	39
Figure III.13 : L'importation des données expérimentales temporels.....	39
Figure III.14 : Fenêtre des données d'entrée et de sortie avec période d'échantillonnage.....	40
Figure III.15 : La fenêtre import Data de la GUI de la toolbox identool.....	40
Figure III.16 : Fenêtre du filtrage des données comportant le diagramme de Bode.....	41
Figure III.17 : Fenêtre des données filtrées.....	41
Figure III.18 : Fenêtre du choix de la méthode d'estimation.....	42
Figure III.19 : Fenêtre de l'estimation des paramètres de la GUI identool.....	42
Figure III.20 : Fenêtre du résultat d'estimation obtenu.....	43
Figure III.21 : Fenêtre de la validation du modèle estimé (sortie mesurée et sortie prédite)....	43

Table des matières

Figure III.22 : Visualisation des résultats de l'identification sur command window.....44

Figure IV.1 : programme de simulation de l'identification par les moindres carrés récursives
appliquée au moteur à courant continu.....50

Figure IV.2 : vitesse du moteur à CC mesurée et prédite par le modèle identifié.....50

Liste des tableaux :

Tableau III.1: Tableau des bits à utiliser pour obtenir une séquence de longueur maximale.....27

Tableau III.2: Modèles paramétriques implantés sous Matlab.....45

Tableau IV.1: Données mesurées.....51

Liste d'abréviation et symboles

argmin : Argument à minimizer.

ARMA : Autoregressive moving average.

ARMAX : Autoregressive moving average avec entrée exogène.

ARX : Autoregressive avec entrée exogène.

BJ : Box–Jenkins.

$Cov\{\}$: Covariance.

C : L'ensemble des nombres complexes.

$e(t_k)$: Signal du bruit blanc.

$E\{\}$: Espérance mathématique.

FIR : Réponse impulsionnelle finie.

$\phi(t_k)$: Vecteur de régression à instant t_k .

Φ_N : Matrice de régression.

$\eta(t_k)$: Bruits de mesures filtrés à instant t_k .

I_n : Matrice d'identité ($n \times n$).

IV : Variable Instrumentale.

LS : Moindres carrés.

LTI : Linéaire à temps invariant

m : L'ordre du numérateur de la fonction de transfert.

MSE : La moyenne des carrés de l'erreur.

n : L'ordre du modèle.

N : Nombre d'échantillons de donnés.

NSR : Rapport du signal au bruit.

N : L'ensemble des nombres naturel.

OE : L'erreur de sortie.

P_θ : Matrice de covariance.

PDF : La fonction de densité de probabilité.

PEM : La méthode de l'erreur de prédiction.

q : L'opérateur d'avance.

Table des matières

q^{-1} : L'opérateur de retard.

\Re : L'ensemble des nombres réels.

$R_u(\tau)$: La fonction d'auto covariance du signal u .

$R_{yu}(\tau)$: La fonction de covariance des signaux y et u .

s : variable de Laplace.

σ Singular value

σ_e^2 : Variance du signal stochastique e .

t_k : pas d'échantillonnage.

T_s : période d'échantillonnage.

θ : Vecteur de paramètres.

$\hat{\theta}$: Vecteur de paramètres estimés.

$Var\{\}$: Variance.

$\xi(t_k)$: bruits stochastique coloré à instant t_k .

Chapitre 1: Généralités

1.1 Introduction

En automatique pour analyser et commander un système nous avons besoin d'un modèle mathématique de comportement, qui décrit avec une certaine précision et profondeur la dynamique de ce système en régime transitoire, ce modèle mathématique peut être obtenu de deux manières.

- Par modélisation, et cela en établissant le modèle de comportement à partir d'un modèle de connaissance (électrique, mécanique, biologique,...etc.).
- Par identification, et cela par une procédure expérimentale selon un protocole bien établi, le modèle de comportement est obtenu suite à l'utilisation un algorithme d'estimation paramétrique ou non parmi tant d'autres. Si le modèle connaissance s'avère trop peu précis ou impossible à obtenir, on peut alors se sur un modèle de comportement entrée-sortie. On parle alors d'identification du système.

1.2 Exposé du problème

L'Automatique consiste en l'étude des systèmes réels des différentes disciplines scientifiques (Electronique, mécanique, thermique, chimie, écologie, biologie, économie, sociologie, physique, cosmologie...), en vue de l'analyse, de la prédiction, de la surveillance, de la commande, et / ou de l'optimisation des systèmes. La condition nécessaire pour cela est l'obtention d'un modèle mathématique du système réel (on réalise une modélisation). Un système est un objet dans lequel des variables de différents types interagissent et produisent des signaux observables. Lorsque le modèle du système n'est pas connu, il est nécessaire de procéder à son identification.

En effet, la modélisation réalisée à partir d'un comportement du système et / ou de lois physiques, consiste à déterminer la structure des équations qui régissent le comportement de ce système, et aussi, à fixer, a priori la valeur de ses paramètres (longueurs, masses, inerties, capacités, résistances, frottements...). Mais, il est souvent impossible d'obtenir une connaissance à priori complète et précise de tous les paramètres du modèle. Pour affiner et compléter cette connaissance, il est alors nécessaire de procéder à une identification du système : à partir des réactions de celui-ci à des sollicitations données et connues. On peut, si le système est observable, identifier les paramètres encore inconnus.

L'identification, bien que représentant un des grands chapitres de l'automatique, ne peut plus être considérée comme seulement une partie intégrante de cette discipline, à l'usage essentiel des stratégies de commande ou de diagnostic. Son emploi en sciences de l'ingénieur, parfois avec une terminologie différente, montre qu'elle constitue une discipline à part entière des sciences expérimentales, intimement associée à la modélisation.

L'identification, ou la recherche de modèles à partir de données expérimentales, est une préoccupation majeure dans la plupart des disciplines scientifiques. Elle désigne à la fois une démarche scientifique et un ensemble de techniques visant à déterminer des modèles mathématiques capables de reproduire aussi fidèlement que possible le comportement d'un système physique, chimique, biologique, économique...

Dans la vie courante, la plupart des systèmes peuvent être appréhendés en utilisant des modèles mentaux, sans avoir besoin de procéder à une formalisation mathématique. Par exemple, la conduite d'une automobile, nécessite de savoir, que tourner le volant vers la droite, permet de faire tourner le véhicule vers la droite.

L'ensemble des informations, de force, d'amplitude de mouvement ... étant en quelque sorte stockées dans le cerveau via le muscle. L'apprentissage de la conduite consiste en une identification par le conducteur de l'ensemble du système (véhicule, route, environnement, conducteur...). Au départ, les trajectoires sont peu sûres, et au fur et à mesure que la base d'expérience du conducteur grandit, les erreurs d'approximation du système faites par le conducteur se réduisent, jusqu'à obtention de la connaissance du système à un ordre de plus en plus élevé de la part du conducteur. Pour cela, le conducteur, à l'aide de ses organes sensoriels (œil, bras, oreille...), réalise en permanence des mesures de l'état du système en fonction des sollicitations qu'il impose à son véhicule.

L'identification des systèmes commandés à l'aide de calculateurs nécessite, en revanche de décrire leurs propriétés au moins à l'aide de valeurs mesurées (tables de valeurs) ou de représentations graphiques, afin de déterminer leur réponse à un échelon, ou en fréquence (identification non paramétrique). Pour des systèmes complexes, il est nécessaire d'utiliser des modèles mathématiques (identification paramétrique). L'utilisation de modèles mathématiques est inhérente à tous les domaines des techniques de l'ingénieur et de la physique. Elle En effet, la modélisation réalisée à partir d'un comportement du système et / ou de lois physiques, consiste à déterminer la structure des équations qui régissent le comportement de ce système, et aussi, à fixer, a priori la valeur de ses paramètres (longueurs, masses, inerties, capacités, résistances, frottements...). Mais, il est souvent impossible d'obtenir une connaissance a priori complète et précise de tous les paramètres du modèle.

L'identification, bien que représentant un des grands chapitres de l'automatique, ne peut plus être considérée comme seulement une partie intégrante de cette discipline, à l'usage essentiel des stratégies de commande ou de diagnostic. Son emploi en sciences de l'ingénieur, parfois avec une terminologie différente, montre qu'elle constitue une discipline à part entière des sciences expérimentales, intimement associée à la modélisation.

Elle aide à la conception des systèmes et est un instrument de simulation et de prédiction qui est très largement utilisé dans tous les domaines, y compris dans des activités non techniques comme l'écologie, l'économie, la sociologie ou la biologie.

Un modèle doit être construit à partir de données observées (le modèle mental de la conduite d'un véhicule est développé à l'aide de l'expérience de la conduite). Un système réel est un objet constitué d'éléments de complexité variable, à l'image même d'un modèle

mathématique. Selon LJUNG, il existe un impénétrable mais transparent écran entre la description mathématique de notre monde et le monde réel. On peut regarder à travers cette fenêtre et comparer certains aspects du système physique avec sa description mathématique, mais il est impossible d'établir une relation exacte entre les deux. La question de la susceptibilité de la nature à sa description mathématique présente de vrais et profonds aspects philosophiques. Nous devons avoir cependant une vue pragmatique des modèles. Il est préférable de choisir un modèle en considérant davantage son utilité que sa « vérité ».

En automatique, on distingue deux approches pour l'identification des systèmes :

- Approche en boucle ouverte
- Approche en boucle fermée

1.3 Algorithme général d'identification

Il faut répondre aux questions suivantes :

- Quel genre de modèle ?
- Paramétrique ou non paramétrique ?
- Quelles mesures effectuer ?
- Quel signal d'excitation ?
- Quelle structure adopter ?
- Doit-on y inclure des connaissances a priori ?
- Comment choisir le bon modèle parmi tous ceux calculés ?
- Ce modèle est-il adapté à ce que l'on va lui demander ?

Notez que le processus d'identification est un processus itératif, chaque passage apportant un peu plus de connaissances (fig 1.1). Dans le chapitre suivant nous allons présenter quelques méthodes d'identification non paramétrique.

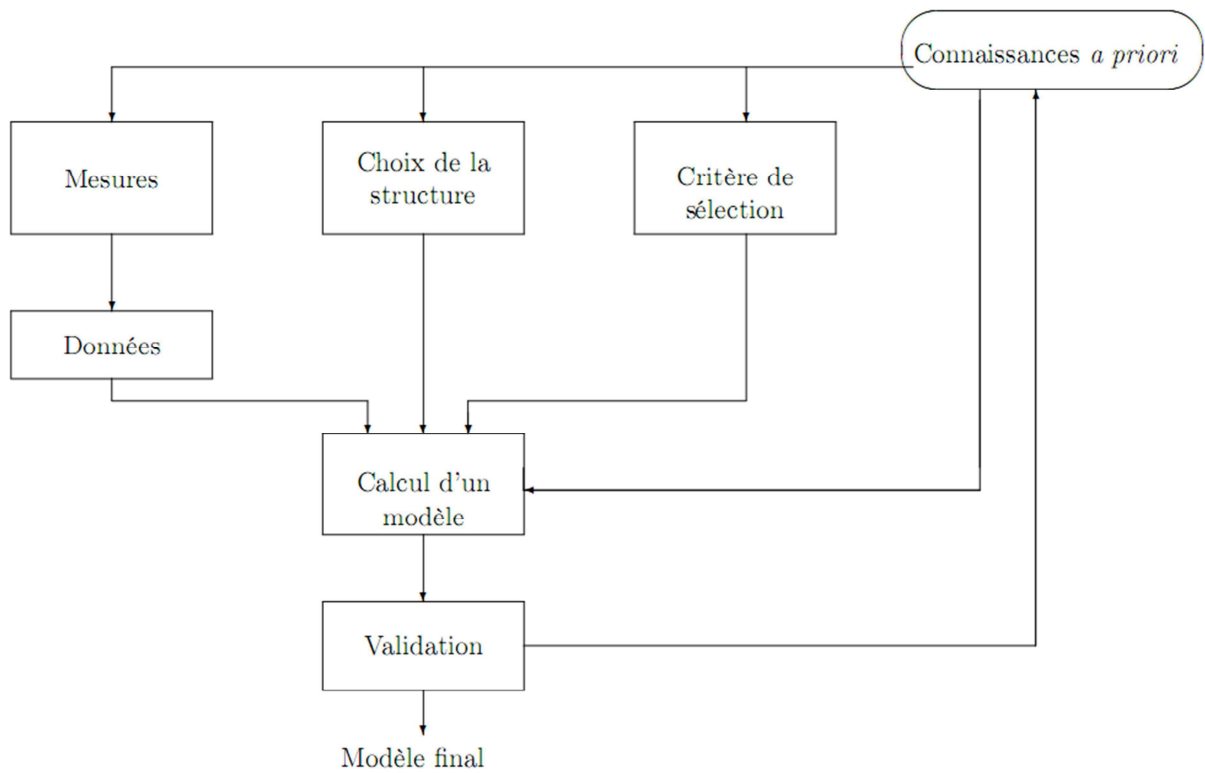


Fig. 1.1 – Algorithme général d'identification des systèmes.

La réponse à ces questions et les différentes structures et stratégies à adopter sont détaillées dans les chapitres suivants.

Chapitre 2 : Identification non paramétrique

2.1 Introduction

L'objectif est d'obtenir un modèle qui "se comporte comme" le système. La plupart du temps, les paramètres de ces modèles n'ont qu'un rapport lointain avec les paramètres d'un modèle de connaissances.

En premier lieu, il faut donc répondre aux deux questions suivantes :

- Quelle entrée ?
- Comment traiter la sortie ?

2.2 Quelle entrée ?

2.2.1 Analyse harmonique

Entrée sinusoïdale de type $u = A\sin(\omega t)$, ω balaye l'espace des pulsations susceptibles de contenir une pulsation de coupure du système. En notant l'amplitude et le déphasage de la sortie vis-à-vis de l'entrée on trace un diagramme de Bode. De l'analyse de ce diagramme on détermine le modèle.

Les résultats sont difficiles à exploiter si les constantes de temps sont proches. Bonne excitation sur tout le spectre de fréquences. Ce n'est pas une commande industrielle classique et par conséquent elle est difficile à mettre en œuvre.

2.2.2 Réponse impulsionnelle

Idéalement la meilleure méthode car le spectre est constant. Mais il est impossible de réaliser un Dirac correct, sauf peut-être en électronique...

2.2.3 Réponse indicielle

Le spectre est correct, la commande est facile à implanter car c'est une commande classique. C'est la méthode la plus utilisée.

2.3 Etude des réponses indicielles

2.3.1 Méthode de Broïda

La méthode de Broïda consiste à "faire coller" un modèle de la forme $G(s) = \frac{ke^{-T}}{1 + \tau s}$ sur la réponse du système.

Les valeurs de T et de τ sont calculées à partir des relations suivantes :

$$\begin{aligned}\tau &= 5,5 \times (t_2 - t_1) \\ T &= 2,8 \times t_1 - 1,8 \times t_2\end{aligned}\tag{2.1}$$

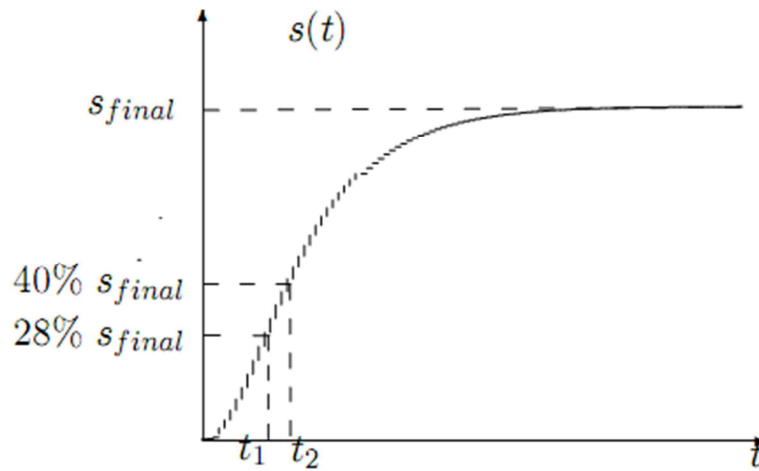


Fig. 2.1 – Méthode de Broida : Réponse indicielle.

On utilisant matlab nous pouvons extraire la valeur finale de la sortie y_f puis calculer y_1 qui correspond à $0.28*y_f$ et y_2 qui correspond à $0.4*y_f$, ayant ces deux valeurs, nous pourrions déterminer les valeurs des temps t_1 et t_2 par l'emplacement de y_1 et y_2 dans le vecteurs des mesures de y .

En pratique, la prise en compte du retard pur est difficile dans une synthèse de correcteur. On utilise donc l'approximation suivante :

$$G(s) = \frac{ke^{-T}}{1 + \tau s} \approx \frac{k(1 - Ts)}{1 + \tau s} \approx \frac{k}{(1 + \tau s)(1 + Ts)} \quad (2.2)$$

Cette approximation reste valable jusqu'à la pulsation $\omega = 1/T$

2.3.2 Méthode de Strecj

La méthode est dédiée au modèle suivant :

$$G(s) = \frac{ke^{-T}}{(1 + \tau s)^n} \quad (2.3)$$

1. Mesurer Tu/Ta .
2. Dans la colonne Tu/Ta du tableau de la figure 2.4 trouver la valeur immédiatement inférieure à ce ratio.
3. Sur la ligne de ce ratio déterminer n .
4. Toujours à l'aide des valeurs numériques de cette ligne, calculer τ avec Ta/τ .
5. Calculer la nouvelle valeur de Tu' avec Tu/τ .
6. En déduire T avec $T = Tu - Tu'$.

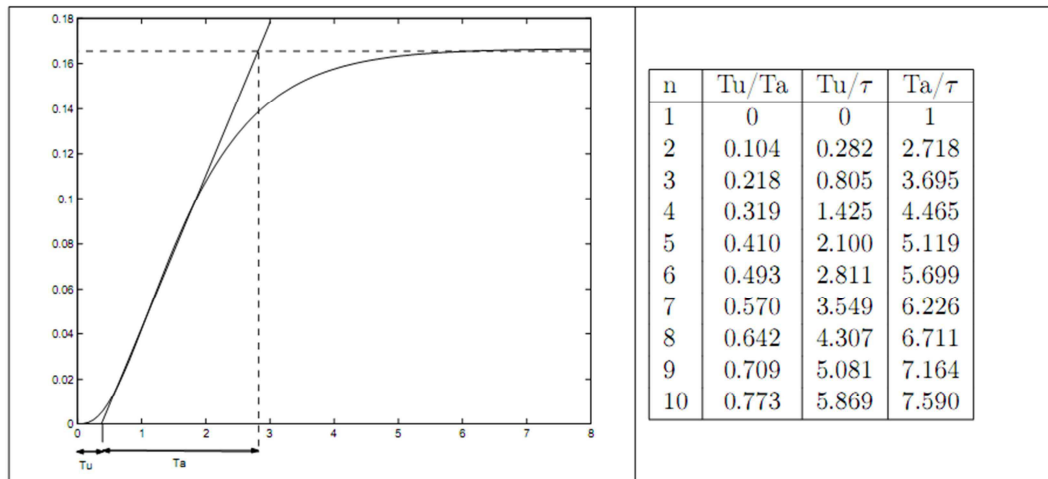


Fig 2.2 – Méthode de Strejc : Réponse indicielle.

Note : si $T_u = T_a < 0.1$, donc $n = 1$, On applique plutôt la méthode de Broïda.

2.3.3 Méthode de Strejc avec intégrateur

Identifier un modèle sous la forme :

$$G(s) = \frac{k_u}{s} \frac{1}{(1 + \tau s)^n} \quad (2.4)$$

1. Avec x/y : déterminer n

2. Calcul de τ avec $\tau = T_u / n$ (2.5)

3. Calcul de $k_u = \frac{\Delta y}{\Delta u} \times \frac{1}{\Delta t}$ (2.6)

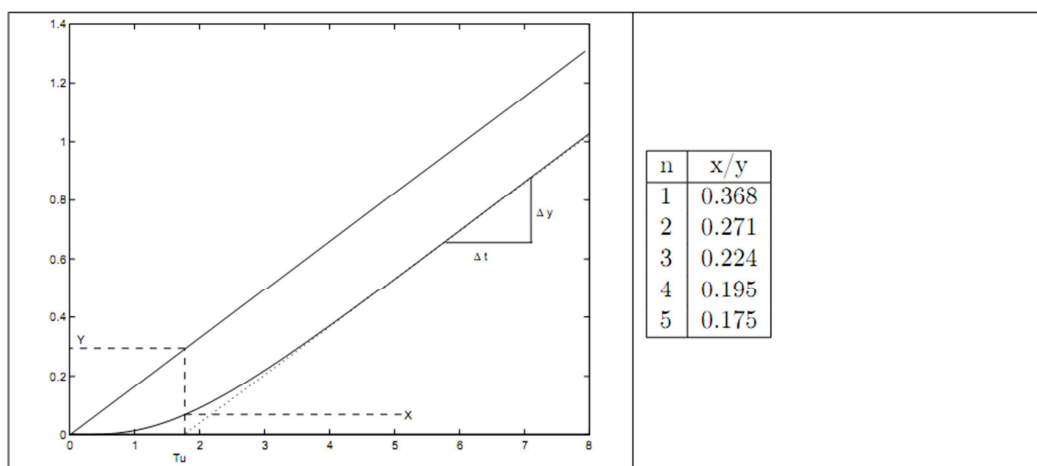


Fig 2.3 – Méthode de Strejc (système avec intégrateur).

En pratique il est assez rare d'identifier un système intégrateur en boucle ouverte car le système dérive rapidement en dehors de son domaine de linéarité (saturation, ...). En pratique, on procède à une identification en boucle fermée.

Exercice 2.1

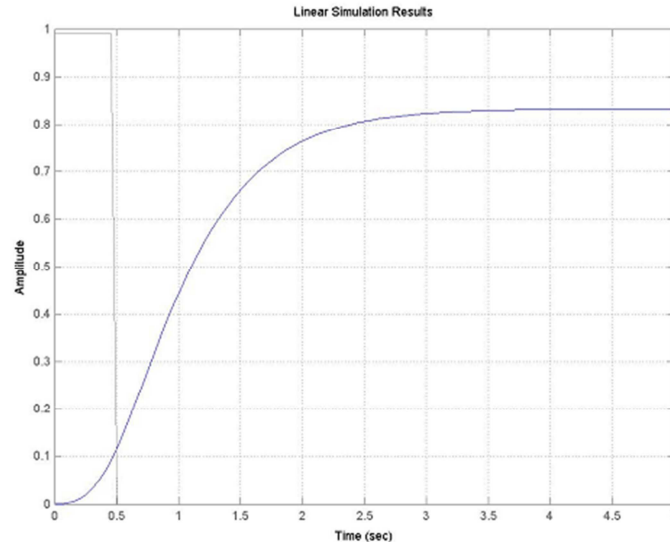


Fig. 2.4 – Réponse du système à un créneau unité de durée $t = 0,5s$.

1. Identifier le système avec la méthode de Strejc, comme si la réponse était la réponse à un échelon.
2. Déduisez-en la fonction de transfert du système.

Solution :

1. Modèle Strejc

On mesure $Tu = 0,3s$ et $Ta = 1,3s$. En appliquant la méthode :

$$Tu/Ta = 0,2308 \Rightarrow n = 2$$

$$Ta/\tau = 2,718 \Rightarrow \tau = Ta/2,718 = 0,4783$$

$$Tu/\tau = 0,282 \Rightarrow Tu' = 0,1349$$

$$Tr = Tu - Tu' \Rightarrow Tr = 0,1651$$

$$\text{Gain} : 0.83$$

d'où le modèle :

$$G(p) = \frac{Ae^{-Trp}}{(1 + \tau p)^n} = \frac{0.83e^{-0,134p}}{(1 + 0,5887p)^2}$$

2.4 Méthode de corrélation

Les méthodes de corrélation consistent à utiliser les propriétés statistiques des bruits de mesure de façon à obtenir une forme de la réponse impulsionnelle.

2.4.1 Rappels sur les signaux aléatoires

Pour des signaux aléatoires $x(t)$ et $y(t)$ on peut définir les fonctions d'autocorrélation, Σ_{xx} , et de corrélation, Σ_{xy} :

$$\begin{aligned}\Sigma_{xx}(t_1, t_2) &= E\{x(t_1)x(t_2)\}, \\ \Sigma_{xy}(t_1, t_2) &= E\{x(t_1)y(t_2)\},\end{aligned}\tag{2.7}$$

Où $E\{\cdot\}$ représente l'espérance mathématique. Ces fonctions sont différentes, mais peuvent en être déduites, des fonctions d'auto-covariance et de covariance, qui sont les fonctions de corrélation correspondant aux signaux centrés.

Dans le cas de signaux stationnaires, les quantités statistiques sont invariantes pour toute translation du temps. Les fonctions de corrélation ne dépendent plus que de l'écart $\tau = t_1 - t_2$. De plus, l'hypothèse d'ergodicité, qui est vérifiée par la plupart des signaux aléatoires utilisés en pratique, permet de remplacer l'opérateur $E\{\cdot\}$ par une moyenne temporelle. Ainsi pour des signaux aléatoires stationnaires ergodiques, ce que nous supposons toujours, les fonctions de corrélation sont données par :

$$\begin{aligned}R_{xx}(\tau) &= \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T x(t)x(t+\tau)dt, \\ R_{xy}(\tau) &= \lim_{T \rightarrow \infty} \frac{1}{2T} \int_{-T}^T x(t)y(t+\tau)dt,\end{aligned}\tag{2.8}$$

Et deux signaux sont non corrélés si pour tout τ , $R_{xy}(\tau) = 0$.

Ces fonctions vérifient quelques propriétés, que nous pouvons rappeler :

- ✓ $\forall \tau, R_{xx}(0) \geq |R_{xx}(\tau)|$;
- ✓ $\forall \tau, R_{xy}(\tau) = |R_{yx}(-\tau)|$;
- ✓ Si x et y sont non corrélés, et $z(t) = x(t) + y(t)$, alors :

$$R_{zz}(\tau) = R_{xx}(\tau) + R_{yy}(\tau);\tag{2.9}$$

- ✓ Soient u et y les entrées et sorties d'un système linéaire stationnaire de réponse impulsionnelle $h(t)$, alors :

$$R_{uy}(\tau) = \int_0^{\infty} h(t) R_{uu}(\tau - t)dt,\tag{2.10}$$

Le calcul pratique des fonctions de corrélation (2.8) doit être modifié de façon à tenir compte des impératifs techniques de mise-en-œuvre. On doit donc remplacer l'opérateur d'avance par un retard et tenir compte d'une évaluation en temps fini, on obtient alors les estimations :

$$\begin{aligned}\hat{R}_{xx}(\tau) &= \frac{1}{T} \int_0^T x(t-\tau)x(t)dt, \\ \hat{R}_{xy}(\tau) &= \frac{1}{T} \int_0^T x(t-\tau)y(t)dt,\end{aligned}\tag{2.11}$$

Dans le cas d'une implantation numérique de ces formules pour des signaux échantillonnés, il vient :

$$\begin{aligned}\hat{R}_{xx}(k) &= \frac{1}{N} \sum_{r=0}^{N-1} x_{r-k}x_r, \\ \hat{R}_{xy}(\tau) &= \frac{1}{N} \sum_{r=0}^{N-1} x_{r-k}y_r,\end{aligned}\tag{2.12}$$

2.4.2 Obtention de la réponse impulsionnelle

Supposons qu'à la sortie $y(t)$ du système à identifier, de réponse impulsionnelle $h(t)$, se superpose un bruit de mesure $b(t)$.

Lorsque l'entrée sur ce système est un signal aléatoire $u(t)$, la fonction de corrélation entre $m(t)$ et $u(t)$ s'écrit :

$$R_{um}(\tau) = \int_0^{\infty} h(t) R_{uu}(\tau-t)dt + R_{ub}(\tau).\tag{2.13}$$

Cette relation présente deux avantages par rapport à la méthode précédente de déconvolution :

- ✓ Réduction de la longueur des enregistrements : quand $u(t)$ est strictement aléatoire, $R_{uu}(\tau)$ s'annule rapidement, on a donc moins de données à traiter ;
- ✓ Réduction du niveau de bruit : lorsque $u(t)$ et $b(t)$ sont non corrélés, $R_{ub}(\tau) = 0$, et la relation (2.13) se réduit à un produit de convolution non bruité.

De plus, lorsqu'on utilise les estimations, sur une même période de temps, des fonctions de corrélations $R_{um}(\tau)$ et $R_{ub}(\tau)$, cette relation reste vraie sans approximations. En effet, pour tout intervalle $[t_1, t_2]$, on a :

$$\begin{aligned}\int_{t_1}^{t_2} u(t)y(t+\tau)dt &= \int_{t_1}^{t_2} u(t) \int_0^{\infty} h(\lambda)u(t+\tau-\lambda)d\lambda dt \\ &= \int_0^{\infty} h(\lambda) \left[\int_{t_1}^{t_2} u(t)u(t+\tau-\lambda)dt \right] d\lambda.\end{aligned}\tag{2.14}$$

Comme les signaux sont stationnaires, les estimations des corrélations sont liées, pour T dans (2.11) suffisamment grand, par :

$$\hat{R}_{uu}(\tau) = \int_0^{\infty} h(t) \hat{R}_{uu}(\tau - t) dt. \quad (2.15)$$

De façon à avoir les signaux $u(t)$ et $b(t)$ décorrélés, on excite le système par un bruit blanc défini par :

$$R_{uu}(\tau) = \delta(\tau), \quad (2.16)$$

Ce qui correspond à un spectre d'autocorrélation constamment égal à 1. La mesure de $\hat{R}_{uu}(\tau)$ donne directement la réponse impulsionnelle par :

$$\hat{R}_{uu}(\tau) = h(\tau) \quad (2.17)$$

2.4.3 Modalités d'application

Comme il n'est pas possible de construire un bruit blanc, on utilise des signaux binaires pseudo-aléatoires dont la génération et les propriétés statistiques. Nous ne détaillerons leur application que dans le cas des systèmes discrets, l'obtention de la réponse impulsionnelle des systèmes continus se faisant par la même méthode. La fonction d'autocorrélation des signaux binaires pseudo-aléatoires discrets de longueur L unités de temps, à valeurs $\{0, a\}$, est :

$$\hat{R}_{uu}(k) = a^2 \frac{L+1}{L} \sum_{n=-\infty}^{+\infty} \delta(k - nL) - \frac{a^2}{L}, \quad (2.18)$$

Qui approche la fonction d'autocorrélation d'un bruit blanc (lorsque L devient grand). Il est légitime de supposer qu'un tel signal est non corrélé avec les bruits de mesures. On obtient donc :

$$\begin{aligned} \hat{R}_{uu}(k) &= \sum_{j=0}^{+\infty} h_j \hat{R}_{uu}(k - j) \\ &= a^2 \frac{L+1}{L} \sum_{n=-\infty}^{+\infty} h_{k-nL} - \sum_{j=0}^{+\infty} \frac{a^2}{L} h_j. \end{aligned} \quad (2.19)$$

En se plaçant dans le cas où $h_j = 0$, pour $j > N$, et $L > N$ on obtient pour $k \in \{0, \dots, N\}$:

$$R_{uu}(k) = a^2 \frac{L+1}{L} h_k - \sum_{j=0}^N \frac{a^2}{L} h_j, \quad (2.20)$$

Où le second terme à droite est une constante $-a^2 K_0 / L$, K_0 correspondant au gain statique du système. On peut donc, par translation, calculer pour chaque k la valeur de la séquence de pondération.

Cet inconvénient peut être évité en utilisant des séquences ternaires pseudo-aléatoires [Foulard et al. , 1977] qui présentent l'avantage d'avoir une fonction d'autocorrélation à valeurs nulles entre pics. Cependant ces séquences sont plus délicates à mettre en œuvre, car basées sur 3 états, et demandent une durée d'expérimentation deux fois plus longue que les signaux binaires pseudo-aléatoires.

Dans le cas où on utilise une séquence binaire, Landau [Lan88] préconise quelques règles pratiques pour en choisir la longueur. Notamment, afin de déterminer le gain statique du système, il convient de choisir une séquence dont l'impulsion de durée maximale soit supérieure au temps de montée, t_M , du système. On doit donc avoir :

$$v\Delta > t_M \quad (2.21)$$

Où v est le nombre de cellules et Δ la période de l'horloge générant le signal binaire pseudo-aléatoire. D'autre part, pour balayer tout le spectre de fréquence, il faut que la longueur de l'expérience L soit supérieure à la période de la séquence pseudo-aléatoire, soit :

$$(2^v - 1)\Delta < L. \quad (2.22)$$

Lorsque la condition (2.21) n'est pas vérifiée, il est préférable, pour ne pas trop augmenter Δ plutôt que v . En pratique, on prendra une période d'horloge multiple de la période d'échantillonnage.

2.4.4 Exemple

Soit la réponse impulsionnelle $g(k)$ discrète d'un système ayant comme entrée un signal SBPA $u(k)$ d'amplitude 1, donnée par le tableau suivant :

$u(k)$	1	-1	1	1	1	-1	-1	1
$g(k)$	0	0.7	0.210	0.063	0.019	0.006	0.002	0.005

Le script sous matlab suivant permet l'estimation de cette réponse impulsionnelle par la méthode de corrélation et compare les réponses du système réel et celle obtenue par la réponse impulsionnelle estimée.

```
clear, close all, clc
uk=[1 -1 1 1 1 -1 -1];
gk=[0 0.7 0.210 0.063 0.019 0.006 0.002 0.005];
yk=conv(uk,gk);
yk=yk(1:13);
Ruu=xcorr(uk);
Ruy=xcorr(uk,yk);
Ruy=conv(Ruu,gk);
Rk0=Ruu(1:7)';
Ruyk=Ruy(1:7)';
Rk1=[Ruu(2) Ruu(1:6)]'; Rk2=[Ruu(3) Ruu(2) Ruu(1:5)]';
Rk3=[Ruu(4) Ruu(3) Ruu(2) Ruu(1:4)]';
Rk4=[Ruu(5) Ruu(4) Ruu(3) Ruu(2) Ruu(1:3)]';
Rk5=[Ruu(6) Ruu(5) Ruu(4) Ruu(3) Ruu(2) Ruu(1:2)]';
Rk6=[Ruu(7) Ruu(6) Ruu(5) Ruu(4) Ruu(3) Ruu(2) Ruu(1)]';
Rk=[Rk0 Rk1 Rk2 Rk3 Rk4 Rk5 Rk6];
gke=Rk\Ruyk;
Ruyel=conv(Ruu,gke);
yke=conv(uk,gke)';
k=1:13;
plot(k,yk)
hold on
plot(k,yke,'r')
```

Vu que les calculs des fonctions de corrélation et de convolution sont lourds, alors l'outil matlab est utilisé pour réaliser cette méthode d'identification sur cet exemple.

La fonction *xcorr.m* permet le calcul des fonctions d'autocorrélation et de corrélation et la fonction *conv.m* permet le calcul de la convolution des signaux.

Le résultat du programme est présenté par la figure suivante :

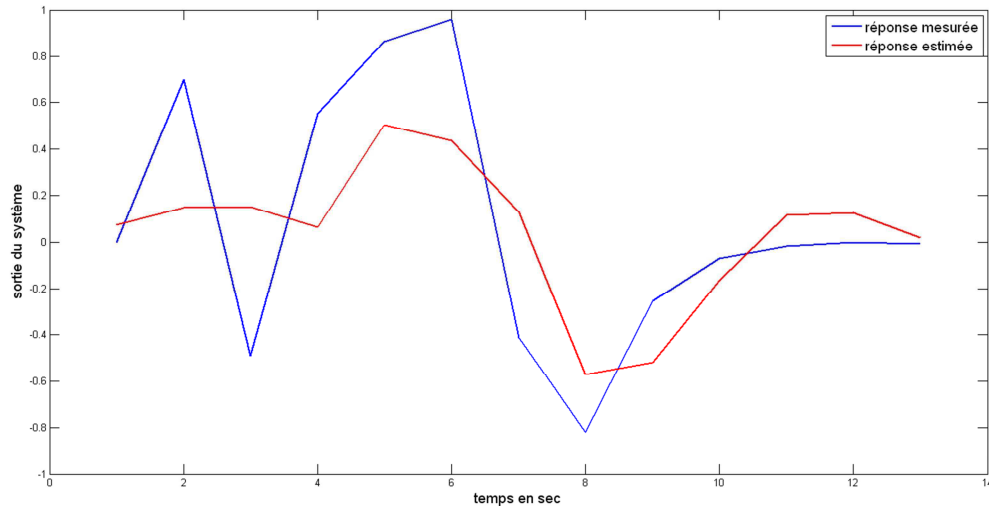


Fig. 2.5 – Réponse du système réel et par la réponse impulsionnelle estimée.

On constate que l'estimation est mauvaise au début mais s'améliore avec le temps.

Chapitre 3 : Identification paramétrique

3.1 Choix du signal d'excitation

Pour bien identifier, il faut bien exciter dans tout le spectre de fréquences susceptible de contenir des constantes de temps du système.

- $\sin(\omega t)$: parfait d'un point de vue spectre (balayage en fréquence) mais peu de systèmes acceptent ce genre d'entrées.
- $\delta(t)$: parfait du point de vue théorique, mais, hormis en électronique, il est impossible de réaliser un tel signal.
- $u(t)$: moins bon d'un point de vue spectral ($u(f) = \text{sinc}(f)$), mais facile à implanter
- $b(t)$: bruit blanc idéal d'un point de vue spectral mais comment le réaliser ?

Il arrive que vous n'ayez aucune possibilité d'exciter le système (ex : machine en production), il faudra alors profiter des commandes "naturelles" du système comme signal d'entrée du système. Dans ce cas, le premier travail consiste à calculer le spectre du signal d'entrée (FFT par exemple). Il faudra vérifier a posteriori que les constantes de temps identifiées sont bien dans des domaines de fréquences dans lesquels le système a été excité.

3.2 Les séquences binaires pseudo aléatoires (SBPA)

L'un des moyens de réaliser un signal "aléatoire" est la mise en œuvre de Séquences Binaires Pseudo Aléatoires (en anglais PRBS : Pseudo Random Binary Sequence)

Principe :

Pour que le système fonctionne, on doit initialiser le registre à n'importe quelle valeur binaire sauf zéro. le nombre codé en binaire $A = b_7b_6b_5b_4b_3b_2b_1b_0$ est "aléatoire. De la même façon, le bit b_7 semble indépendant de ses valeurs précédentes.

Une séquence sur N bits à une longueur de $2^N - 1 = L$ dont $2^{N-1} \ll 1$ et $2^{N-1} - 1 \ll 0$, sa valeur moyenne est donc non nulle. $E(s(t)) = \frac{a}{L}$

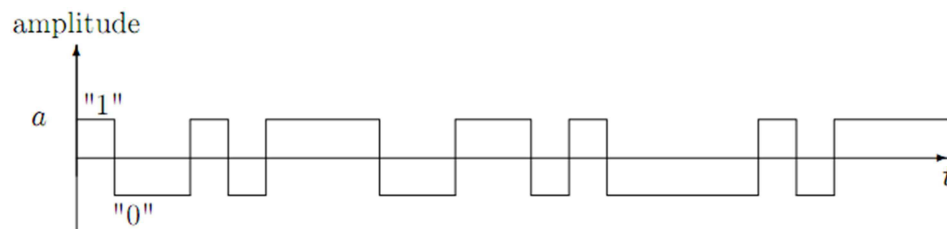


Fig 3.1 – diagramme temporel d'une séquence binaire pseudo aléatoire.

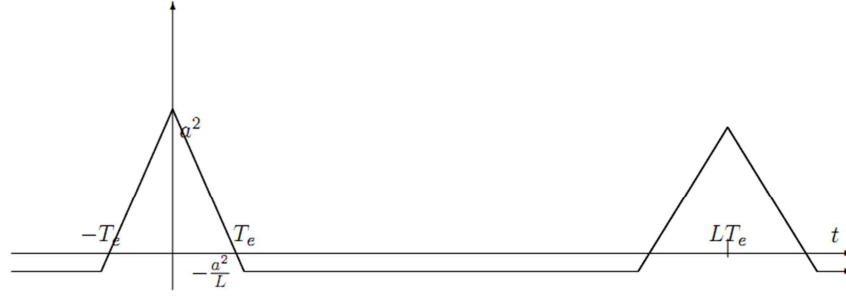


Fig 3.2 – Fonction d’auto-corrélation d’une SBPA.

Pour déterminer la forme du spectre il faut en premier lieu calculer sa fonction d’auto-corrélation, le spectre étant la transformée de Fourier de la fonction d’auto-corrélation.

Fonction d’auto-corrélation

$$C_{xx}(\tau) = \frac{1}{LT} \int_0^{LT} x(\tau) x(t-\tau) dt \quad (3.1)$$

Si $|\tau| \leq T$

$$C_{xx}(\tau) = a^2 \left(1 - \frac{L+1}{L} \frac{|\tau|}{T} \right) \quad (3.2)$$

Si $|\tau| \geq T$ $C_{xx}(\tau) = -\frac{a^2}{L}$ (3.3)

On en déduit le spectre qui est la transformée de Fourier de la fonction d’auto-corrélation.

$$P(f) = a^2 \frac{L+1}{L} \sum_{-\infty}^{+\infty} \delta\left(v - \frac{n}{LT}\right) \left(\frac{\sin \pi \frac{n}{L}}{\pi \frac{n}{L}} \right)^2 - \frac{a^2}{L} \delta(v) \quad (3.4)$$

Le spectre de la SBPA représenté en figure 3.5 est donc un peigne de Dirac modulé par un sinus cardinal. On considère que le bruit peut être considéré comme un bruit blanc dans le premier tiers du premier lobe ($\pm 1=3/Te$). On remarque donc que :

- plus L grand \Rightarrow plus la valeur moyenne est faible.
- plus L grand \Rightarrow plus de raies, plus proche du bruit blanc.

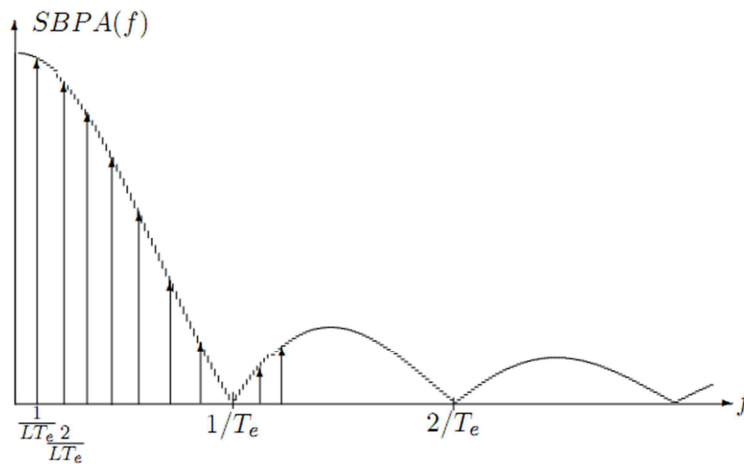


Fig. 3.3 – Spectre d'une SBPA.

3.2.1 Choix des paramètres d'une SBPA

Il faut donc déterminer a ; L et T_e "optimum" vis-à-vis du système à identifier.

Le premier compromis concerne L et T_e .

Plus la séquence est longue :

- plus il y a d'informations
- plus le transitoire est négligeable
- plus les organes souffrent
- plus les dérives s'accroissent

Le deuxième compromis est : plus a est petit,

- plus le signal est noyé dans le bruit (le vrai celui-là !)
- moins les organes souffrent
- moins les dérives s'accroissent

Le choix de a est donc avant tout lié au bruit présent dans le système. On choisira a juste assez grand pour avoir du signal en sortie du système.

Choix de L et T_e Une SBPA de longueur $L = 2N$ envoyée à la fréquence $1/T_e$ présente un plateau de longueur maximale de longueur $N \times T_e$. Le plus petit plateau étant de longueur T_e

Le choix de N (donc L) et T_e est donc, encore une fois, un compromis entre :

- une bonne identification du gain statique
- une bonne excitation sur la bande de fréquence du système

Chapitre 3 Identification paramétrique

Soit τ_{\max} la plus grande constante de temps du système et τ_{\min} la plus petite constante de temps du système.

Pour une bonne identification du gain statique on choisira :

$$NT_e = 3 \text{ à } 5 \tau_{\max} \quad (3.5)$$

Pour une bonne excitation sur le spectre entre 0 et la plus haute fréquence de coupure on choisira :

$$0,3 \frac{1}{T_e} = \frac{1}{2\pi\tau_{\min}} \quad (3.6)$$

La résolution de ses deux équations donne N et T_e

Note 1 : si $\tau_{\max} \gg \tau_{\min}$ N devient vite très grand donc LT_e devient prohibitif !

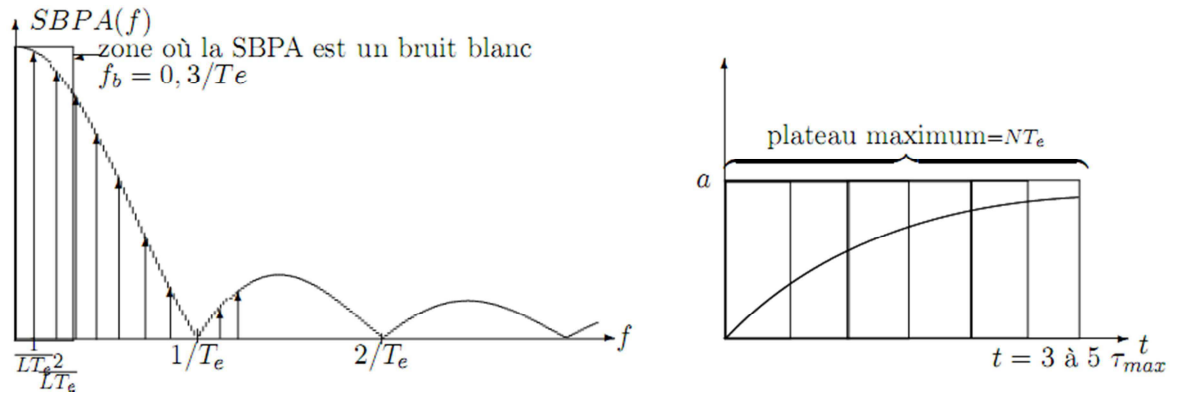


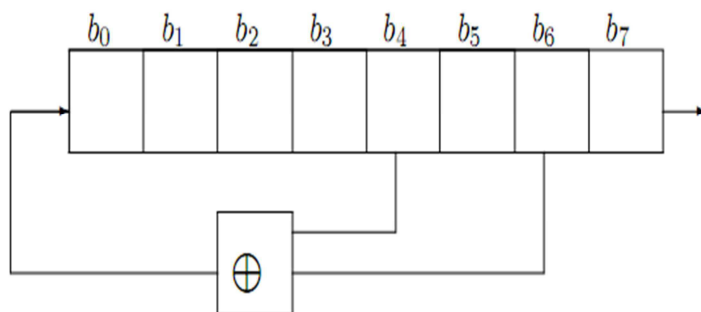
Fig. 3.4 – Paramètres d'une SBPA.

Note 2 : Si f_e est bloquée et trop grande (beaucoup de systèmes sont sur échantillonnés), il est parfaitement possible choisir $T_e = n/f_e$ en répétant chaque bit n fois

Note 3 : Si f_e est bloquée et trop petite, abandonnez l'idée d'identifier la petite constante de temps : de toutes façons vous ne pourrez pas la contrôler !

Tab. 3.1 – Tableau des bits à utiliser pour obtenir une séquence de longueur maximale.

n	bits
3	$b_2 \oplus b_1$
4	$b_3 \oplus b_2$
5	$b_4 \oplus b_2$
6	$b_5 \oplus b_4$
7	$b_6 \oplus b_5$
8	$b_7 \oplus b_3 \oplus b_2 \oplus b_1$
9	$b_8 \oplus b_4$
10	$b_9 \oplus b_6$



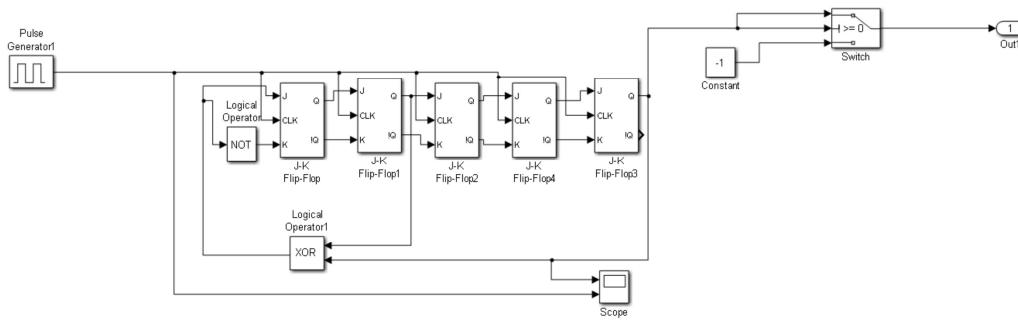


Fig. 3.5 – Programme simulink pour la génération d'une SBPA avec 5 bascules.

Le script suivant d'une fonction sous matlab, permet de générer le même signal SBPA.

```
function x=SBPA(n)

global yf
%Programme de génération d'un signal SBPA.
X1=0;X2=0;X3=0;X4=0;X5=1;S1=X5;n=50;S=[];
for i=2:length(yf)
J=xor(X2,X5);
if (J==0)
X1=0;
else
X1=1;
end
X5=X4;X4=X3;X3=X2;X2=X1;
S(i,:)=X5;
end
S=[S1;S];
e=6*ones(length(yf)-n+2,1);
x=e+0.1*S;
figure(3)
plot(x)
```

Exercice 3.1

En prenant une période d'échantillonnage $T_e=100 \text{ ms}$, quelles sont les constantes de temps τ_{\min} et τ_{\max} que l'on peut correctement identifier en utilisant une SBPA générée par 3 bascules en déduire sa longueur ?

Solution : En utilisant l'équation (3.5) nous pouvons calculer la constante de temps max, en fixant le choix de $3\tau_{\max}$ par exemple, soit avec $N=3$ (trois bascules)

$$N T_e = 5 \tau_{\max} \Rightarrow \tau_{\max} = N T_e / 5 = 3 \times 0.1 / 3 = 0.1 \text{ s}$$

L'équation (3.6) nous permet d'obtenir la constante de temps min, soit

$$0,3 \frac{1}{T_e} = \frac{1}{2\pi\tau_{\min}} \Rightarrow \tau_{\min} = \frac{T_e}{2\pi \cdot 0,3} = \frac{0,1}{0,6\pi} = 0,053 \text{ s}$$

La longueur de la séquence SBPA est donnée par $L = 2^N - 1 = 2^3 - 1 = 7$.

3.3 Identification basée sur l'erreur de sortie

Le principe de cette méthode d'identification est illustré en figure 3.5. Le modèle est une fonction de n paramètres θ_i , i variant de 1 à n . Il s'agit alors de déterminer les paramètres θ_i tels que le critère soit minimum. Le critère est en général choisi de la forme $J = \sum \varepsilon^2$.

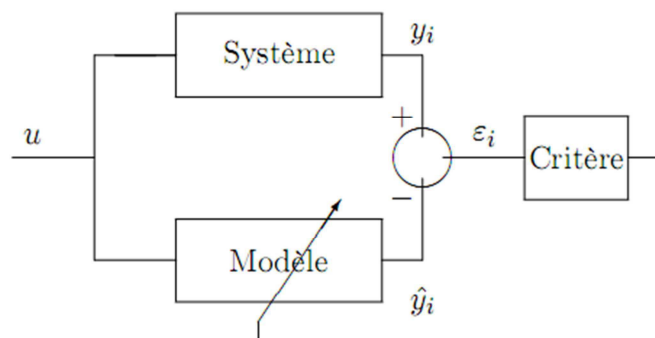


Fig. 3.6 – Principe d'identification fondé sur l'erreur de sortie.

La recherche des paramètres optimaux se fait par programmation non linéaire. IL s'agit d'utiliser un algorithme qui à partir de paramètres non optimaux θ_i et un critère J donne les paramètres .

Ces algorithmes sont nombreux, à titre de d'exemple voici les plus utilisés.

- gradient
- quasi Newton (Levenberg-Marquart)
- Nelder et Mead
- algorithmes génétiques
- recuit simulé

Avantages de cette méthode :

- pas d'hypothèse sur la forme du modèle : non linéaire.
- adapté à la recherche de paramètres physiques si modèle de connaissances (modèle continu !) L'inconvénient majeur est qu'aucun de ces algorithmes n'est capable de garantir que le résultat est réellement l'optimum.

3.4 Identification basée sur l'erreur de prédiction

Dans cette partie, nous supposons que le modèle obtenu est un prédicteur, c'est à dire qu'il permet de calculer la sortie à l'instant i en fonction des entrées et des sorties réelles aux instants précédents u_{i-k} et y_{i-k}

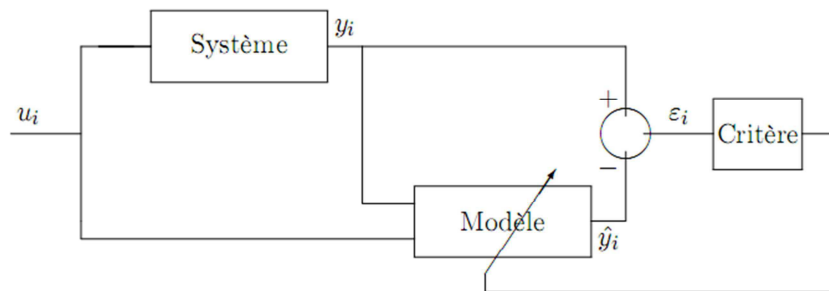


Fig. 3.7 – Principe d'identification fondée sur l'erreur de prédiction.

3.4.1 Méthode des moindres carrés simples

Les calculs suivants seront fondés sur le modèle présenté en figure 3.7.

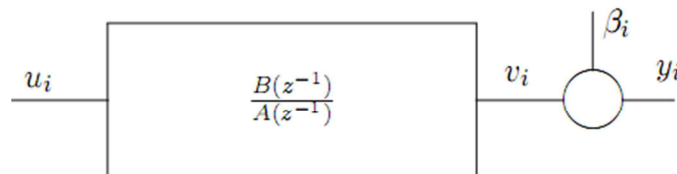


Fig. 3.8 – Modèle du système étudié.

La mise en équation du système donne :

$$\begin{aligned}
 Av_i &= Bu_i \\
 y_i &= v_i + \beta_i \\
 A(y_i - \beta_i) &= Bu_i \\
 Ay_i &= Bu_i + A\beta_i
 \end{aligned}$$

posons :

$$e_i = A\beta_i$$

e_i sont les *résidus* de l'estimation. On obtient finalement :

$$Ay_i = Bu_i + e_i \tag{3.7}$$

Chapitre 3 Identification paramétrique

Si nous possédons N mesures consécutives, on peut écrire $N - n$ fois l'équation 3.7. Où n est l'ordre, supposé connu du polynôme A , on appellera p l'ordre du polynôme B .

Sous forme matricielle, on obtient :

$$\begin{bmatrix} y_N \\ y_{N-1} \\ y_{N-2} \\ y_{N-3} \\ \vdots \\ y_{n+2} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} -y_{N-1} & -y_{N-2} & \cdots & -y_{N-n} & u_N & \cdots & u_{N-p} \\ -y_{N-2} & -y_{N-3} & & -y_{N-n-1} & u_{N-1} & & u_{N-p-1} \\ -y_{N-3} & \vdots & & \vdots & u_{N-2} & & u_{N-p-2} \\ \vdots & -y_{n+1} & & & \vdots & & u_{N-p-3} \\ -y_{n+1} & -y_n & & & \vdots & & u_{N-p} \\ -y_n & -y_{n-1} & \cdots & -y_1 & u_{n+1} & & \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \\ b_0 \\ b_1 \\ \vdots \\ b_m \end{bmatrix} + \begin{bmatrix} e_N \\ e_{N-1} \\ e_{N-2} \\ \vdots \\ e_{n+2} \\ e_{n+1} \end{bmatrix}$$

soit

$$y = X\theta + e$$

le critère J est :

$$J = \sum e^2 = e^T e$$

donc

$$J = (y - X\theta)^T (y - X\theta) = y^T y - \theta^T X^T y - y^T X\theta + \theta^T X^T X\theta$$

Nous cherchons la valeur $\hat{\theta}$ de θ qui minimise J . Ainsi

$$\left. \frac{\partial J}{\partial \theta} \right|_{\theta=\hat{\theta}} = 0 = -2X^T y + 2X^T X\theta \Big|_{\theta=\hat{\theta}}$$

D'où on en déduit :

$$\hat{\theta} = (X^T X)^{-1} X^T y \quad (3.8)$$

Il reste à vérifier que la valeur obtenue est bien un minimum

$$\left. \frac{\partial^2 J}{\partial \theta^2} \right| = 2X^T X \quad (3.9)$$

Matrice définie positive : c'est bien un minimum

Exercice 3.2

Un système sans bruit soumis à une rampe unité donne la sortie suivante.

k	0	1	2	3	4	5
u_k	0	1	2	3	4	5
y_k	0	0	7	7	42	-14

On se propose d'identifier un modèle paramétrique de la forme

$$y_k = -a_1 y_{k-1} + b_1 u_{k-1}^2$$

1. Ecrivez la relation précédente pour $k = 4$ et $k = 3$.
2. Montrez que ces deux équations peuvent se mettre sous une forme matricielle de la forme :

$$y = X\theta + e$$

3. Déterminez les coefficients a_1 et b_1 .

Solution :

$$1. \quad \begin{cases} 42 = -7a_1 + 9b_1 \\ 7 = -7a_1 + 4b_1 \end{cases}$$

2. On se propose d'identifier un modèle paramétrique de la forme :

$$\begin{bmatrix} 42 \\ 7 \end{bmatrix} = \begin{bmatrix} -7 & 9 \\ -7 & 4 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = y = X\theta + \underbrace{e}_{=0}$$

3. La solution est

$$\hat{\theta} = (X^T X)^{-1} X^T y = \begin{bmatrix} 3 \\ 7 \end{bmatrix}$$

Exercice 3.3

La réponse indicielle d'un système linéaire, stationnaire, causal et initialement au repos a donné pour les 3 premiers points d'échantillonnage $Te = 0.2s$ les valeurs suivantes :

$$y_1=0.18 ; y_2=0.33 \text{ et } y_3=0.45.$$

Le système peut être modélisé par la fonction de transfert du premier ordre :

$$G(z^{-1}) = \frac{b_1 z^{-1}}{1 + a_1 z^{-1}}$$

1. Expliciter la fonction quadratique relative à la minimisation de l'erreur de sortie.
2. Répéter le développement précédent pour l'erreur d'équation (de prédiction).
3. Déterminer les valeurs numériques de a_1 et b_1 .

Solution :

1. La sortie du modèle sur la base des entrées passées uniquement (erreur de sortie) est :

$$\hat{y}_p(k) = -\hat{a}_1 \hat{y}_p(k-1) + \hat{b}_1 u(k-1) \quad k=1, 2, 3$$

La prédiction du modèle sur la base des entrées et sorties passées (erreur d'équation) est :

$$\hat{y}(k) = -\hat{a}_1 \hat{y}(k-1) + \hat{b}_1 u(k-1) \quad k=1, 2, 3$$

En substituant les valeurs des sorties de la réponse indicielle données dans les deux équations récurrentes, on obtient pour l'erreur de sortie :

$$\begin{cases} \hat{y}_p(1) = -a_1 \hat{y}_p(0) + b_1 u(0) \\ \hat{y}_p(2) = -a_1 \hat{y}_p(1) + b_1 u(1) \\ \hat{y}_p(3) = -a_1 \hat{y}_p(2) + b_1 u(2) \end{cases} \Rightarrow \begin{cases} \hat{y}_p(1) = -a_1 \cdot 0 + b_1 \cdot 1 = b_1 \\ \hat{y}_p(2) = -a_1 \cdot b_1 + b_1 \cdot 1 = b_1(1 - a_1) \\ \hat{y}_p(3) = -a_1 \cdot b_1(1 - a_1) + b_1 \cdot 1 = b_1(1 - a_1 + a_1^2) \end{cases}$$

Pour l'erreur d'équation :

$$\begin{cases} \hat{y}(1) = -a_1 y(0) + b_1 u(0) \\ \hat{y}(2) = -a_1 y(1) + b_1 u(1) \\ \hat{y}(3) = -a_1 y(2) + b_1 u(2) \end{cases} \Rightarrow \begin{cases} \hat{y}(1) = -a_1 \cdot 0 + b_1 \cdot 1 = b_1 \\ \hat{y}(2) = -a_1 \cdot 0.18 + b_1 \cdot 1 = -0.18a_1 + b_1 \\ \hat{y}(3) = -a_1 \cdot 0.33 + b_1 \cdot 1 = -0.33a_1 + b_1 \end{cases}$$

Le critère quadratique pour minimiser l'erreur de sortie (OE) :

$$\min_{\theta} J(\theta) = \sum_{k=1}^N e_s^2(k) = \sum_{k=1}^N [y(k) - \hat{y}_p(k, \theta)]^2$$

Cela donne pour $N = 3$ et $\theta = [a_1 \quad b_1]^T$

$$\min_{\theta} \left\{ [0.18 - b_1]^2 + [0.33 - b_1(1 - a_1)]^2 + [0.45 - b_1(1 - a_1 + a_1^2)]^2 \right\}$$

L'erreur $e_s(k)$ est non linéaire par rapport aux paramètres a_1 et b_1 .

2. Le critère quadratique pour minimiser l'erreur d'équation (ARX) :

$$\min_{\theta} J(\theta) = \sum_{k=1}^N e_e^2(k) = \sum_{k=1}^N [y(k) - \hat{y}_p(k, \theta)]^2$$

Cela donne pour $N = 3$ et $\theta = [a_1 \quad b_1]^T$

$$\min_{\theta} \left\{ [0.18 - b_1]^2 + [0.33 + 0.18a_1 - b_1]^2 + [0.45 + 0.33a_1 - b_1]^2 \right\}$$

L'erreur $e_e(k)$ est linéaire par rapport aux paramètres a_1 et b_1 .

3. Identification du système (calcul des paramètres a_1 et b_1) :

L'identification se fait par la minimisation des carrées de l'erreur ce qui nécessite l'annulation du gradient du critère quadratique par rapport aux paramètres a_1 et b_1 .

3.1. Minimisation de l'erreur de sortie :

À partir du critère quadratique de l'erreur de sortie (OE) nous avons :

$$\begin{aligned} \frac{\partial J}{\partial a_1} &= 2(0.33 - b_1(1 - a_1))b_1 + 2(0.45 - b_1(1 - a_1 + a_1^2))b_1(1 - 2a_1) \\ &= 1.56b_1 - 4b_1^2 - 1.8a_1b_1 + 8a_1b_1^2 - 6a_1^2b_1^2 + 4a_1^3b_1^2 = 0 \\ \frac{\partial J}{\partial b_1} &= 2(0.18 - b_1)(-1) + 2(0.33 - b_1(1 - a_1))(-1 + a_1) + 2(0.45 - b_1(1 - a_1 + a_1^2))(-1 + a_1 - a_1^2) \\ &= -1.92 + 1.56a_1 + 6b_1 - 8a_1b_1 - 0.9a_1^2 + 8a_1^2b_1 - 4a_1^3b_1 + 2a_1^4b_1 = 0 \end{aligned}$$

On obtient un système de deux équations algébriques non linéaire à deux inconnues est de degré 5 qui n'est pas aisé de résoudre. Il existe plusieurs méthodes numériques pour résoudre un tel système, la plus efficace est la méthode de Newton-Raphson.

Chapitre 3 Identification paramétrique

Il possède donc plusieurs solutions. La résolution numérique nécessite une condition initiale relativement proche de la solution, une identification antérieure peut fournir les valeurs initiaux de l'algorithme numérique de résolution, tel $\theta_0 = [-0.6 \ 0.1]^T$:

La solution est donnée par la fonction *fsolve.m* sous matlab par :

$$a_1 = -0.8191 \quad \text{et} \quad b_1 = 0.1809$$

On obtient ainsi la fonction de transfert :

$$G(z^{-1}) = \frac{0.1809z^{-1}}{1 - 0.8191z^{-1}}$$

3.2. Minimisation de l'erreur d'équation :

À partir du critère quadratique de l'erreur d'équation nous avons :

$$\begin{aligned} \frac{\partial J}{\partial a_1} &= 2(0.33 + 0.18a_1 - b_1)0.18 + 2(0.45 + 0.33a_1 - b_1)0.33 \\ &= 0.4158 + 1.02b_1 + 0.2826a_1 = 0 \end{aligned}$$

$$\begin{aligned} \frac{\partial J}{\partial b_1} &= 2(0.18 - b_1)(-1) + 2(0.33 + 0.18a_1 - b_1)(-1) + 2(0.45 + 0.33a_1 - b_1)(-1) \\ &= -1.92 - 1.02a_1 + 6b_1 = 0 \end{aligned}$$

On obtient un système de deux équations algébriques linéaire à deux inconnues qui possède une solution unique et donc aisément calculable.

On forme le système d'équation sous forme matricielle puis on le résout:

$$\begin{bmatrix} 0.2826 & -1.02 \\ -1.02 & 6 \end{bmatrix} \begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} -0.4158 \\ 1.92 \end{bmatrix} \Rightarrow \begin{bmatrix} \hat{a}_1 \\ \hat{b}_1 \end{bmatrix} = \begin{bmatrix} 0.2826 & 1.02 \\ -1.02 & 6 \end{bmatrix}^{-1} \begin{bmatrix} -0.4158 \\ 1.92 \end{bmatrix}$$

$$\hat{a}_1 = -0.819 \quad \text{et} \quad \hat{b}_1 = 0.181$$

On obtient ainsi la fonction de transfert :

$$G(z^{-1}) = \frac{0.181z^{-1}}{1 - 0.819z^{-1}}$$

3.4.2 Calcul du biais de l'estimateur

$$\begin{aligned} E[\hat{\theta}] &= E[(X^T X)^{-1} X^T y] \quad \text{et} \quad y = X\theta + e \\ &= E[\theta + (X^T X)^{-1} X^T e] \\ &= \theta + E[(X^T X)^{-1} X^T e] \end{aligned}$$

L'estimateur est non biaisé si

$$E[\hat{\theta}] = \theta$$

donc si

$$E[(X^T X)^{-1} X^T e] = 0 \tag{3.10}$$

Chapitre 3 Identification paramétrique

Il faut donc que : X et e soient non corrélés

e soit centré.

Mais ce n'est pas le cas. Le calcul de ce biais sur l'exemple présenté en figure 3.8 permet de le montrer.

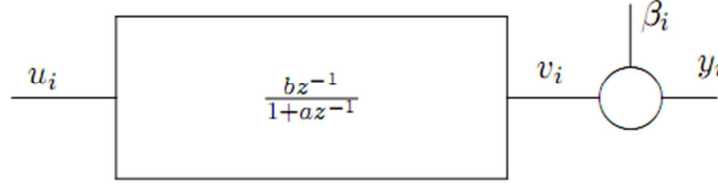


Fig. 3.9 – Système du premier ordre bruité.

Les équations du système sont :

$$\begin{aligned} v_i + av_{i-1} &= bu_{i-1} \\ y_i &= v_i + \beta_i \end{aligned} \quad (3.11)$$

Où β_i est un bruit répondant aux deux équations suivantes :

$$\begin{aligned} E[\beta_i] &= 0 \\ E[\beta_i, \beta_{i-k}] &= \sigma^2 \delta(k) \end{aligned} \quad (3.12)$$

En réécrivant les équations du système on obtient :

$$\begin{aligned} y_i + ay_{i-1} &= bu_{i-1} + \beta_i + a\beta_{i-1} \\ &= bu_{i-1} + e_i \end{aligned}$$

$$\begin{aligned} E[e_i] &= E[\beta_i + a\beta_{i-1}] = E[\beta_i] + aE[\beta_{i-1}] = 0 \\ E[e_i, e_{i-k}] &= E[(\beta_i + a\beta_{i-1})(\beta_{i-k} + a\beta_{i-k-1})] \\ &= E[\beta_i, \beta_{i-k}] + aE[\beta_i, \beta_{i-k-1}] + aE[\beta_{i-1}, \beta_{i-k}] + a^2E[\beta_{i-1}, \beta_{i-k-1}] \\ &= \sigma^2[(1 + a^2)\delta(k) + a\delta(k-1) + a\delta(k+1)] \end{aligned} \quad (3.13)$$

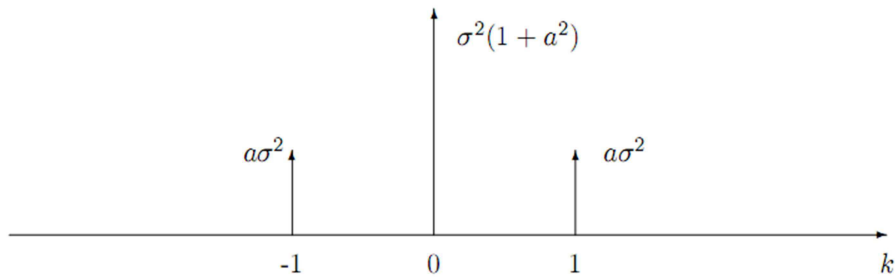


Fig. 3.10 – Fonction d'auto-corrélation des résidus d'estimation.

La fonction d'auto-corrélation (voir fig. 3.9) montre bien que les résidus e_i ne sont pas un bruit blanc, notre estimateur est bien biaisé ! Il reste donc à trouver une méthode qui permette de rendre ce biais nul. Deux méthodes sont proposées ci-après : La méthode des moindres carrés généralisés et la méthode de la matrice instrumentale.

3.4.3 Méthode de la Matrice Instrumentale

Puisque le biais de l'estimateur des moindres carrés est biaisé à cause de la corrélation entre X et e , on se propose de déterminer une autre matrice qui permette le calcul de θ tout en évitant cette corrélation et donc le biais de l'estimateur.

Posons :

$$\hat{\theta} = (Z^T X)^{-1} Z^T y \quad (3.14)$$

Où Z est la matrice instrumentale.

Quelles sont les conditions sur Z pour que l'équation précédente ait un sens ?

En poursuivant le calcul :

$$\hat{\theta} = (Z^T X)^{-1} Z^T (X\theta + e)$$

$$\hat{\theta} = \theta + (Z^T X)^{-1} Z^T e$$

Pour que

$$E[\hat{\theta}] = \theta$$

il faut :

$$E[Z^T X] \neq 0 \text{ et } E[Z^T e] = 0 \quad (3.15)$$

Plusieurs choix de Z sont possibles, en voici deux :

Le premier choix consiste à faire deux campagnes de mesures avec la même entrée u_i , le plus souvent consécutives (décalées dans le temps). Vous avez alors la possibilité de déterminer deux matrices X soit X_1 et X_2 . En posant

$$\hat{\theta} = (X_2^T X_1)^{-1} X_2^T y_1 \quad (3.16)$$

Il n'y a plus de corrélation entre X_2 et y_1 (le bruit est stochastique et ergodique) le biais est bien nul. La deuxième proposition consiste à créer les données nécessaires à la création de la matrice X_2 à l'aide d'un modèle de types moindres carrés simples issu de la première campagne de mesure.

3.5 Identification en boucle fermée

Ce domaine de l'identification est actuellement très actif. En effet ces modèles sont nécessaires dans le cadre de la commande prédictive. Plus pragmatiquement, rappelons que le plus souvent l'identification en milieu industriel s'effectue sur une machine en production. Il

Chapitre 3 Identification paramétrique

La méthode "Joint Input-Output" Avec cette méthode on se propose de déterminer simultanément les deux modèles :

- le modèle du système
- le modèle du correcteur.

Ceci s'effectue facilement en utilisant les modèles paramétriques d'état sous Matlab.

$$\begin{bmatrix} y \\ v \end{bmatrix} = Au + B\xi \quad (3.19)$$

La matrice A contient alors les dynamiques du correcteur G_{uv} et celle du système en boucle fermée G_{uy} . Comme pour la méthode précédente on remonte au modèle du système par calcul.

Exercice 3.4

Soit la fonction de transfert identifiée en boucle fermée par la méthode indirecte. Et soit un correcteur PI de forme parallèle associé à cette fonction de transfert donné par :

$$M(s) = \frac{30(s+1)}{s^2 + 31s + 30} \quad \text{et} \quad H(s) = 10 + \frac{10}{s}$$

Déduisez le modèle du système en boucle ouverte.

Solution :

Avant d'utiliser l'équation (3.18) nous devons d'abord écrire la fonction de transfert du correcteur, soit $H(s) = 10 + \frac{10}{s} = \frac{10(s+1)}{s}$

$$G(s) = \frac{M(s)}{H(s) - M(s)H(s)} = \frac{\frac{30(s+1)}{s^2 + 31s + 30}}{\frac{10(s+1)}{s} - \left(\frac{30(s+1)}{s^2 + 31s + 30} \right) \left(\frac{10(s+1)}{s} \right)}$$

$$G(s) = \frac{30s(s+1)}{10(s+1)(s^2 + 31s + 30 - 30s - 30)} = \frac{3s}{(s^2 + s)} = \frac{3}{s+1}$$

3.6 Boite d'outil d'identification sous matlab « identool »

Matlab est avant tout un outil de calcul ! Matlab propose donc toute une série d'outils d'aide à l'identification. En particulier, quatre modèles linéaires, les plus couramment utilisés sont des fonctions de Matlab. Dans le tableau 3.2 ces modèles sont présentés avec leurs utilisations "classique" et la méthode de détermination des paramètres. Notez qu'une lecture de la documentation concernant la boîte à outils "identool" est une excellente activité !

Chapitre 3 Identification paramétrique

Face à la facilité de calcul de modèles, le problème n'est plus vraiment de choisir une structure adaptée mais faire le tri parmi les quantités de modèles calculés.

3.7 Exemple d'identification par la toolbox identtool

1^{ère} étape : Tapez *ident* sur l'éditeur de command window de matlab.

La GUI suivante de la toolbox s'ouvre.

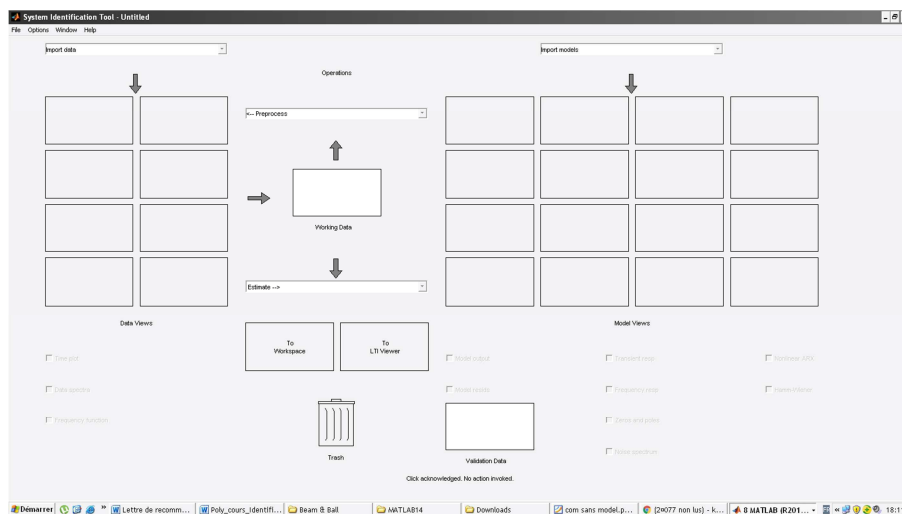


Fig. 3.12 – La GUI de la toolbox identtool de matlab.

2^{ème} étape : cliquez sur import data à gauche de la fenêtre de la GUI. Choisissez le type de données expérimentales utilisées pour votre identification. Par exemple time domain data.

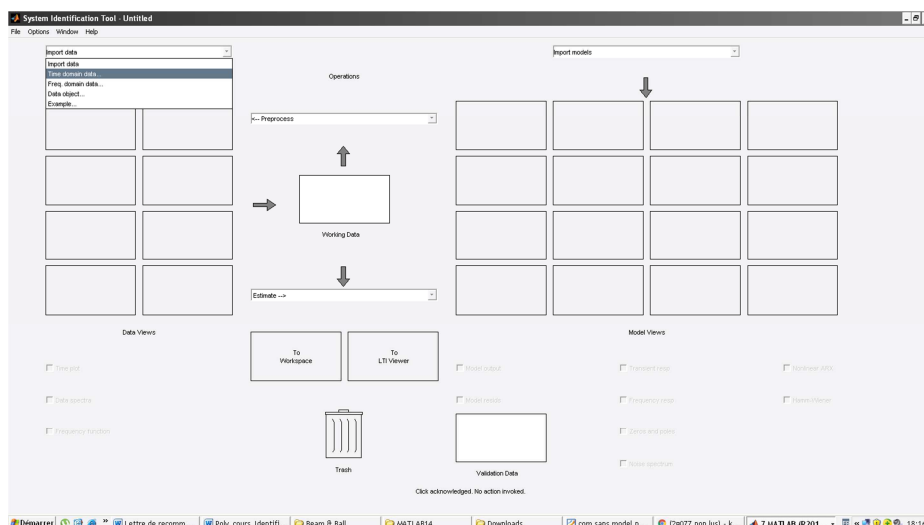


Fig. 3.13 – L'importation des données expérimentales temporels.

Chapitre 3 Identification paramétrique

Alors une autre fenêtre s'ouvre pour importer ces données.

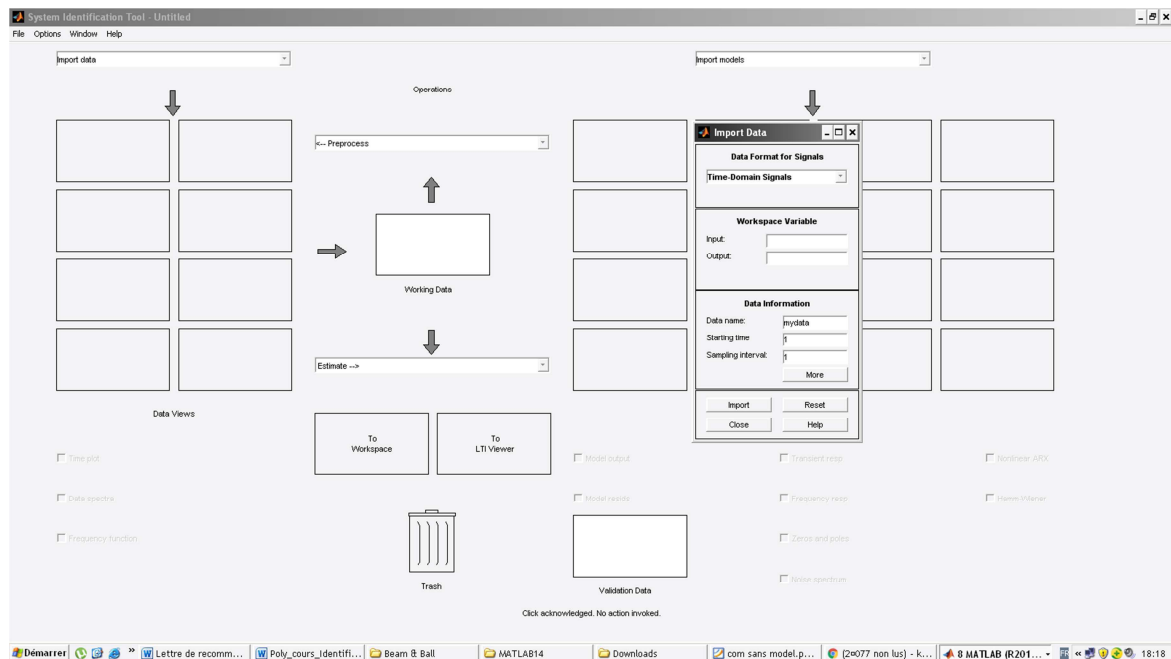


Fig. 3.14 – Fenêtre des données d'entrée et de sortie avec période d'échantillonnage.

3^{ème} étape : remplissez cette fenêtre par les symboles *u* et *y* qui désignent respectivement les signaux d'entrée et de sortie ainsi que la période d'échantillonnage de l'acquisition de ces données. Vous pouvez aussi leur attribuer un nom dans *mydata*.

4^{ème} étape : cliquez sur import, alors un signal apparaît sur une subdivision de la fenêtre de la GUI, identool.

5^{ème} étape : filtrez les données si nécessaire en utilisant la rubrique Preprocess, puis les insérées sous le nom de *mydata* dans une autre subdivision de la GUI.

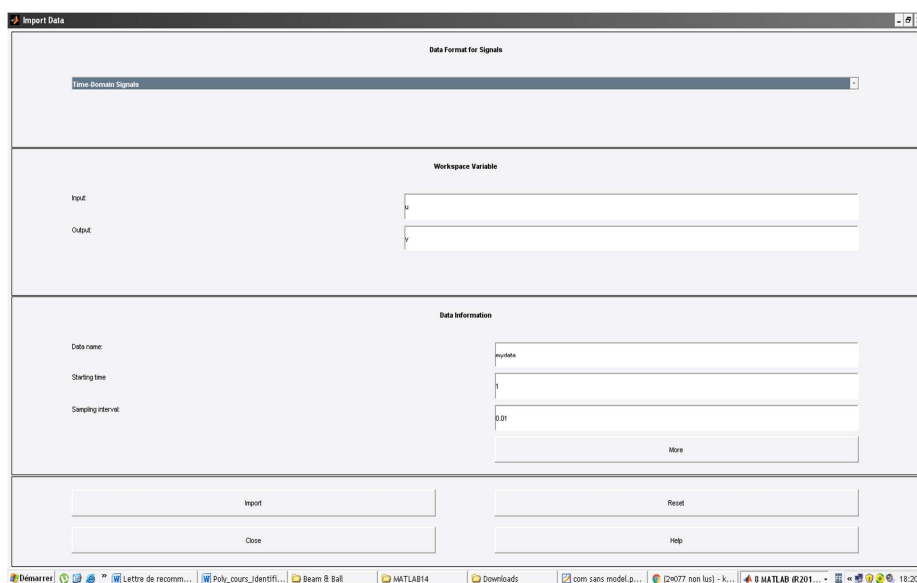


Fig. 3.15 – La fenêtre import Data de la GUI de la toolbox identool.

Chapitre 3 Identification paramétrique

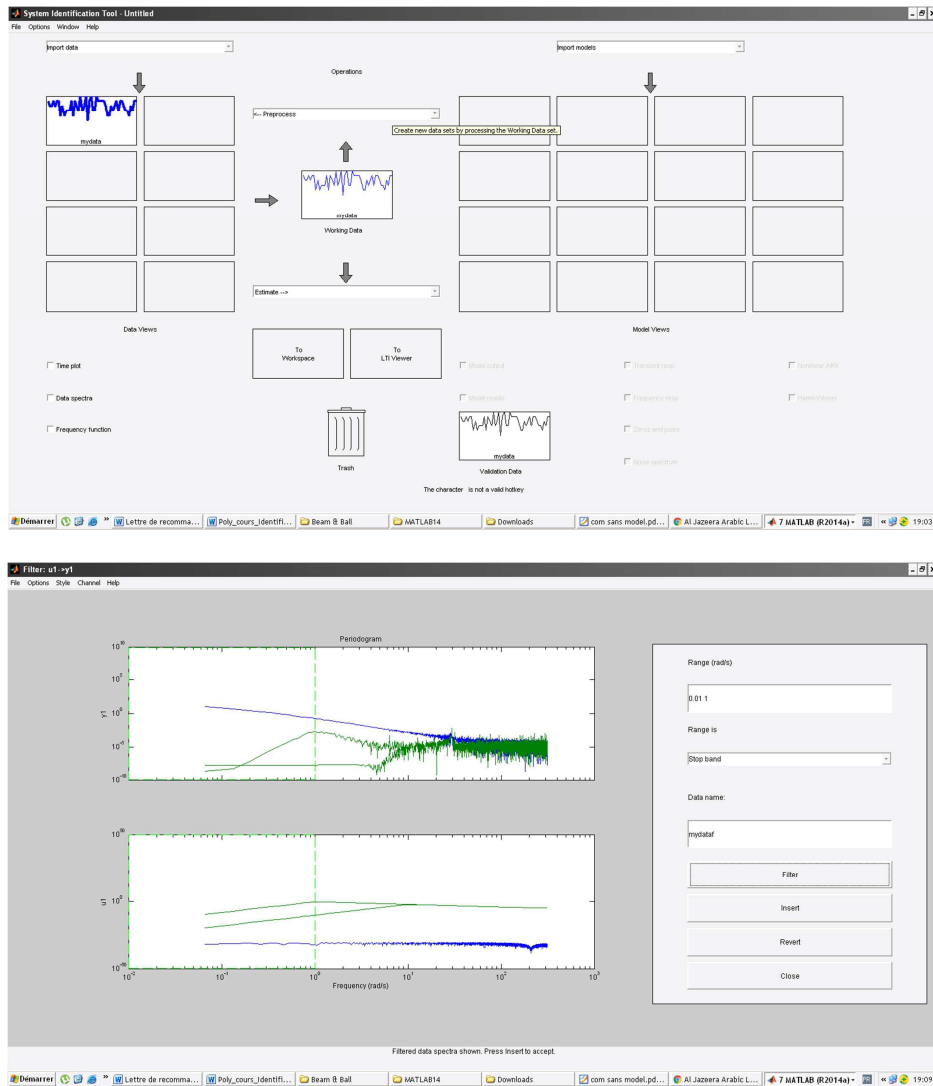


Fig. 3.16 – Fenêtre du filtrage des données comportant le diagramme de Bode.

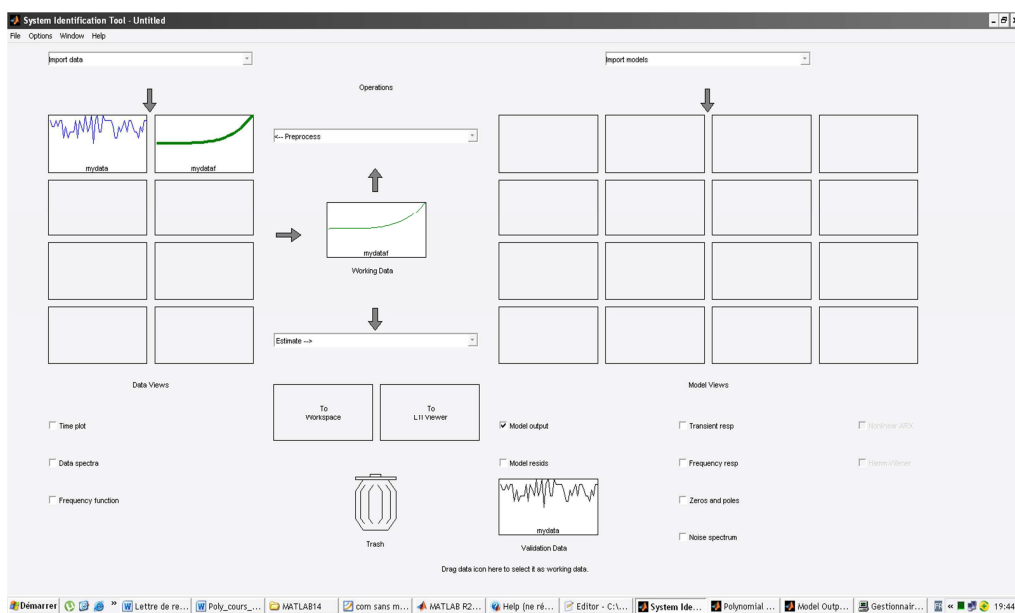


Fig. 3.17 – Fenêtre des données filtrées.

Chapitre 3 Identification paramétrique

6^{ème} étape : cliquez sur la rubrique Estimate et choisissez Polynomial models, alors une fenêtre regroupant toutes les méthodes d'identification paramétriques dédiées à la fonction de transfert.

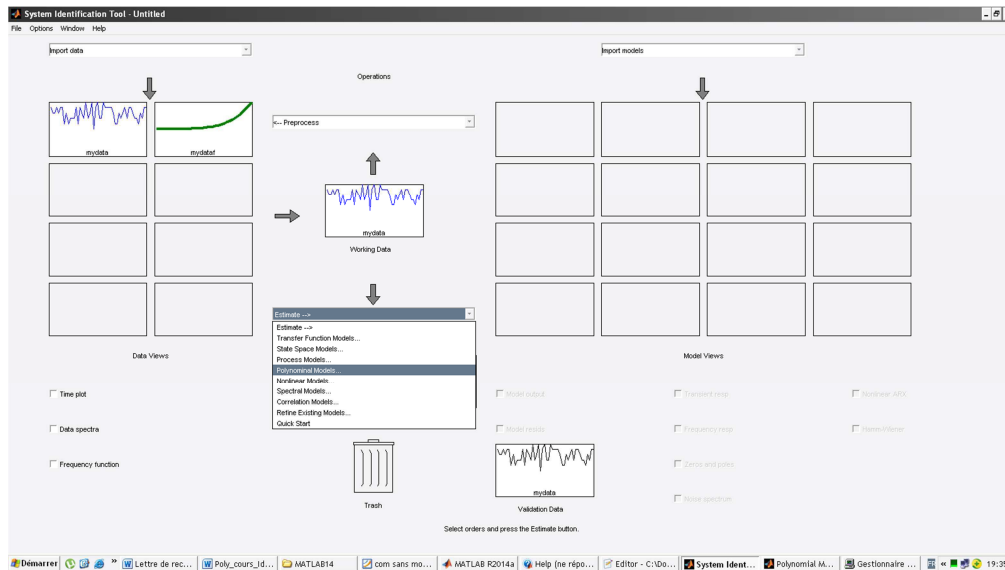


Fig. 3.18 – Fenêtre du choix de la méthode d'estimation.

7^{ème} étape : Choisissez la méthode désirée par exemple ARX et spécifiez les ordres du numérateur, dénominateur et le retard (1 s'il n'y a pas de retard) du modèle choisi, puis cliquez sur le bouton Estimate, quand le processus d'estimation est achevé, le résultat s'affiche sur une subdivision à droite de la fenêtre de la GUI avec le nom de l'algorithme muni des ordres du modèle choisi par exemple arx221.

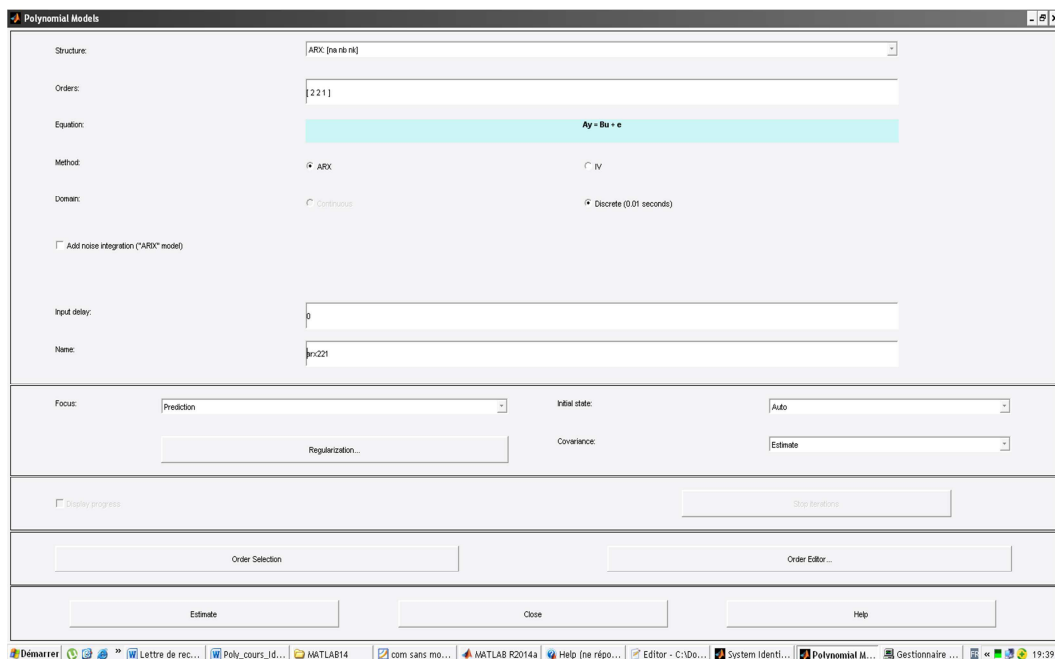


Fig. 3.19 – Fenêtre de l'estimation des paramètres de la GUI identool.

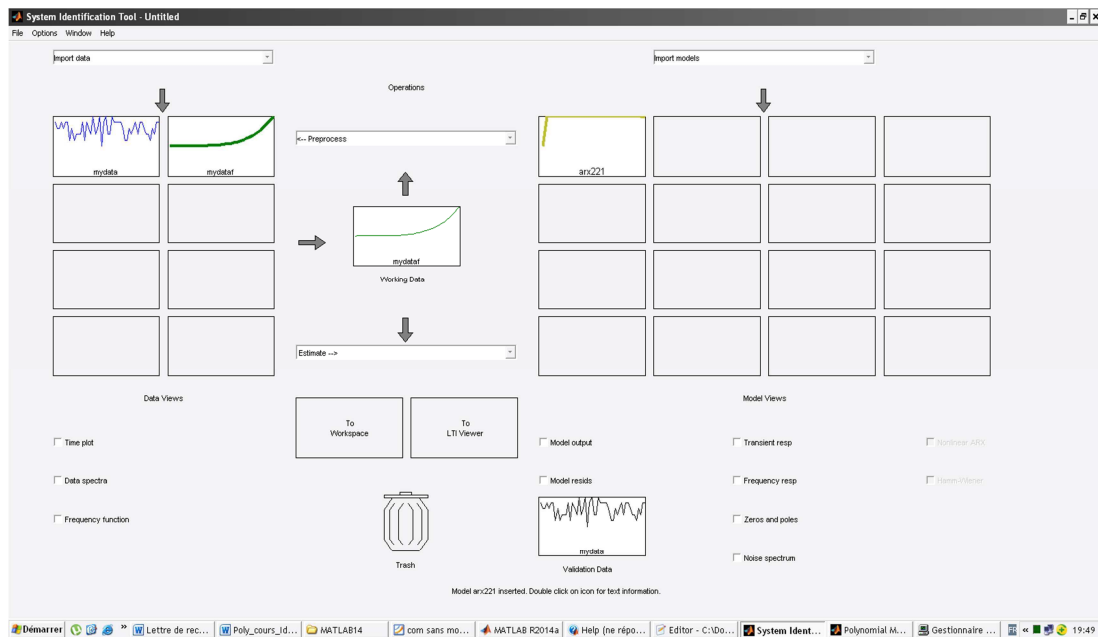


Fig. 3.20 – Fenêtre du résultat d'estimation obtenu.

8^{ème} étape : cochez la case Model output en bas de la GUI, la réponse du modèle identifié est tracée avec la réponse réelle du système en mentionnant un critère de validation du résultat en pourcentage.

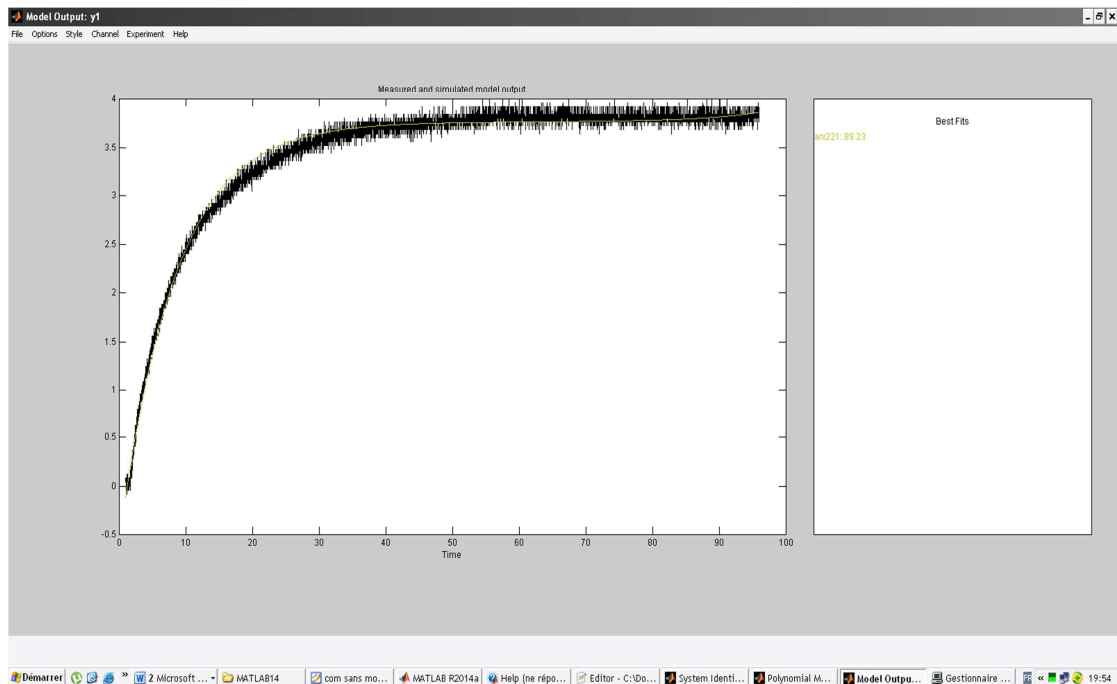


Fig. 3.21 – Fenêtre de la validation du modèle estimé (sortie mesurée et sortie prédite).

Chapitre 3 Identification paramétrique

9^{ème} étape : glissez la subdivision du résultat sur la subdivision To workspace en bas de la GUI, pour obtenir les paramètres estimés du modèle de la fonction de transfert validée ainsi que la moyenne des carrés de l'erreur d'estimation MSE.

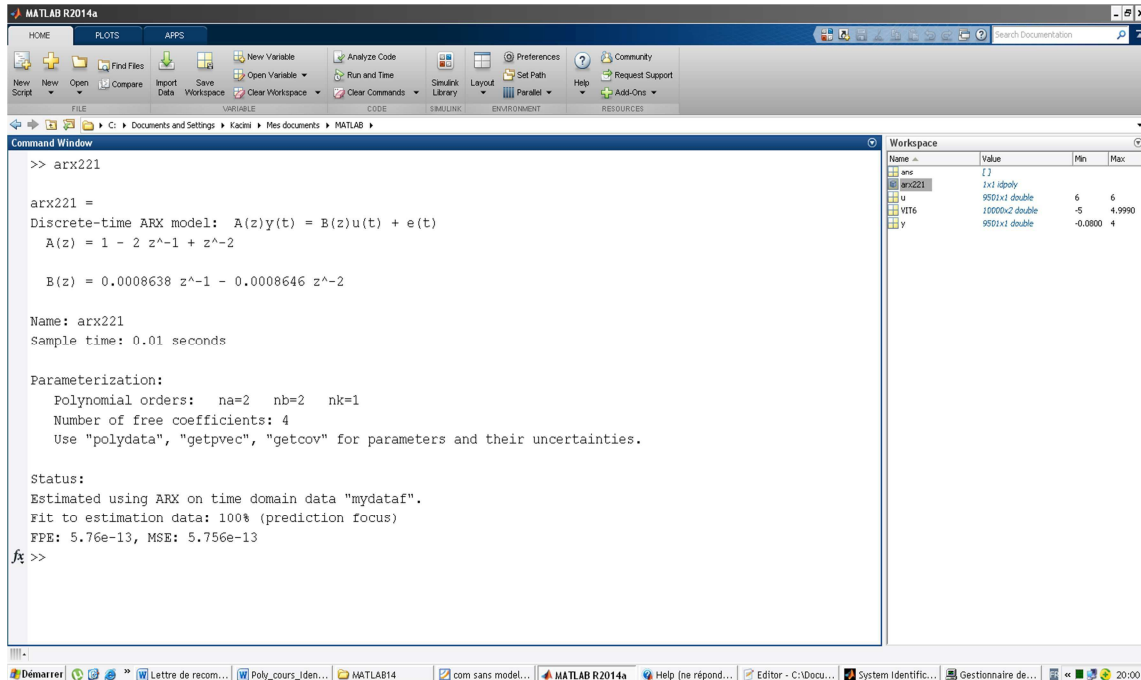
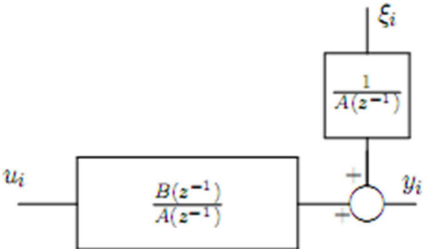
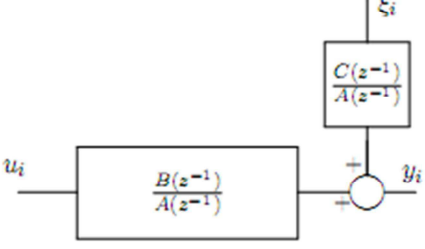
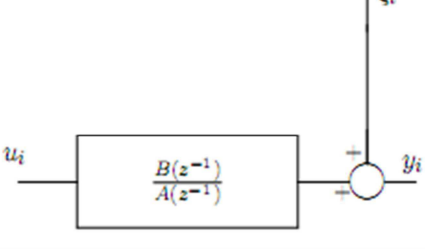
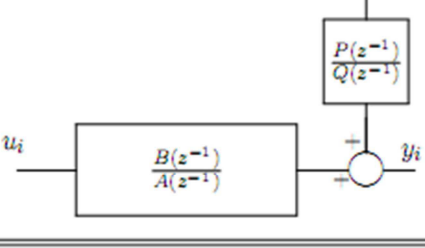


Fig. 3.22 – Visualisation des résultats de l'identification sur command window.

D'autres algorithmes peuvent être testés afin de valider le modèle qui correspond aux meilleurs résultats obtenus.

Tab. 3.2 – Modèles paramétriques implantés sous Matlab.

	<p>Modèle ARX Le modèle le plus simple, donne souvent de bons résultats. le seul hic est le traitement du bruit qui est soumis à la même dynamique que l'entrée. A utiliser en première approximation ou lorsque le bruit est surtout à l'entrée.</p> <p>Méthode de détermination des paramètres : Moindres carrés ou Matrice instrumentale</p>
	<p>Modèle ARMAX Proche du modèle ARX, il s'utilise dans les mêmes cas. Il permet en outre de créer un modèle de bruit un peu plus réaliste. C'est le modèle le plus utilisé.</p> <p>Méthode de détermination des paramètres : Maximum de vraisemblance</p>
	<p>Modèle OE Bien que semblant plus simple que les précédents, le calcul des paramètres s'avère plus difficile. Parfait lorsque le bruit est surtout un bruit de capteur, donc proche de la sortie.</p> <p>Méthode de détermination des paramètres : Maximum de vraisemblance</p>
	<p>Modèle de Box-Jenkins Le modèle complet par excellence, la dynamique différente pour l'entrée et le bruit en font un bon modèle.</p> <p>Méthode de détermination des paramètres : Maximum de vraisemblance</p>

3.8 Cas pratique

Afin de mieux maîtriser la toolbox de matlab identool précédemment présentée dans ce chapitre, nous le concluons avec le cas pratique suivant :

On désire identifier un moteur à courant continu de vitesse nominale correspondant à une alimentation en tension continue de 6volts, deux expériences ont été réalisées en boucle ouverte sur la plateforme du banc d'essai de ce moteur, avec deux échelons de tension d'alimentation de 6v et 8v ajoutées par un signal SBPA d'amplitude de 0.1v.

Les fichiers Excel des deux expériences sont nommés VIT6.slx et VIT8.slx

1. Ramener les deux fichiers sur workspace de matlab en utilisant la fonction import data puis sauvegardez-les en fichier.mat.
2. Préparez les données d'identification on utilisant le fichier VIT6.mat.
3. Ouvrir la boite outil d'identification de matlab en tapant ident.
4. Sur la gui de la toolbox, cliquez sur import time data avec :
 $u=6*\text{ones}(\text{length}(\text{VIT6}),1)$; $y=\text{VIT6}$ et sample = 0.025. (période d'échantillonnage).
5. Sur le bouton d'estimate sélectionnez polynomial, commencez par la méthode arx, Essayez différents nombres de pôles et de zéros, comparez les résultats.
6. Faites la même chose avec les méthodes armax et oe.
7. Validez le modèle qui donne le plus grand pourcentage du critère de similitude.
8. Quelle est la méthode qui donne les meilleurs résultats ? pourquoi selon vous ?

Chapitre 4 : Estimations récursives

4.1 Moindres carrés récursifs

L'estimation de paramètres par la méthode des moindres carrés simples présente un inconvénient majeur, la nécessité de calculer l'inverse d'une matrice, ce qui est long et parfois impossible sur un microcontrôleur. On se propose à travers cet exercice de déterminer une forme récursive de cette estimation.

Les avantages d'une formulation récursive tiennent essentiellement en deux points.

- La possibilité de traiter un plus grand nombre de données que dans le cas de la formulation directe (pas de pseudo-inverse à calculer), notamment dans le cas de l'implantation sur un microcontrôleur.

- Dans le cas des systèmes variant dans le temps, la forme récursive permet de "suivre" les paramètres du système. Dans ce cas, l'identification en ligne est généralement suivie d'une commande qui elle aussi "s'adapte" aux paramètres courants, on entre alors dans le domaine des commandes auto-adaptatives qui dépassent le cadre de ce cours.

Comme dans tout problème récursif on s'intéresse d'abord à la boucle, ensuite comment on en sort et enfin comment on y entre. Supposons que nous possédions une estimation des paramètres à l'instant N :

$$\theta_N = (X_N^T X_N)^{-1} X_N^T Y_N$$

à l'instant suivant la nouvelle estimation est :

$$\theta_{N+1} = (X_{N+1}^T X_{N+1})^{-1} X_{N+1}^T Y_{N+1}$$

Avec :

$$X_{N+1} = \begin{bmatrix} x_{N+1} \\ X_N \end{bmatrix} \quad , \quad Y_{N+1} = \begin{bmatrix} y_{N+1} \\ Y_N \end{bmatrix} \quad \text{et} \quad E_{N+1} = \begin{bmatrix} e_{N+1} \\ E_N \end{bmatrix}$$

où $x_{N+1} = [-y_N - y_{N-1} \dots - y_{N-n+1} u_{N+1} \dots u_{N-p+1}]$
 θ_{N+1} s'écrit alors :

$$\begin{aligned} \hat{\theta}_{N+1} &= (X_{N+1}^T X_{N+1})^{-1} X_{N+1}^T Y_{N+1} \\ &= (X_N^T X_N + x_{N+1}^T x_{N+1})^{-1} (X_N^T Y_N + x_{N+1}^T y_{N+1}) \end{aligned}$$

En utilisant le lemme d'inversion matricielle :

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1}$$

On obtient :

$$\hat{\theta}_{N+1} = \left((X_N^T X_N)^{-1} - (X_N^T X_N)^{-1} x_{N+1}^T [1 + x_{N+1} (X_N^T X_N)^{-1} x_{N+1}^T]^{-1} x_{N+1} (X_N^T X_N)^{-1} \right) (X_N^T Y_N + x_{N+1}^T y_{N+1})$$

En posant $a = 1 + x_{N+1} (X_N^T X_N)^{-1} x_{N+1}^T$

$$\begin{aligned} \hat{\theta}_{N+1} &= \left((X_N^T X_N)^{-1} - (X_N^T X_N)^{-1} x_{N+1}^T [1 + x_{N+1} (X_N^T X_N)^{-1} x_{N+1}^T]^{-1} x_{N+1} (X_N^T X_N)^{-1} \right) (X_N^T Y_N + x_{N+1}^T y_{N+1}) \\ \hat{\theta}_{N+1} &= \left((X_N^T X_N)^{-1} - (X_N^T X_N)^{-1} x_{N+1}^T a^{-1} x_{N+1} (X_N^T X_N)^{-1} \right) (X_N^T Y_N + x_{N+1}^T y_{N+1}) \\ &= \underbrace{(X_N^T X_N)^{-1} X_N^T Y_N}_{\hat{\theta}_N} + (X_N^T X_N)^{-1} x_{N+1}^T y_{N+1} - \underbrace{(X_N^T X_N)^{-1} x_{N+1}^T a^{-1} x_{N+1} (X_N^T X_N)^{-1} X_N^T Y_N}_{\hat{\theta}_N} \\ &\quad - (X_N^T X_N)^{-1} x_{N+1}^T a^{-1} x_{N+1} (X_N^T X_N)^{-1} x_{N+1}^T y_{N+1} \\ &= \hat{\theta}_N + (X_N^T X_N)^{-1} x_{N+1}^T y_{N+1} - (X_N^T X_N)^{-1} x_{N+1}^T a^{-1} x_{N+1} \hat{\theta}_N \\ &\quad - (X_N^T X_N)^{-1} x_{N+1}^T a^{-1} x_{N+1} (X_N^T X_N)^{-1} x_{N+1}^T y_{N+1} \\ &= \hat{\theta}_N - (X_N^T X_N)^{-1} x_{N+1}^T a^{-1} (x_{N+1} \hat{\theta}_N) \\ &\quad + (X_N^T X_N)^{-1} x_{N+1}^T a^{-1} (-x_{N+1} (X_N^T X_N)^{-1} x_{N+1}^T y_{N+1}) + (X_N^T X_N)^{-1} x_{N+1}^T y_{N+1} \\ &= \hat{\theta}_N - (X_N^T X_N)^{-1} x_{N+1}^T a^{-1} (x_{N+1} \hat{\theta}_N) \\ &\quad + (X_N^T X_N)^{-1} x_{N+1}^T (-a^{-1} x_{N+1} (X_N^T X_N)^{-1} x_{N+1}^T y_{N+1} + y_{N+1}) \\ &= \hat{\theta}_N - (X_N^T X_N)^{-1} x_{N+1}^T a^{-1} (x_{N+1} \hat{\theta}_N) \\ &\quad + (X_N^T X_N)^{-1} x_{N+1}^T a^{-1} (-x_{N+1} (X_N^T X_N)^{-1} x_{N+1}^T + a) y_{N+1} \\ &= \hat{\theta}_N - (X_N^T X_N)^{-1} x_{N+1}^T a^{-1} (x_{N+1} \hat{\theta}_N) \\ &\quad + (X_N^T X_N)^{-1} x_{N+1}^T a^{-1} y_{N+1} \\ &= \hat{\theta}_N + (X_N^T X_N)^{-1} x_{N+1}^T a^{-1} (y_{N+1} - x_{N+1} \hat{\theta}_N) \end{aligned}$$

On obtient finalement une formule de récurrence :

$$\hat{\theta}_{N+1} = \hat{\theta}_N + (X_N^T X_N)^{-1} x_{N+1}^T a^{-1} (y_{N+1} - x_{N+1} \hat{\theta}_N)$$

Le terme $(y_{N+1} - x_{N+1} \hat{\theta})$ représente l'erreur d'estimation à l'aide des paramètres précédents.

L'équation est alors : la nouvelle estimation est l'ancienne estimation corrigée par un terme proportionnel à l'erreur d'estimation précédente, que l'on peut récrire sous la forme.

$$\hat{\theta}_{N+1} = \hat{\theta}_N + K_{N+1} (y_{N+1} - x_{N+1} \hat{\theta}_N)$$

L'algorithme général est donc :

$$\hat{\theta}_{N+1} = \hat{\theta}_N + K_{N+1} (y_{N+1} - x_{N+1} \hat{\theta}_N) \quad (4.1)$$

$$K_{N+1} = P_N x_{N+1}^T (I + x_{N+1} P_N x_{N+1}^T)^{-1} \quad (4.2)$$

$$P_{N+1} = P_N - K_{N+1} x_{N+1} P_N \quad (4.3)$$

On remarquera une analogie avec le filtre de Kalman, en fait c'est bien l'équivalent d'un filtre de Kalman appliqué sur le système suivant :

La condition de convergence est la même : e_N doit être un bruit blanc.

L'initialisation de l'algorithme se fait de deux façons :

- Si on connaît une première estimation de θ (en provenance d'une méthode non récursive par exemple), on l'utilise. Dans ce cas on prend $P_0 = \lambda I$ avec λ petit (petite variance du bruit)
- Si on ne connaît pas de première approximation, on prend θ_0 quelconque et $P_0 = \lambda I$ avec λ grand (grande variance du bruit).

Cet algorithme ne peut être utilisé que si les paramètres du système sont constants, en effet lorsque N tend vers l'infini, P_N tend vers 0, alors une variation même importante des paramètres n'influe plus sur l'estimation de θ . Notons encore que sur le plan numérique la forme $P_{N+1} = P_N - K_{N+1}x_{N+1}P_N$ est très mal conditionnée, on lui préfère la forme :

$$P_{N+1} = (I - K_{N+1}x_{N+1})P_N(I - K_{N+1}x_{N+1})^T + K_{N+1}K_{N+1}^T$$

Si les paramètres évoluent brusquement, une solution consiste à réinitialiser $P_N = \lambda I$ avec λ grand si les paramètres évoluent lentement on peut utiliser :

$$\hat{\theta}_{N+1} = \hat{\theta}_N + K_{N+1} (y_{N+1} - x_{N+1}\hat{\theta}_N) \quad (4.4)$$

$$K_{N+1} = P_N x_{N+1}^T (\lambda I + x_{N+1} P_N x_{N+1}^T)^{-1} \quad (4.5)$$

$$P_{N+1} = (P_N - K_{N+1}x_{N+1}P_N) \frac{1}{\lambda} \quad (4.6)$$

Cependant cet algorithme présente l'inconvénient de faire croître P_N de façon exponentielle s'il n'y a plus d'excitation.

Voici d'autres choix pertinents

- à gain constant : on force alors $P_{N+1} = P_N$; cela évite la diminution du gain en cours de recherche des paramètres, et on donne ainsi plus de poids aux acquisitions les plus récentes ; cette option convient bien à un système dont les paramètres varient.

- à gain décroissant : $P_N = cste$, cas classique, qui convient à un système à paramètres constants.

- à trace constante : on garde $\text{tr}(P_N) = cste$ (multiplication par un facteur correctif à chaque itération) ; on a les mêmes avantages que dans la recherche à gain constant mais on module en cours de recherche le poids relatif de chaque paramètre.

Cette méthode présente les mêmes avantages que la méthode des moindres carrés non récursive, mais aussi le même inconvénient : l'estimateur est biaisé ! Aussi on lui préfère généralement d'autres méthodes telles que la méthode de la variable instrumentale récursive.

4.2 Variable instrumentale récursive

L'algorithme, très proche de celui énoncé en

$$\hat{\theta}_{N+1} = \hat{\theta}_N + K_{N+1} (y_{N+1} - x_{N+1} \hat{\theta}_N) \quad (4.7)$$

$$K_{N+1} = P_N Z_{N+1}^T (\sigma^2 + x_{N+1} P_N Z_{N+1}^T)^{-1} \quad (4.8)$$

$$P_{N+1} = P_N - K_{N+1} x_{N+1} P_N \quad (4.9)$$

Dans ce cas Z_{N+1} représente un vecteur ligne instrumental composé un peu comme la matrice instrumentale, soit d'observations retardées de la sortie soit de la sortie d'un modèle auxiliaire.

Beaucoup d'autres algorithmes existent, le lecteur intéressé en trouvera pléthore dans la littérature. Néanmoins ils sont tous à peu près fondés sur le même principe. En fait, ils se résument à l'expression d'un asservissement à gain réglable : on asservit les paramètres pour annuler une erreur, d'équation ou de sortie selon la méthode. La valeur du gain influe, comme sur n'importe quel système, sur la stabilité, soit la convergence de la méthode, et sur la rapidité.

Moduler le gain revient donc à moduler la rapidité de convergence : si l'on a affaire à un procédé dont il faut poursuivre les paramètres, qui varient effectivement dans le temps, on aura intérêt à avoir un gain fort ; si par contre le système est fortement bruité, on a intérêt à avoir un gain faible, sinon la sortie du modèle va poursuivre le bruit en faisant varier les paramètres à chaque période d'échantillonnage, ce qui n'a pas de sens physique. Un bon réglage du gain demande plusieurs essais et pas mal de bon sens.

Exercice 4.1

Considérons le système à temps discret décrit par l'équation de récurrence suivante :

$$y(k+1) + ay^2(k) = bu(k)$$

Le signal d'excitation $u(k)$ est un échelon unitaire et la sortie $y(k)$ mesurée est donnée par le tableau :

k	0	1	2	3	4	5	6
$y(k)$	0	0.01	1.05	1.69	3.02	7.4	39.3

Tableau 4.1 : Données mesurées.

1. Déterminez les paramètres estimés \hat{b} par la méthode des moindres carrés ordinaire.
2. Supposons qu'une nouvelle mesure $y(7) = 1082$ est disponible. Trouvez les nouvelles valeurs de \hat{b} en utilisant la méthode des moindres carrés récursives.

Solution :

1. Pour $k = 1, 2, 3, 4$, et 5 nous avons :

$$y(2) = -ay^2(1) + bu(1)$$

$$y(3) = -ay^2(2) + bu(2)$$

$$y(4) = -ay^2(3) + bu(3)$$

$$y(5) = -ay^2(4) + bu(4)$$

$$y(6) = -ay^2(5) + bu(5)$$

Ces équations peuvent être écrites sous forme matricielle comme suite :

$$Y = \Phi \theta$$

Où

$$Y = [y(2) \ y(3) \ y(4) \ y(5) \ y(6)]^T = [1.05 \ 1.69 \ 3.02 \ 7.4 \ 39.3]^T$$

$$\Phi = \begin{bmatrix} -y^2(1) & u(1) \\ -y^2(2) & u(2) \\ -y^2(3) & u(3) \\ -y^2(4) & u(4) \\ -y^2(5) & u(5) \end{bmatrix} = \begin{bmatrix} -0.0001 & 1 \\ -1.1025 & 1 \\ -2.8561 & 1 \\ -9.1204 & 1 \\ -54.76 & 1 \end{bmatrix} \quad \text{et} \quad \theta = \begin{bmatrix} a \\ b \end{bmatrix}$$

$$\Phi^T \Phi = \begin{bmatrix} 3091.2 & -67.8 \\ -67.8 & 5 \end{bmatrix} \quad \text{et} \quad \Phi^T Y = \begin{bmatrix} -2230 \\ 52.5 \end{bmatrix}$$

$$(\Phi^T \Phi)^{-1} = \begin{bmatrix} 0.0005 & 0.0063 \\ 0.0063 & 0.2848 \end{bmatrix}$$

Finalement les paramètres a et b calculés par la méthode des moindres carrés ordinaire sont donnés par :

$$\hat{\theta} = \begin{bmatrix} \hat{a} \\ \hat{b} \end{bmatrix} = (\Phi^T \Phi)^{-1} \Phi^T Y = \begin{bmatrix} 0.0005 & 0.0063 \\ 0.0063 & 0.2848 \end{bmatrix} \begin{bmatrix} -2230 \\ 52.5 \end{bmatrix} = \begin{bmatrix} -0.6994 \\ 1.0025 \end{bmatrix}$$

2. En utilisant la méthode des moindres carrés récursives, nous obtenons :

$$P(6) = (\varphi(6) \varphi^T(6))^{-1} = \left(\begin{bmatrix} -y^2(5) \\ u(5) \end{bmatrix} \begin{bmatrix} -y^2(5) & u(5) \end{bmatrix} \right)^{-1} = \begin{bmatrix} 8.9919 \cdot 10^6 & -3000 \\ -3000 & 1 \end{bmatrix}$$

$$\text{Ou} \quad P(6) = (\Phi^T \Phi)^{-1} = \begin{bmatrix} 0.0005 & 0.0063 \\ 0.0063 & 0.2848 \end{bmatrix}$$

$$\hat{\theta}(7) = \hat{\theta}(6) + P(7) \varphi(7) [y(7) - \varphi(7) \hat{\theta}(6)]$$

$$\text{Avec} \quad \hat{\theta}(6) = \begin{bmatrix} -0.6994 \\ 1.0025 \end{bmatrix} \quad \text{et} \quad y(7) = 1082.$$

$$\varphi^T(7) = [-y^2(6) \ u(6)] = [-1544.5 \ 1]$$

$$P(7) = P(6) - \frac{P(6) \varphi(7) \varphi^T(7) P(6)}{1 + \varphi^T(7) P(6) \varphi(7)}$$

$$P(7) = \begin{bmatrix} 0.0005 & 0.0063 \\ 0.0063 & 0.2848 \end{bmatrix} - \frac{1}{1174.6} \begin{bmatrix} 0.5867 & 7.2348 \\ 7.2348 & 89.2184 \end{bmatrix}$$

$$P(7) = \begin{bmatrix} 5.1313\text{e-}07 & 0.0001 \\ 0.0001 & 0.2088 \end{bmatrix}$$

$$P(7)\varphi(7) = \begin{bmatrix} -0.0007 \\ -0.0087 \end{bmatrix} \quad \text{et} \quad \varphi^T(7)\hat{\theta}(6) = \begin{bmatrix} -1544.5 & 1 \end{bmatrix} \begin{bmatrix} -0.6994 \\ 1.0025 \end{bmatrix} = 1081.2$$

$$\hat{\theta}(7) = \begin{bmatrix} \hat{a} \\ \hat{b} \end{bmatrix} = \begin{bmatrix} -0.6994 \\ 1.0025 \end{bmatrix} + \begin{bmatrix} -0.0007 \\ -0.0087 \end{bmatrix} (1082 - 1081.2) = \begin{bmatrix} -0.6999 \\ 0.9963 \end{bmatrix}$$

4.3 Cas pratique

La même expérience utilisée dans le chapitre précédent pour illustrer l'identification par la méthode des moindres carrés simple, sera exploitée pour appliquer l'identification par la méthode des moindres carrés récursive.

On désire identifier un moteur à courant continu de vitesse nominale correspondant à une alimentation en tension continue de 6volts, deux expériences ont été réalisées en boucle ouverte sur la plateforme du banc d'essai de ce moteur, avec deux échelons de tension d'alimentation de 6v et 8v ajoutées par un signal SBPA d'amplitude de 0.1v.

Les fichiers Excel des deux expériences sont nommés VIT6.slx et VIT8.slx

1. Ramener les deux fichiers sur workspace de matlab en utilisant la fonction import data puis sauvegardez-les en fichier.mat.
2. Ouvrir le fichier simulink *rls_mot1.slx* et visualisez les différents blocks utilisés dans ce programme de simulation.
3. Lancez le programme et visualisez les résultats et les différentes figures.
4. Refaire la même procédure avec le fichier VIT8.slx
5. Comparez les résultats obtenus par les deux simulations.

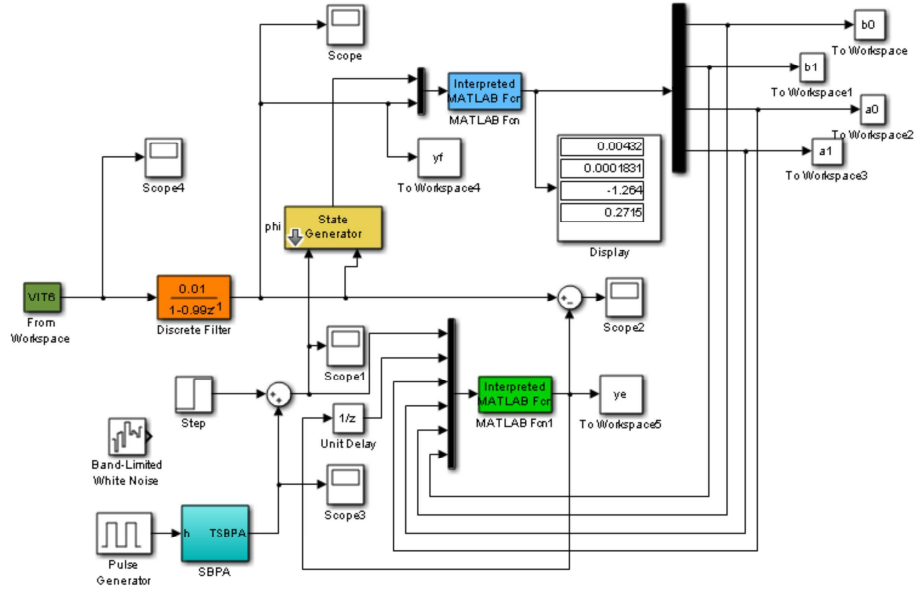


Figure 4.1 programme de simulation de l'identification par les moindres carrés récursives appliquée au moteur à courant continu.

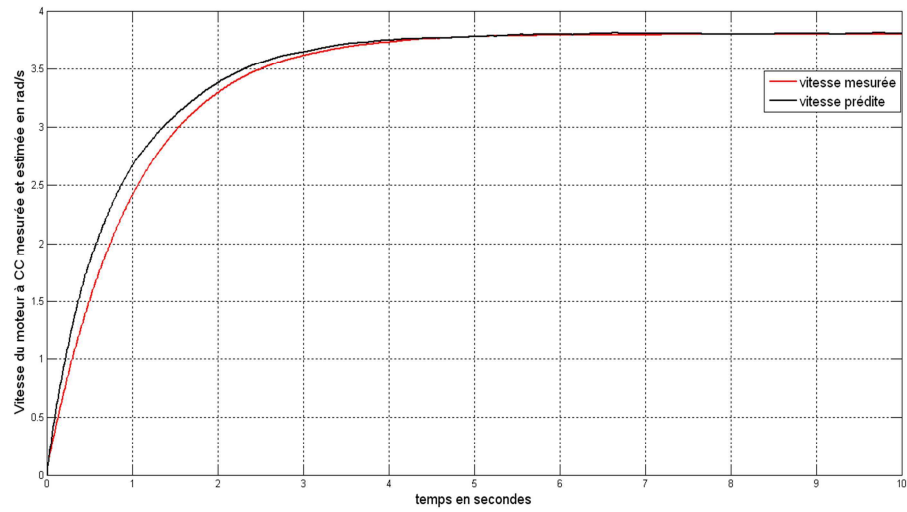


Figure 4.2 vitesse du moteur à CC mesurée et prédite par le modèle identifié.

Le modèle discret de $T_e = 10ms$ identifié par cette méthode est donné par :

$$G(z^{-1}) = \frac{0.0042z^{-1} + 0.0001831z^{-2}}{1 - 1.264z^{-1} + 0.2715z^{-2}} \Leftrightarrow G(z) = \frac{0.0042z + 0.0001831}{z^2 - 1.264z + 0.2715}$$

Références bibliographiques

Références Bibliographie

- [1] G. Cabodevila. Identification des systèmes : polycopie de cours. Ecole Nationale Supérieure de Mécanique et des Microtechniques, Besançon cedex- France. <http://intranet-tice.ens2m.fr>, Ensmm, France, 2009/2010.
- [2] C. Foulard, S.Gentil, and J.P. Sandraz. Commande et régulation par ordinateur numérique : de la théorie aux applications. Eyrolles, Paris, 1987.
- [3] Lennart Ljung. System identification (2nd ed.) : theory for the user. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1999.
- [4] Lennart Ljung and Torkel Glad. Modeling of dynamic systems. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1994.
- [5] Lennart Ljung. System identification toolbox for use with matlab, 1997.
- [6] Ioan D. Landau. Identification des systèmes : Hermès, 1998.
- [7] R. Ben Abdenmour , P. Borne, M. Ksouri & F. M'sahli. Identification et commande numérique des procédés industriels : Technip, 2001.
- [8] Eric Walter. Identification des modèles paramétriques : Masson, 1994.
- [9] P. Borne. Modélisation et Identification des Processus : Tome1 et Tome2, Technip, 1992.
- [10] Landau, I. D. and Zito, G. (2005). Digital Control Systems: Design, Identification and Implementation. Springer.
- [11] Albertos, P., Valera, A., Romero, J., and Esparza, A. (2005). An application of iterative identification and control in the robotics field. Current Trends in Nonlinear Systems and Control, pp. 217–229. Birkhäuser.
- [12] P.C. Young. The refined instrumental variable method: unified estimation of discrete and continuous-time transfer function models. Journal Européen des Systèmes Automatisés, 2008.
- [13] T. Ribarits, M. Deistler, and B. Hanzon. On new parametrization methods for the estimation of linear state-space models. International Journal of Adaptive Control and Signal Processing, 18(9-10):717–743, 2004.
- [14] T.I. Salsbury. Continuous-time model identification for closed loop control performance assessment. Control Engineering Practice, 15:109–121, 2007.