



Université de Ghardaïa



Faculté des Sciences et de la Technologie

Département des Sciences et des Techniques

Polycopie du Cours :
Modélisation et Identification des
Systemes Dynamiques

Filière : Automatique

Spécialité : Automatique

Préparé par :

- BECHOUAT Mohcene

Année Universitaire : 2017-2018

Table des matières

Avant-propos	v
Introduction	1
1 Modélisation des Systèmes dynamiques	2
1.1 Introduction	2
1.2 Systèmes dynamiques	2
1.3 Représentation de la fonction de transfert	3
1.4 Propriétés de la transformée de Laplace	4
1.4.1 La dérivée	4
1.4.2 L'intégrale	5
1.4.3 Le retard	5
1.5 Représentation de l'espace d'état	5
1.6 Modélisation des systèmes mécaniques	6
1.7 Modélisation des systèmes électriques	9
2 Rappels des méthodes de base en Automatique	12
2.1 Introduction	12
2.2 Réponse temporelle d'un système	12
2.2.1 Réponse impulsionnelle	12
2.2.2 Réponse à un échelon unitaire ou réponse indicielle	13
2.3 Identification d'un système à partir de sa réponse indicielle	13
2.3.1 Système du 1 ^{er} ordre	13
2.3.2 Système du 2 ^{ème} ordre	15

2.3.2.1	Système apériodique	15
2.3.2.2	Système pseudo-périodique	16
2.4	Identification d'un système à partir d'une réponse fréquentielle	17
3	Principe d'ajustement du modèle	19
3.1	Identification des modèles paramétriques	19
3.1.1	Modèle linéaire par rapport aux paramètres	20
3.1.1.1	Modèle hypothèse ARX (AutoRégressif à variables eXo- gènes)	21
3.1.1.2	Modèle hypothèse ARMAX (AutoRégressif à Moyenne Ajustée et à variables eXogènes)	21
3.2	Minimisation du critère d'ajustement et calcul de la solution optimale	22
3.2.1	Méthode basée sur l'erreur de sortie	22
3.2.2	Méthode basée sur l'erreur de prédiction	23
3.3	Forme matricielle de la méthode des moindres-carrés	24
4	Analyse de la méthode des moindres-carrés	26
4.1	Biais d'estimation	26
4.2	Variance de l'estimation	27
4.3	Estimateur du maximum de vraisemblance	27
4.4	Rejet des mesures aberrantes	28
5	Moindres-carrés récursifs	30
5.1	Principe du calcul récursif	30
5.2	Mise en œuvre de la méthode récursive	31
5.3	Facteur de pondération	32
5.4	Facteur d'oubli	33
	TRAVAUX PRATIQUES	41
TP1	Prise en main de SIMULINK	43
TP2	Simulation d'un système décrit par l'équation d'état	45
TP3	Identification non paramétrique par La méthode de Strejc	47
TP4	Identification non paramétrique par La méthode de Broida	50

TP5 : Identification paramétrique par La méthode de moindres carrés	52
Bibliographie et Webographie	56

Avant-propos

Ce polycopie a pour objectif de sensibiliser les étudiants aussi bien à la question de la modélisation ainsi qu'à l'identification des systèmes dynamiques linéaires en leur donnant tous les détails nécessaires, à savoir : les lois fondamentales de la modélisation et de l'identification, la présentation de notions fondamentales ainsi que des méthodes de base qui permettent à un automaticien de développer des modèles de représentation décrivant le comportement entrée-sortie d'un processus, quel qu'il soit, qu'il faut commander dans le but de mettre au point un contrôleur performant. A la fin du cours, les étudiants seront capables de modéliser, d'analyser et de concevoir des systèmes dynamiques linéaires.

De nombreux travaux pratiques sont introduits dans ce polycopié, conformément aux programmes de 3^{ème} année licence LMD Automatique, afin de permettre aux étudiants d'analyser et de comprendre les phénomènes qu'ils sont appelés à rencontrer dans ce domaine.

Introduction

La première étape du processus de conception d'un contrôle consiste à développer des modèles mathématiques appropriés du système à contrôler. Ces modèles peuvent être dérivés soit des lois physiques soit des données expérimentales.

Un modèle est un produit (physique ou numérique) qui représente un système d'intérêt. C'est une représentation simplifiée du système réel qu'il représente et qui s'approche, le plus possible, de la plupart des caractéristiques importantes de ce système. Un bon modèle est un compromis judicieux entre réalisme et simplicité. En effet, une caractéristique-clé d'un modèle est la manipulabilité. Par ailleurs, un modèle peut être soit un modèle physique soit un modèle conceptuel. A titre d'exemple, le modèle d'architecture d'une maison, celui d'un avion ou d'un mannequin de mode ou encore celui d'un organisme en recherche biologique constituent des modèles physiques. Par contre, un modèle informatique, un modèle statistique ou mathématique, ou encore un modèle économique constituent des modèles conceptuels.

L'identification est la sélection d'un modèle pour un processus (c'est-à-dire le système étudié ou le dispositif testé) tout en veillant à utiliser un nombre de mesures des entrées et des sorties aussi limitées que possible, qui peuvent être perturbées volontairement par le bruit, et une connaissance *a priori* du système.

Dans ce cours, les techniques les plus connues de la modélisation ou l'identification sont citées et détaillées d'une manière facile et sont ordonnées de manière à répondre aux questions suivantes :

- Qu'est ce qu'un système dynamique ?
- Qu'est ce que la modélisation ?
- Comment modélise-t-on un système dynamique linéaire ?
- Qu'est ce que l'identification ?
- Comment identifie-t-on un système dynamique linéaire ?

Chapitre 1

Modélisation des Systèmes dynamiques

1.1 Introduction

Dans ce chapitre, nous introduisons la représentation des systèmes dynamiques par la fonction de transfert et par l'espace d'état. Nous passons ensuite en revue quelques approches de base de la modélisation des systèmes mécaniques et électriques et nous montrerons comment exploiter ces modèles dans *MATLAB* pour une analyse plus poussée.

1.2 Systèmes dynamiques

Les systèmes dynamiques sont des systèmes qui changent ou évoluent dans le temps selon une règle fixe. Pour de nombreux systèmes physiques, cette règle peut être définie comme un ensemble d'équations différentielles du premier ordre :

$$\dot{x} = f(x(t), u(t), t) \quad (1.1)$$

$x(t)$: est le vecteur d'état.

$u(t)$: est le vecteur des entrées.

f : est une fonction, éventuellement non linéaire, donnant la dérivée temporelle \dot{x} (taux de changement) du vecteur d'état pour un état et une entrée pendant un laps de temps défini.

Exemple :

Dans un système mécanique simple qui fait appel à l'ensemble masse-ressort-amortisseur, les variables d'état pourraient être la position et la vitesse de la masse. Le vecteur des entrées représente les "forces" appliquées extérieurement sur le système.

1.3 Représentation de la fonction de transfert

Les systèmes dynamiques linéaires ont une propriété extrêmement importante. De ce point de vue, si l'entrée d'un système est de type sinusoïdal avec une fréquence f_c alors la sortie sera également sinusoïdale avec la même fréquence f_c , mais avec une amplitude et une phase probablement différentes. Ces différences de magnitude et de phase sont fonctions de la fréquence et capturent ce que l'on appelle la réponse en fréquence du système.

En utilisant la transformation de Laplace, il est possible de convertir la représentation du domaine temporel d'un système en une représentation d'entrée/sortie dans le domaine fréquentiel, appelée fonction de transfert. Ce faisant, elle transforme également l'équation différentielle gouvernante en une équation algébrique souvent plus facile à analyser.

La transformée de Laplace d'une fonction de domaine temporel $f(t)$ est définie ci-dessous :

$$F(P) = \int_0^{+\infty} f(t) \cdot e^{-P \cdot t} \cdot dt \quad (1.2)$$

Où le paramètre $P = \delta + \omega \cdot j$ est une variable de fréquence complexe. Il est très rare, en pratique, d'évaluer directement une transformation de Laplace (bien que vous deviez certainement savoir comment le faire). Il est beaucoup plus commun de rechercher la transformation d'une fonction temporelle dans le **tableau de Laplace**, (Tableau 1.1) qui reproduit les fonctions les plus usuelles dans le domaine d'asservissement.

$f(t)$	$F(P)$
<i>Impulsion</i> = $\delta(t)$	1
<i>Échelon</i> = E	$\frac{E}{P}$
t^n ($n = 1$ s'appelle rampe)	$\frac{n!}{P^{(n+1)}}$
$e^{-a \cdot t}$	$\frac{1}{P+a}$
$\sin(\omega \cdot t)$	$\frac{\omega}{P^2+\omega^2}$
$\cos(\omega \cdot t)$	$\frac{P}{P^2+\omega^2}$

TAB. 1.1 – Laplace et Laplace inverse des fonctions les plus usuelles.

1.4 Propriétés de la transformée de Laplace

1.4.1 La dérivée

La transformée de Laplace de la dérivée $n^{\text{ième}}$ d'une fonction est particulièrement importante :

$$L \left\{ \frac{d^n f}{dt^n} \right\} = P^n \cdot F(P) - P^{(n-1)} \cdot f(0^+) - P^{(n-2)} \cdot f'(0^+) - \dots - f^{(n-1)}(0^+) \quad (1.3)$$

Exemple :

Les méthodes de domaine fréquentiel sont le plus souvent utilisées pour analyser des systèmes *SISO* (*single-input / single-output*), par exemple les systèmes régis par une équation différentielle à coefficients constants, comme indiqué ci-dessous :

$$a_n \cdot \frac{d^n y(t)}{dt^n} + \dots + a_0 \cdot \frac{dy(t)}{dt} = b_n \cdot \frac{d^n u(t)}{dt^n} + \dots + b_0 \cdot \frac{du(t)}{dt} \quad (1.4)$$

La transformée de Laplace de cette équation est donnée ci-dessous :

$$a_n \cdot P_n \cdot Y(P) + \dots + a_0 \cdot Y(P) = b_n \cdot P_n \cdot U(P) + \dots + b_0 \cdot U(P) \quad (1.5)$$

Où $Y(P)$ et $U(P)$ sont respectivement les transformées de Laplace de $y(t)$ et $u(t)$. Notez que lorsque nous trouvons des fonctions de transfert, nous supposons toujours que chacune des conditions initiales $y^n(0^+), \dots, y(0^+), u^n(0^+), \dots, u(0^+)$ sont nulles.

La fonction de transfert $H(P)$ d'un système linéaire qui est le rapport entre la transformée de Laplace de sa sortie $Y(P)$ et celle de son entrée $U(P)$ s'écrit :

$$H(P) = \frac{Y(P)}{U(P)} = \frac{b_n \cdot P_n + \dots + b_0}{a_n \cdot P_n + \dots + a_0} \quad (1.6)$$

1.4.2 L'intégrale

La transformée de Laplace de l'intégrale d'une fonction est donnée par :

$$L \left[\int f(t) \cdot dt \right] = \frac{F(P)}{P} + \dots + \frac{1}{P} \left[\int f(t) \cdot dt \right]_{t=0} \quad (1.7)$$

1.4.3 Le retard

La transformée de Laplace d'une fonction avec un retard τ est donnée par :

$$L[f(t - \tau)] = F(P) \cdot e^{-\tau \cdot P} \quad (1.8)$$

1.5 Représentation de l'espace d'état

La représentation d'un comportement réel d'un système par l'espace d'état est donnée ci-dessous :

$$\begin{cases} \dot{X} = A \cdot X + B \cdot U \\ Y = C \cdot X + D \cdot U \end{cases} \quad (1.9)$$

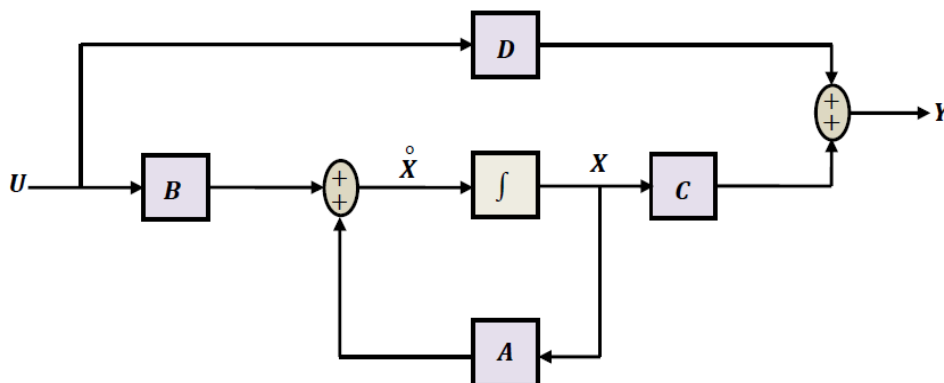


FIG. 1.1 – Représentation d'état matricielle.

Où X est le vecteur des variables d'état de dimension $(n \times 1)$, \dot{X} est la dérivée temporelle du vecteur d'état de dimension $(n \times 1)$. U est le vecteur de commande de dimension $(m \times 1)$, Y est le vecteur de sortie de dimension $(q \times 1)$, A est la matrice d'évolution de dimension $(n \times n)$, B est la matrice de commande de dimension $(n \times m)$, C est la matrice d'observation de dimension $(q \times n)$, D est la matrice d'action directe de dimension $(q \times m)$.

L'équation de sortie (Équation 1.9) est nécessaire car il y a souvent des variables d'état qui sont inaccessibles ou qui ne présentent pas un intérêt évident. La matrice de sortie C est utilisée pour spécifier les variables d'état (ou combinaisons de variables) qui sont disponibles pour l'utilisation par le contrôleur. De plus, il est souvent préférable d'avoir l'absence de l'action directe à laquelle les éléments de la matrice D doivent être forcés par zéro.

La représentation par l'espace d'état, également appelée représentation dans le domaine temporel, peut facilement prendre en charge des systèmes *MIMO* (*Multi-Input/Multi-Output*), des systèmes avec des conditions initiales non nulles et des systèmes non linéaires via l'Équation 1.1. Par conséquent, la représentation de l'espace d'état est largement utilisée dans la théorie du contrôle «moderne».

Observation :

Notons que nous pouvons également déterminer directement la fonction de transfert à partir de la représentation de l'espace d'état comme suit :

$$G(P) = \frac{Y(P)}{U(P)} = C \cdot (P \cdot I - A)^{-1} \cdot B + D \quad (1.10)$$

Où :

$I(n \times n)$ est la matrice d'identité.

1.6 Modélisation des systèmes mécaniques

Les lois du mouvement de Newton forment la base de l'analyse des systèmes mécaniques. La deuxième loi de Newton, (Équation 1.11), stipule que la somme des forces F agissant sur un corps est égale au produit de sa masse m par son accélération a . La troisième loi de Newton stipule que si deux corps sont connectés, alors ils éprouveraient des forces de même amplitude mais agissant dans des directions opposées.

$$\sum F = m \cdot a = m \cdot \frac{d^2x}{dt^2} \quad (1.11)$$

En appliquant cette équation, il est préférable de construire un diagramme de corps libre (en anglais “**Free Body Diagram (FBD)**”) du système montrant toutes les forces appliquées.

Exemple :

Système Masse-Ressort-Amortisseur :

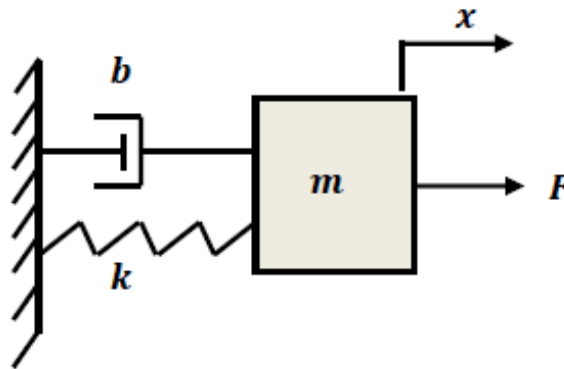


FIG. 1.2 – Système Masse-Ressort-Amortisseur.

Le diagramme de corps libre pour ce système est montré dans la Figure 1.3.

La force du ressort :

$f_k = k \cdot x$ (où « k : une constante de raideur ») est proportionnelle au déplacement x de la masse m .

Parallèlement, la force d’amortissement :

$f_b = b \cdot \dot{x}$ (où « b : le coefficient d’amortissement visqueux») est proportionnelle à la vitesse $v = \dot{x}$ de la masse m .

Les deux forces s’opposent au mouvement de la masse m et sont donc représentées dans la direction négative de x et F . Il faut aussi remarquer ici que pour $x = 0$, cela correspond à la position de la masse lorsque le ressort n’est pas étiré.

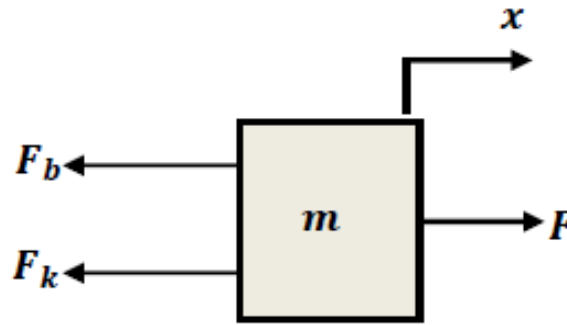


FIG. 1.3 – Diagramme de corps libre.

En additionnant les forces et en appliquant la seconde loi de Newton ([Équation 1.11](#)) selon toutes les directions, (dans ce cas, il n’y a pas de forces fonctionnant dans la direction Y ; elles existent seulement selon la direction des X) nous obtenons :

$$\sum F_x = F(t) - k \cdot x - b \cdot \dot{x} = m \cdot \ddot{x} \quad (1.12)$$

Cette équation ([Équation 1.12](#)), connue sous le nom d’équation dominante, caractérise complètement l’état dynamique du système. Plus tard, nous verrons comment l’utiliser pour calculer la réponse du système pour toute entrée externe $F(t)$, ainsi que pour analyser les propriétés du système telles que la stabilité et la performance.

Pour déterminer la représentation d’état du système masse-ressort-amortisseur, nous devons réduire l’équation régissant le second ordre à un ensemble de deux équations différentielles du premier ordre. À cette fin, nous choisissons la position et la vitesse comme variables d’état.

$$X = \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \quad (1.13)$$

Notons également que ces variables d’état correspondent respectivement à l’énergie potentielle et l’énergie cinétique de la masse. Souvent, lors de la modification de variables d’état, il est utile de considérer les éléments de stockage d’énergie indépendants dans le système.

Alors, pour trouver l’espace d’état de ce système, On pose que :

$$\dot{x} = \dot{x} \quad (1.14)$$

La dérivée de l'Équation 1.14 donne l'Équation 1.15, elle est simplifiée et reformée dans l'Équation 1.16 :

$$\dot{x} = \ddot{x} \tag{1.15}$$

$$\ddot{x} = -\frac{k}{m} \cdot x - \frac{b}{m} \cdot \dot{x} + \frac{1}{m} \cdot F(t) \tag{1.16}$$

Donc, On peut reformer l'Équation 1.14 et l'Équation 1.16 en espace d'état par l'équation suivante :

$$\dot{X} = \begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{b}{m} \end{bmatrix} \cdot \begin{bmatrix} x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} \cdot F(t) \tag{1.17}$$

Si, par exemple, nous voulons contrôler la position de la masse, l'équation de sortie serait la suivante :

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \tag{1.18}$$

Pour déterminer la fonction de transfert du système masse-ressort- amortisseur, on applique la transformée de Laplace à l'équation (l'Équation 1.12) avec des conditions initiales nulles, on aura :

$$m \cdot P^2 \cdot X(P) + b \cdot P \cdot X(P) + k \cdot X(P) = F(P) \tag{1.19}$$

Par conséquent, la fonction de transfert de l'entrée de force excitatrice à la sortie de déplacement est :

$$H(P) = \frac{X(P)}{F(P)} = \frac{1}{m \cdot P^2 + b \cdot P + k} \tag{1.20}$$

1.7 Modélisation des systèmes électriques

Comme pour les lois de Newton dans les systèmes mécaniques, les lois de Kirchoff sont considérées comme un outil analytique de base dans les systèmes électriques. La loi des nœuds stipule que dans un circuit, la somme des courants électriques entrants dans un

noeud est égale à la somme des courants électriques sortants de ce noeud. De même, la loi des mailles stipule que dans une maille, la somme algébrique des chutes de tensions aux bornes de toutes les résistances est nulle.

Exemple :

Circuit RLC :

Nous allons maintenant considérer un circuit contenant trois éléments électriques passifs en série : une résistance, une inductance et un condensateur. Ce circuit est communément connu sous le nom de circuit RLC.

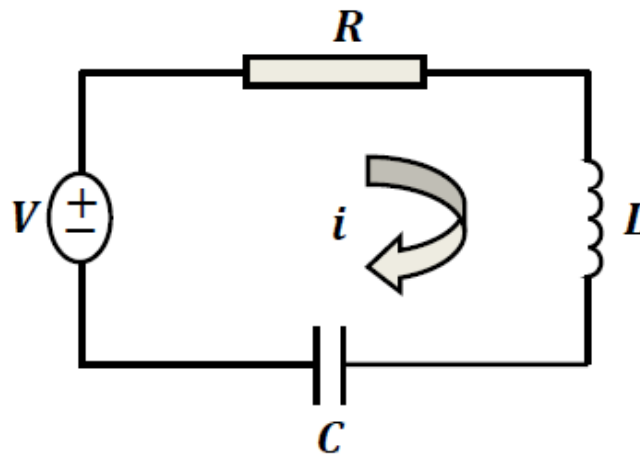


FIG. 1.4 – Circuit RLC.

Puisque ce circuit est constitué d'une seule maille, le courant $i(t)$ est le même à travers tout le circuit. En appliquant la loi des mailles à ce circuit et en tenant compte de la convention des signes indiquées dans la [Figure 1.4](#), nous arrivons à l'[Équation 1.21](#) de décision suivante :

$$V(t) - L \cdot \frac{di}{dt} - R \cdot i - \frac{1}{C} \int i \cdot dt = 0 \quad (1.21)$$

Nous notons que l'[Équation 1.21](#) de décision pour le circuit RLC a une forme analogue à celle du système mécanique masse-ressort-amortisseur. En particulier, ils sont tous des systèmes du second ordre où la charge (intégrale du courant) correspond au déplacement, l'inductance à la masse, la résistance à l'amortissement visqueux et la capacité inverse à la raideur du ressort. Ces analogies ainsi que d'autres s'avèrent être un concept très utile pour comprendre le comportement des systèmes dynamiques.

La représentation d'état est obtenue en choisissant la charge et le courant comme variables d'état :

$$X = \begin{bmatrix} q \\ i = \frac{dq}{dt} \end{bmatrix} \quad (1.22)$$

L'équation d'état du système électrique se déduit de la même manière que pour l'espace d'état du système **masse –ressort –amortisseur** :

$$\dot{X} = \begin{bmatrix} \dot{q} \\ \dot{i} \end{bmatrix} = \begin{bmatrix} i \\ \dot{i} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{R}{L} & -\frac{1}{L \cdot C} \end{bmatrix} \cdot \begin{bmatrix} q \\ \dot{q} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} \cdot V(t) \quad (1.23)$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} \quad (1.24)$$

La représentation de la fonction de transfert peut être trouvée en considérant la transformée de Laplace à partir de l'équation d'espace d'état comme suit :

$$G(P) = \frac{I(P)}{V(P)} = \begin{bmatrix} 1 & 0 \end{bmatrix} \cdot \left(P \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ -\frac{R}{L} & -\frac{1}{L \cdot C} \end{bmatrix} \right)^{-1} \cdot \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} \quad (1.25)$$

alors :

$$G(P) = \frac{1}{L \cdot P^2 + R \cdot P + \frac{1}{C}} \quad (1.26)$$

Pour représenter les modèles dans l'environnement Matlab, on utilise la fonction **tf** pour déclarer le modèle comme une fonction de transfert (pour plus détails, taper **help tf**). De même, on utilise la fonction **ss** pour déclarer le modèle comme un espace d'état. Ainsi, les fonctions **ss2tf** et **tf2ss** donnent le passage entre les deux modèles.

Par exemple, le modèle précédent est déclaré par :

$$G = tf(1, [L, R, 1/C]) \quad (1.27)$$

Chapitre 2

Rappels des méthodes de base en Automatique

2.1 Introduction

Si le modèle de connaissance s'avère trop peu précis ou impossible à obtenir, on peut alors le définir à partir de ces données d'entrée - sortie. On parle dans ce chapitre de l'identification non paramétrique du système. L'objectif est d'obtenir un modèle qui "se comporte comme" le système. La plupart du temps, les paramètres de ce modèle n'ont qu'un rapport lointain avec les paramètres d'un modèle de connaissance. En premier lieu, il faut donc répondre aux deux questions suivantes :

- Quelles sont les grandeurs d'entrée-sortie ?
- Comment traiter la sortie ?

2.2 Réponse temporelle d'un système

2.2.1 Réponse impulsionnelle

Soumettons une impulsion à l'entrée d'un système : $e(t) = \delta(t)$, $L[\delta(t)] = E(P) = 1$, sa réponse, appelée réponse impulsionnelle, aura pour transformée de Laplace $S(P) = 1 \cdot H(P) = H(P)$. Donc, la fonction de transfert est obtenue via la transformée de Laplace du transfert $h(t)$.

2.2.2 Réponse à un échelon unitaire ou réponse indicielle

Soumettons le système à une entrée en échelon unitaire : $e(t) = 1$, $E(P) = \frac{1}{P}$, la transformée de Laplace de la réponse indicielle $q(t)$ est : $L[q(t)] = \frac{1}{P} \cdot H(P)$.

Donc :

- La réponse impulsionnelle est la dérivé de la réponse indicielle : $h(t) = \frac{dq(t)}{dt}$;
- Il est possible de caractériser un système par sa réponse unitaire $q(t)$ ou sa réponse impulsionnelle $h(t)$.

2.3 Identification d'un système à partir de sa réponse indicielle

2.3.1 Système du 1^{er} ordre

La réponse indicielle d'un système de premier ordre est donnée par :

$$s(t) = K \cdot \left(1 - e^{-\frac{t}{\tau}}\right) \tag{2.1}$$

L'allure de l'équation qui représente ce système est illustrée dans la figure suivante :

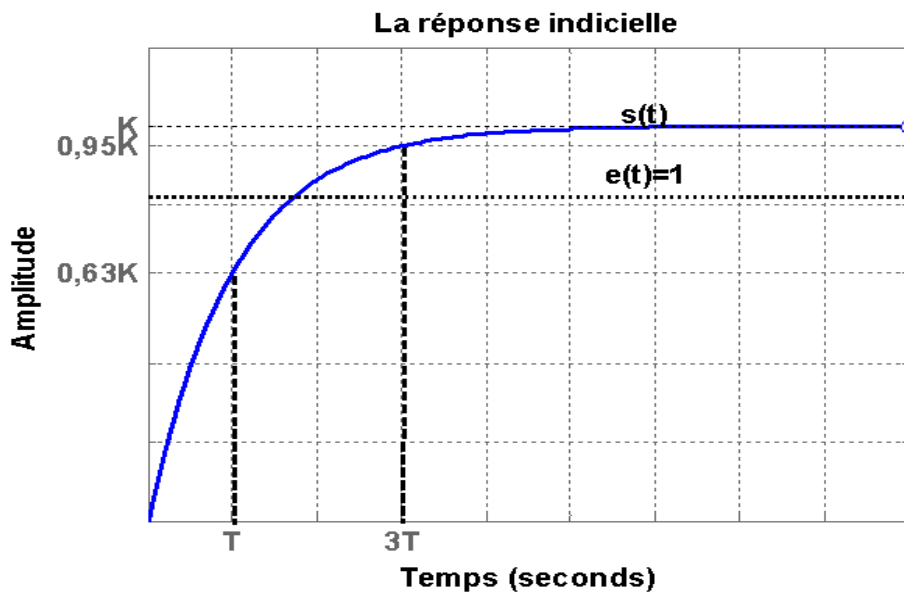


FIG. 2.1 – Réponse indicielle d'un système du 1^{er} ordre.

La connaissance de K et T est donc suffisante pour identifier la fonction :

- Le gain statique du système est calculé selon la formule suivante :

$$K = \lim_{t \rightarrow \infty} s(t) \quad (2.2)$$

- La constante de temps est calculée selon l'une des trois possibilités suivantes :

$$\begin{cases} s^\circ(0) = \frac{K}{T} \\ tr_{5\%} = 3 \cdot T \\ s(T) = 0.63 \cdot K \end{cases} \quad (2.3)$$

En cas de difficulté pour tracer la tangente à l'origine comme illustré dans la [Figure 2.2](#) on utilise :

Pour un système du 1^{er} ordre, la tangente à la courbe à un instant t_1 quelconque coupe l'asymptote de $s(t)$ en un point situé T plus loin, à cette fin la tangente est donnée par :

$$s^\circ(t) = \frac{K}{T} \cdot e^{-\frac{t}{T}} \quad (2.4)$$

L'équation de la tangente est de la forme :

$$y = \frac{K}{T} \cdot e^{-\frac{t_1}{T} \cdot t} + b \quad (2.5)$$

Au point de contact entre la tangente et la courbe, on peut écrire :

$$\frac{K}{T} \cdot e^{-\frac{t_1}{T} \cdot t_1} + b = K \cdot \left(1 - e^{-\frac{t_1}{T}}\right) \cdot b \quad (2.6)$$

D'où :

$$b = K \cdot \left[1 - \left(\frac{t_1 + T}{T}\right) \cdot e^{-\frac{t_1}{T}}\right] \quad (2.7)$$

L'équation de la tangente est donc :

$$y = \frac{K}{T} \cdot e^{-\frac{t_1}{T} \cdot t} + K \cdot \left[1 - \left(\frac{t_1 + T}{T}\right) \cdot e^{-\frac{t_1}{T}}\right] \quad (2.8)$$

L'intersection avec l'asymptote de $s(t)$ est obtenue pour $y = K$, alors :

$$t = t_1 + T \quad (2.9)$$

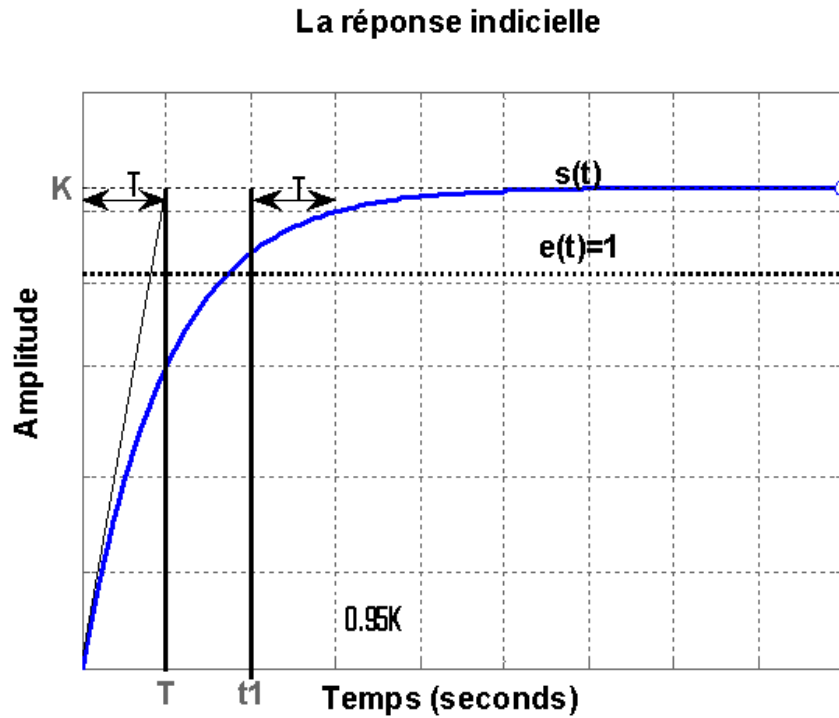


FIG. 2.2 – Réponse indicielle d'un système du 1^{er} ordre (deuxième version).

2.3.2 Système du 2^{ème} ordre

2.3.2.1 Système apériodique

Les notions permettant l'identification complète des systèmes du 2^{ème} ordre apériodique, présenté par l'allure dans la [Figure 2.3](#), dépasse le cadre de ce cours. Aussi, elle ne pourra donc être qu'approximative. La réponse indicielle est exprimée par :

$$s(t) = \frac{K}{T_1 - T_2} \cdot \left[T_1 \cdot \left(1 - e^{-\frac{t}{T_1}} \right) - T_2 \cdot \left(1 - e^{-\frac{t}{T_2}} \right) \right] \quad (2.10)$$

Le tracé obtenu résulte de la différence de deux fonctions dont leurs dérivées à l'origine sont égales :

$$s_1(t) = \frac{K}{T_1 - T_2} \cdot \left[T_1 \cdot \left(1 - e^{-\frac{t}{T_1}} \right) \right] \quad (2.11)$$

$$s_2(t) = \frac{K}{T_1 - T_2} \cdot \left[T_2 \cdot \left(1 - e^{-\frac{t}{T_2}} \right) \right] \quad (2.12)$$

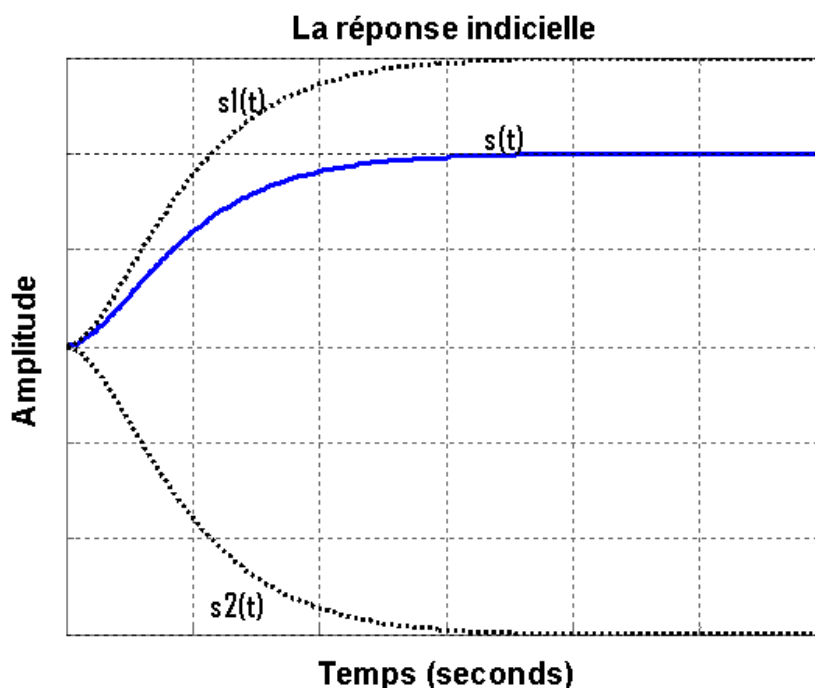


FIG. 2.3 – Réponse indicielle d'un système du 2^{ème} ordre aperiodique.

Sachant que $T_2 < T_1$, on peut considérer qu'au delà du point d'inflexion, $s_1(t)$ représente à peu près l'allure de $s(t)$ et on peut alors déterminer T_1 en utilisant la méthode décrite pour les systèmes du 1^{er} ordre avec difficulté pour tracer la tangente à l'origine. On détermine ensuite T_2 en exprimant $s(t)$ pour une valeur de t bien au delà de l'inflexion, alors $e^{-\frac{t}{T_2}}$ sera négligeable et $s(t)$ étant prise s'exprime par :

$$s(t) = K \cdot \left[T_1 \cdot \left(1 - \frac{T_1 \cdot e^{-\frac{t}{T_1}}}{T_1 - T_2} \right) \right] \quad (2.13)$$

2.3.2.2 Système pseudo-périodique

La réponse indicielle d'un système du 2^{ème} ordre pseudo-périodique est représentée dans la Figure 2.4, où E est connu ; K est une constante donnée par l'asymptote. Le premier dépassement permet de calculer z est donné par :

$$D_1 = s\left(\frac{T}{2}\right) = KE \cdot \left(1 + e^{-\frac{z \cdot \pi}{\sqrt{1-z^2}}} \right) - KE = KE \cdot e^{-\frac{z \cdot \pi}{\sqrt{1-z^2}}} \quad (2.14)$$

Sachant que la valeur $\frac{T}{2}$ est mesurée directement sur le graphe, en conséquence, le calcul de la pulsation ω_0 est assuré via l'expression ci-dessous :

$$\omega = \frac{2 \cdot \pi}{T} \implies \omega = \omega_0 \cdot \sqrt{1 - z^2} \implies \omega_0 = \frac{2 \cdot \pi}{T \cdot \sqrt{1 - z^2}} \quad (2.15)$$

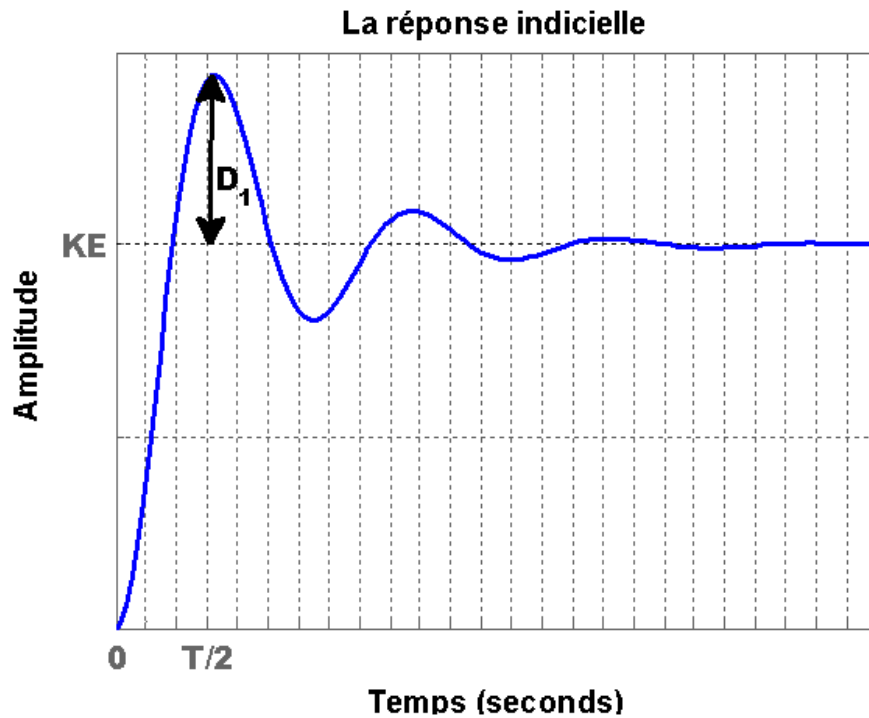


FIG. 2.4 – Réponse indicielle d’un système du 2^{ème} ordre pseudo apériodique.

2.4 Identification d’un système à partir d’une réponse fréquentielle

Dans ce cas, l’excitation utilisée est sinusoïdale définie par $u = A \sin(\omega t)$. De plus, ω qui balaye l’espace des pulsations susceptibles peut contenir une pulsation de coupure, notée par ω_c . En outre, l’amplitude et le déphasage de la sortie vis-à-vis de cette excitation permettent de tracer le diagramme de Bode et l’analyse du système dans le plan fréquentiel permet de déterminer le modèle mathématique désiré. Notant ici que les résultats obtenus sont souvent difficiles à exploiter si les constantes de temps sont proches avec une bonne excitation sur tout le spectre de fréquences. Généralement, l’identification à partir d’une réponse fréquentielle n’est pas une commande industrielle classique et par conséquent elle est difficile à mettre en œuvre.

Exemple d’un système du 1^{er} ordre :

La pulsation de coupure ω_c est obtenue graphiquement à partir du diagramme de phase pour un déphasage de $-\frac{\pi}{4}$. Dans ce cas, la constante de temps T est alors déterminée par l'inverse de cette pulsation et le gain statique K est obtenu à partir de la limite du gain de G_{dB} lorsque $\omega \rightarrow 0$, d'où $K = 10^{(G_{dB}/20)}$ et $T = \frac{1}{\omega_c}$.

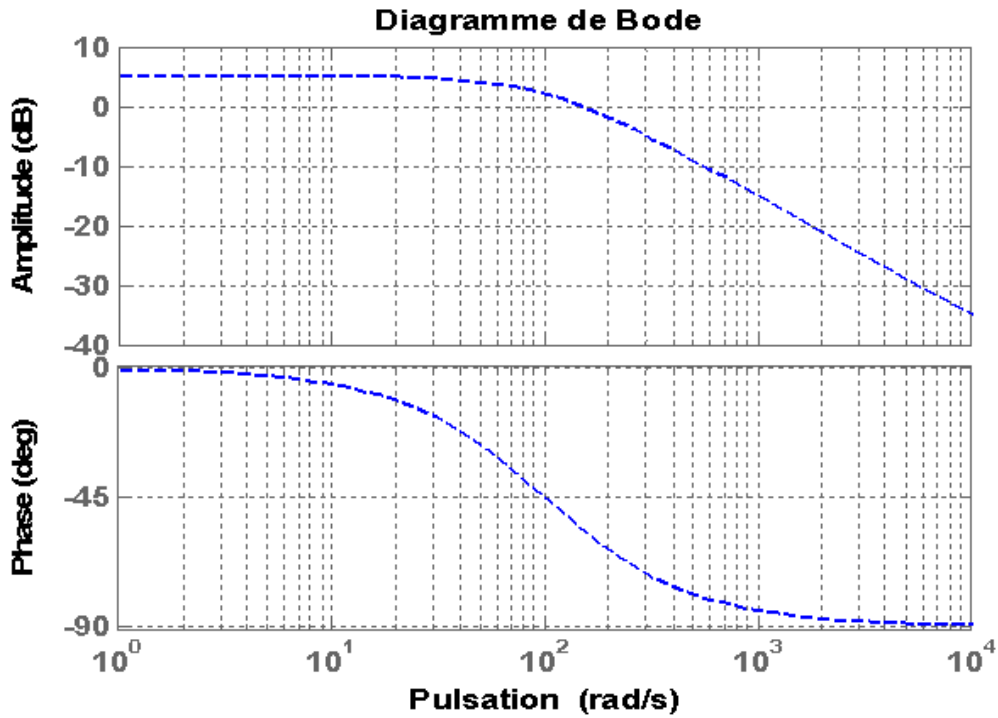


FIG. 2.5 – Diagramme de Bode d'un système du 1^{er} ordre .

Ainsi nous obtenons :

$$K = 10^{\frac{5}{20}} \approx 1.8 \text{ et } T = 0.01 \implies H(P) = \frac{1.8}{1 + 0.01 \cdot P} \quad (2.16)$$

Chapitre 3

Principe d'ajustement du modèle

3.1 Identification des modèles paramétriques

Ce chapitre considère la représentation paramétrique des systèmes dynamiques sur la base des mesures entrées-sorties. Les méthodes paramétriques présentent un avantage considérable quant au nombre de paramètres à identifier. Dans le cas d'une approche paramétrique, la structure du modèle doit être spécifiée à *priori*.

Dans ce chapitre, l'identification des paramètres du modèle se fera sur la base d'un critère d'écart entre des mesures expérimentales provenant du processus et une «simulation» des équations constituant le modèle (Figure 3.1). Pour cela, il faudra choisir au préalable :

- la structure du modèle (ou caractérisation du processus),
- le critère de performance,
- l'algorithme d'identification (c'est-à-dire la méthode de calcul des paramètres),
- Le signal d'excitation u .

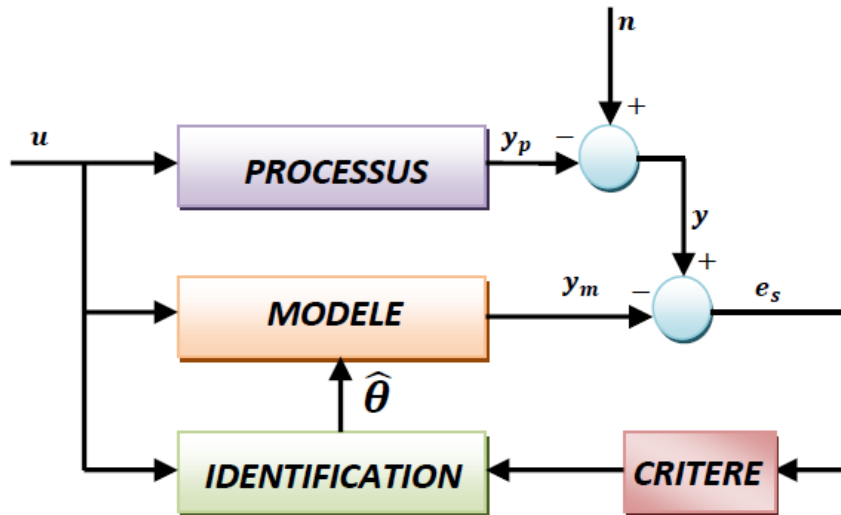


FIG. 3.1 – Schéma pour l'identification des paramètres du modèle.

Où :

u : Entrée(ajustable) ;

y_p : Sortie du processus(non mesurable) ;

n : bruit(inconnu) ;

y : Sortie bruitée(mesurée) ;

y_m : Sortie du modèle(calculable) ;

e_s : Erreur de sortie(calculable) ;

θ : Vecteur des paramètres estimés.

3.1.1 Modèle linéaire par rapport aux paramètres

On cherche à identifier les paramètres d'un modèle entrée-sortie linéaire (en temps discret) sous la forme générale :

$$y(k) = G(q) \cdot u(k) + H(q) \cdot n(k) \quad (3.1)$$

On fait généralement des hypothèses sur la structure du modèle, c'est-à-dire sur la forme des fonctions $G(q)$, $H(q)$. On parle alors de modèle hypothèse. Un modèle-hypothèse traduit le fait que l'on fait une hypothèse sur la structure du système (*ordre, retard, etc.*) et sur le type de bruit (*bruit de sortie, bruit de boucle, etc.*).

3.1.1.1 Modèle hypothèse ARX (AutoRégressif à variables eXogènes)

Ce modèle, appelé aussi “*Equation Error Model Structure*” en anglais, il est représenté souvent sous une forme plus compacte :

$$A(q) \cdot y(k) = B(q) \cdot u(k) + n(k) \quad (3.2)$$

$$A(q) = 1 + a_1 \cdot q^{-1} + a_2 \cdot q^{-2} + \dots + a_{na} \cdot q^{-na} \quad (3.3)$$

$$B(q) = b_1 \cdot q^{-1} + b_2 \cdot q^{-2} + \dots + b_{nb} \cdot q^{-nb} \quad (3.4)$$

Où :

$n(k)$: appelé généralement bruit de boucle.

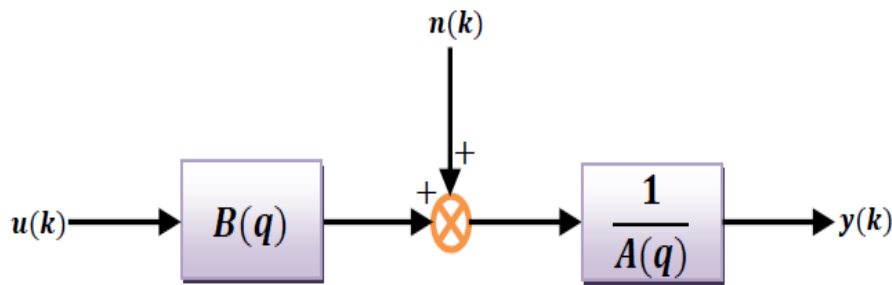


FIG. 3.2 – Modèle hypothèse ARX.

3.1.1.2 Modèle hypothèse ARMAX (AutoRégressif à Moyenne Ajustée et à variables eXogènes)

Ce modèle est souvent donné sous une forme plus compacte comme suit :

$$A(q) \cdot y(k) = B(q) \cdot u(k) + C(q) \cdot n(k) \quad (3.5)$$

Avec :

$$A(q) = 1 + a_1 \cdot q^{-1} + a_2 \cdot q^{-2} + \dots + a_{na} \cdot q^{-na} \quad (3.6)$$

$$B(q) = b_1 \cdot q^{-1} + b_2 \cdot q^{-2} + \dots + b_{nb} \cdot q^{-nb} \quad (3.7)$$

$$C(q) = c_1 \cdot q^{-1} + c_2 \cdot q^{-2} + \dots + c_{nc} \cdot q^{-nc} \quad (3.8)$$

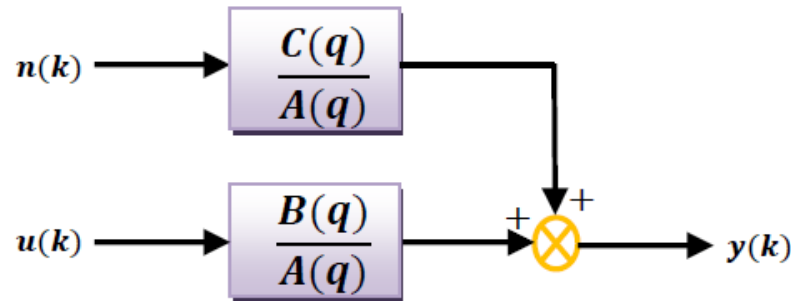


FIG. 3.3 – Modèle hypothèse ARMAX.

3.2 Minimisation du critère d'ajustement et calcul de la solution optimale

Le problème d'identification paramétrique peut être étudié dans un cadre plus général avec la méthode de l'erreur de sortie ou l'erreur de prédiction.

3.2.1 Méthode basée sur l'erreur de sortie

Le principe de cette méthode d'identification est illustré sur la [Figure 3.4](#). Le modèle est une fonction de n paramètres θ . Il s'agit alors de déterminer les paramètres θ tels que le critère soit minimum. Le critère est en général choisi sous forme d'une erreur quadratique moyenne donnée par :

$$J(\theta) = \frac{1}{N} \sum_{K=1}^N e^2(K) \quad (3.9)$$

La recherche des paramètres optimaux θ^Δ se fait par la résolution d'un problème d'optimisation non linéaire. Il s'agit donc de résoudre ce problème par un algorithme d'optimisation adéquat. Parmi les algorithmes les plus utilisés, on peut citer :

- Gradient
- Quasi Newton
- Nelder et Mead

- Algorithmes génétiques
- Recuit simulé

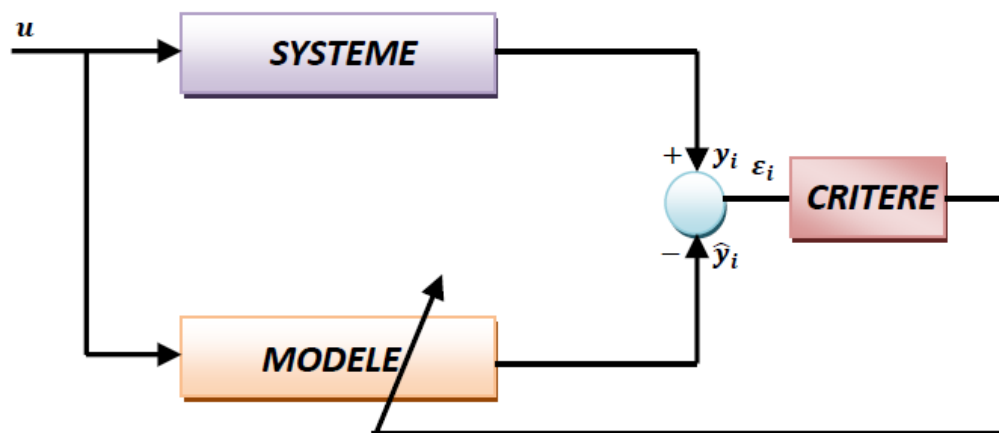


FIG. 3.4 – Principe d'identification fondée sur l'erreur de sortie.

3.2.2 Méthode basée sur l'erreur de prédiction

Dans ce cas, la sortie du modèle est assurée par un prédicteur optimal. Ce dernier a pour but de prédire la sortie prédite à l'instant k en fonction des données d'entrée-sortie réelles suivantes : $u(k-1), u(k-2), \dots, u(k-nb), y(k-1), y(k-2), \dots, y(k-na)$. Le principe de cette méthode peut être illustré par la [Figure 3.5](#) comme suit :

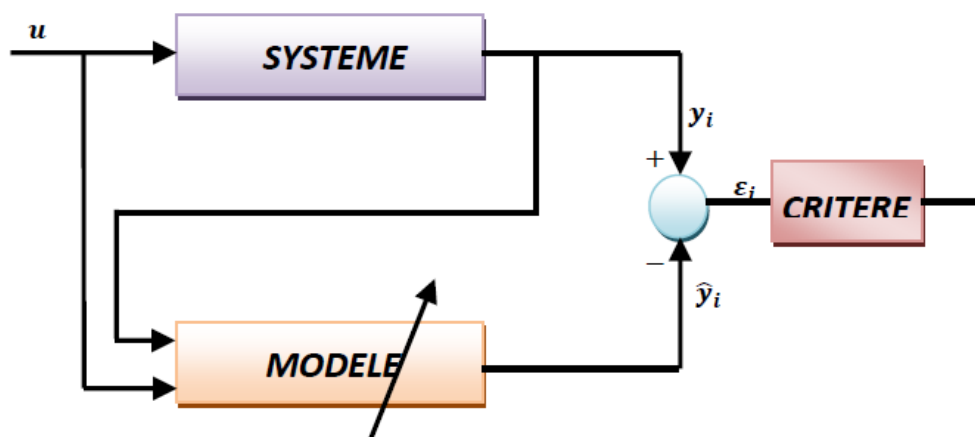


FIG. 3.5 – Principe d'identification fondée sur l'erreur de prédiction.

Cette méthode est basée sur les trois étapes suivantes :

- Choisir la structure du modèle du système. Le choix de la structure dépend des hypothèses sur l'ordre du modèle du processus et la nature du bruit.

- Définir le prédicteur de la sortie en relation avec la structure du modèle. Le prédicteur est une fonction des paramètres du modèle à identifier (θ inconnu) et des signaux d'entrée et sortie.
- Minimiser l'erreur de prédiction avec une méthode numérique. Après avoir défini le prédicteur pour la structure choisie, on obtient l'erreur de prédiction $e(k) = y(k) - \hat{y}(k)$ qui est une fonction du vecteur des paramètres à identifier θ . On obtient ainsi un problème d'optimisation numérique dont le critère à minimiser est la norme 2 de l'erreur de prédiction :

$$\hat{\theta} = \arg \min_{\theta} J(\theta) = \frac{1}{N} \sum_{K=1}^N e^2(K) \quad (3.10)$$

Exemple : identification paramétrique utilisant le modèle ARX

La détermination de ce modèle est assurée via la minimisation de l'erreur de prédiction définie comme suit :

$$e(k) = y(k) - \hat{y}(k) = y(k) - (B(q) \cdot u(k) + [1 - A(q)] \cdot y(k)) \quad (3.11)$$

Dans ce cas, le prédicteur $\hat{y}(k)$ est particulièrement simple à calculer, car il définit une régression linéaire que l'on écrit généralement sous la forme :

$$\hat{y}(k) = \Psi^T(k) \cdot \theta \quad (3.12)$$

Avec :

$$\Psi(k) = [-y(k-1) \dots -y(k-na) \ u(k-1) \dots -u(k-nb)]^T \quad (3.13)$$

$$\theta = [a_1 \dots a_{na} \ b_1 \dots b_{nb}]^T \quad (3.14)$$

3.3 Forme matricielle de la méthode des moindres-carrés

Nous recherchons le vecteur de paramètre θ qui minimise J . Etant donné que le critère est quadratique et qu'il est donné par :

$$J = (y - \Psi^T \cdot \theta) \cdot (y - \Psi^T \cdot \theta)^T = y \cdot y^T - \theta^T \cdot \Psi \cdot y - y^T \cdot \Psi^T \cdot \theta + \theta^T \cdot \Psi \cdot \Psi^T \cdot \theta \quad (3.15)$$

Dans ce cas, le vecteur optimal $\overset{\wedge}{\theta}$ est obtenu par la condition nécessaire d'optimalité, donnée par :

$$\frac{\partial J}{\partial \theta} /_{\theta=\overset{\wedge}{\theta}} = 0 \implies -2 \cdot \Psi \cdot y + 2 \cdot \Psi \cdot \Psi^T \cdot \theta = 0 \quad (3.16)$$

D'où l'on en déduit :

$$\overset{\wedge}{\theta} = (\Psi \cdot \Psi^T)^{-1} \cdot \Psi \cdot y \quad (3.17)$$

Il reste à vérifier que cet optimum représente le vecteur minimisant le critère précédent. Pour cela, on doit passer par la vérification de la condition suffisante qui représente ici la matrice Hessianne. Elle est donc définie par :

$$\frac{\partial^2 J}{\partial \theta^2} /_{\theta=\overset{\wedge}{\theta}} = 0 \implies 2 \cdot \Psi \cdot \Psi^T > 0 \quad (3.18)$$

$\overset{\wedge}{\theta}$ est un vecteur minimum si et seulement si la matrice Hessianne est définie positive.

Chapitre 4

Analyse de la méthode des moindres-carrés

Un estimateur est une valeur $\hat{\theta}$ calculée sur un échantillon tiré au hasard. Dans ce cas, $\hat{\theta}$ est une valeur aléatoire possédant une espérance $E[\hat{\theta}]$ et une variance $Var[\hat{\theta}]$. On comprend alors que sa valeur puisse fluctuer selon l'échantillon. Elle a de très faibles chances de coïncider exactement avec la valeur θ qu'elle est censée représenter. L'objectif est donc de maîtriser l'erreur commise en prenant la valeur $\hat{\theta}$ pour celle de θ .

4.1 Biais d'estimation

En statistique, le biais (*ou fonction de biais*) d'un estimateur est la différence entre la valeur attendue de cet estimateur et la valeur réelle du paramètre estimé. Un estimateur ou une règle de décision avec un biais nul est appelé non biaisé. Sinon, l'estimateur est dit biaisé.

Alors :

$$\text{Si } \hat{\theta} \text{ est l'estimateur de } \theta, \text{Biais}(\hat{\theta}) = E[\hat{\theta}] - \theta \quad (4.1)$$

L'erreur quadratique moyenne (**Mean Squared Error** *en anglais*) est l'espérance du carré de l'erreur entre la valeur vraie et sa valeur estimée :

$$MSE(\hat{\theta}) = E((\hat{\theta} - \theta)^2) \quad (4.2)$$

Rappels :

En théorie des probabilités, l'espérance mathématique d'une variable aléatoire réelle est, intuitivement, la valeur que l'on s'attend à trouver, en moyenne, si l'on répète un grand nombre de fois la même expérience aléatoire. Elle se note $E(X)$ et se lit "espérance de X ".

4.2 Variance de l'estimation

La variance mesure dans quelle mesure un ensemble de données est étalé. La définition technique s'énonce ainsi : "La moyenne des différences quadratiques par rapport à la moyenne", mais tout ce qu'elle fait est de vous donner une idée très générale de la propagation de vos données. Une valeur de zéro signifie qu'il n'y a pas de variable ; Tous les nombres dans l'ensemble de données sont les mêmes.

Soit X_1, X_2, \dots, X_n , on a n point de données (Variables aléatoires indépendantes et identiquement distribuées) d'une distribution F (*distribution Fisher-Snedecor*), l'estimateur $\hat{\theta}$ de θ est une fonction de X_1, X_2, \dots, X_n :

$$\hat{\theta}_n = g(X_1, X_2, \dots, X_n) \quad (4.3)$$

Alors, l'estimateur de la variance de l'échantillon de la loi gaussienne :

$$S^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - X_{moy})^2 \Rightarrow E(S^2) = \sigma^2 \quad (4.4)$$

Où :

$$X_{moy} = \frac{1}{n} \sum_{i=1}^n X_i \quad (4.5)$$

4.3 Estimateur du maximum de vraisemblance

Le principe du maximum de vraisemblance est relativement simple. Comme auparavant, nous commençons avec un échantillon $x = (x_1, x_2, \dots, x_n)$ de variables aléatoires choisies selon une famille de probabilités P_θ . En outre, $f(x/\theta)$, $x = (x_1, x_2, \dots, x_n)$ sera utilisé pour désigner la fonction de densité pour les données lorsque θ est le vrai état de la nature.

Ensuite, le principe de maximum de vraisemblance donne un choix de l'estimateur $\hat{\theta}$ rend les valeurs des paramètres comme des données observées les plus probables.

Alors, la fonction de vraisemblance est la fonction de densité considérée comme une fonction de θ :

$$l(\theta/x) = f(x/\theta) \quad (4.6)$$

L'estimateur du maximum de vraisemblance :

$$\hat{\theta} = \arg \max_{\theta} l(\theta/x) \quad (4.7)$$

Nous apprendrons que, surtout pour les grands échantillons, les estimateurs du maximum de vraisemblance ont de nombreuses propriétés souhaitables, notamment pour les grands échantillons. Cependant, en particulier pour les données de grande dimension, en particulier, la probabilité peut avoir de nombreux maxima locaux. Ainsi, trouver le global du maximum peut être un défi informatique majeur.

Cette classe d'estimateurs a une propriété importante. Si $\hat{\theta}(x)$ est une estimation du maximum de vraisemblance pour θ , alors $g(\hat{\theta}(x))$ est une estimation du maximum de vraisemblance pour $g(\theta)$. Par exemple, si θ est un paramètre pour la variance et $\hat{\theta}$ est le maximum estimateur de vraisemblance, alors $\sqrt{\hat{\theta}}$ est l'estimateur du maximum de vraisemblance pour l'écart-type. A noter ici que la flexibilité dans ce critère d'estimation n'est pas disponible dans le cas des estimateurs sans biais.

Typiquement, en maximisant la fonction de score, En $l(\theta/x)$, le logarithme de la vraisemblance, sera plus facile. L'existence des valeurs de paramètres en tant que variable d'intérêt peu inhabituelle.

4.4 Rejet des mesures aberrantes

Une valeur aberrante dans un ensemble de données peut être définie comme étant une observation (ou un ensemble d'observations) qui semble être inconsistante avec le reste des données. Dit d'une autre manière, il y a une valeur aberrante lorsque l'une ou l'autre parmi les observations d'un ensemble de données, détonne ou n'est pas en harmonie avec les autres observations. Ce qui caractérise la valeur aberrante, c'est son impact sur

l'observateur.

Soit x_1, x_2, \dots, x_n , un échantillon aléatoire uni varié de taille n , provenant d'une distribution F , et soit $x(1), x(2), \dots, x(n)$ les données ordonnées dans l'ordre croissant. Les valeurs $x(1)$ et $x(n)$ sont respectivement, par définition, l'observation extrême inférieure et supérieure. Le fait de déclarer qu'une observation extrême est une valeur aberrante dépend de la manière avec laquelle elle apparaît en fonction du modèle F .

Par exemple :

Dans la [Figure 4.1\(a\)](#), ni la valeur $x(1)$, ni $x(n)$ ne semble correspondre à une valeur aberrante. Par contre, dans la [Figure 4.1\(b\)](#), $x(n)$ est une valeur aberrante supérieure ou située au niveau de la queue droite de la distribution.

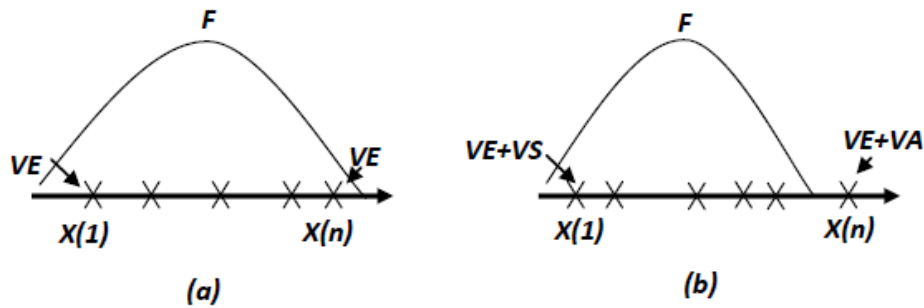


FIG. 4.1 – Exemple de valeurs aberrantes.

Chapitre 5

Moindres-carrés récurrents

5.1 Principe du calcul récursif

L'estimation des paramètres par la méthode des moindres carrés simples présente un inconvénient majeur, à savoir la nécessité de calculer l'inverse d'une matrice, ce qui est long et parfois impossible sur un microcontrôleur.

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 variants 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.

Notons, qu'en vue d'une implantation sur ordinateur, l'Équation 3.17 peut être transformée sous forme récurrente. En considérant les mesures jusqu'à l'instant k et avec les notations suivantes similaires à celles de l'Équation 5.1 :

$$e_k = \begin{bmatrix} e(1) \\ \vdots \\ e(k) \end{bmatrix} \quad y_k = \begin{bmatrix} y(1) \\ \vdots \\ y(k) \end{bmatrix} \quad \Psi_k = \begin{bmatrix} \Psi(1) \\ \vdots \\ \Psi(k) \end{bmatrix} \quad (5.1)$$

L'Équation 3.17 donne :

$$\overset{\Lambda}{\theta}_k = P_k \cdot \sum_{i=1}^k \Psi^T(i) \cdot y(i) \quad (5.2)$$

$$P_k = \left[\sum_{i=1}^k \Psi^T(i) \cdot \Psi(i) \right]^{-1} \quad (5.3)$$

En général, le vecteur de paramètres à l'instant $k + 1$ peut s'écrire :

$$\overset{\Lambda}{\theta}_{k+1} = \overset{\Lambda}{\theta}_k + P_{k+1} \cdot \Psi^T(k+1) \cdot \left[y(k+1) - \Psi(k+1) \cdot \overset{\Lambda}{\theta}_k \right] \quad (5.4)$$

$$P_{k+1}^{-1} = P_k^{-1} + \Psi^T(k+1) \cdot \Psi(k+1) \quad (5.5)$$

5.2 Mise en œuvre de la méthode récursive

Pour éviter l'inversion de la matrice P_{k+1} à chaque itération, on peut utiliser le lemme d'inversion matricielle. Selon ce lemme soient A , C et $C^{-1} + D \cdot A^{-1} \cdot B$ des matrices inversibles, alors :

$$(A + B \cdot C \cdot D)^{-1} = A^{-1} - A^{-1} \cdot B \cdot [C^{-1} + D \cdot A^{-1} \cdot B]^{-1} \cdot D \cdot A^{-1} \quad (5.6)$$

Prenons $A = P_k^{-1}$, $B = \Psi^T(k+1)$, $C = 1$ et $D = \Psi(k+1)$, alors on obtient :

$$P_{k+1} = P_k - \frac{P_k \cdot \Psi^T(k+1) \cdot \Psi(k+1) \cdot P_k}{1 + \Psi(k+1) \cdot P_k \cdot \Psi^T(k+1)} \quad (5.7)$$

Cette équation a l'avantage de ne pas nécessiter l'inversion d'une matrice, mais uniquement celle d'un scalaire.

Il existe deux façons d'initialiser la récurrence :

- des valeurs initiales sont fixées, généralement $\overset{\Lambda}{\theta}_k = 0$ et $P_0 = \alpha \cdot I$ où α représente un très grand scalaire (par exemple $\alpha = 10000$) et I la matrice unité,
- la récurrence commence à l'itération $p + 1$ (si l'on estime p paramètres) avec :

$$\overset{\Lambda}{\theta}_k = \Psi_p \cdot y_p \text{ et } P_p = [\Psi_p \cdot \Psi_p^T]^{-1} \quad (5.8)$$

Il est utile de mentionner qu'il existe d'autres formules récurrentes, équivalentes à celles

présentées ci-dessus, mais qui sont mieux adaptées pour les calculs numériques (équations “*stabilisées*”, équations “*factorisées*”). Elles sont cependant moins didactiques et ne sont pas données ici.

Par rapport à l’algorithme batch (ou par paquet) des moindres carrés donné à l’Équation 3.17, l’identification récurrente offre les avantages suivants :

- estimation du modèle en temps réel,
- compression importante des données, car l’algorithme récurrent traite à chaque instant une seule paire entrée/sortie au lieu de l’ensemble des données,
- exigence mémoire et puissance de calcul plus faibles,
- implantation aisée sur microprocesseur.

5.3 Facteur de pondération

On rappelle que l’équation de l’erreur de prédiction est donnée par :

$$e = y - \Psi^T \cdot \theta \quad (5.9)$$

Cette équation matricielle comporte N lignes. Toutes les erreurs de prédiction ne doivent pas nécessairement avoir la même importance dans le critère quadratique. Il est possible de les pondérer, par exemple, comme suit :

$$e_W = W \cdot [y - \Psi^T \cdot \theta] \quad (5.10)$$

W est une matrice de pondération de dimensions $(N \times N)$.

Les éléments de e_W se calculent à partir des éléments de W et de e de la manière suivante :

$$e_W(i) = \sum_{j=1}^n W_{ij} \cdot e(j) \quad , i = 1, \dots, N \quad (5.11)$$

Après minimisation du critère quadratique, le vecteur des paramètres estimés de l’Équation 3.17 devient alors :

$$\hat{\theta} = (\Psi \cdot W^T \cdot W \cdot \Psi^T)^{-1} \cdot \Psi \cdot W^T \cdot W \cdot \Psi^T \quad (5.12)$$

5.4 Facteur d'oubli

Un critère de choix indique que des mesures plus anciennes possèdent une influence réduite dans le calcul des paramètres actuels. Pour cette raison, l'influence des mesures plus anciennes est artificiellement réduite par l'introduction du facteur d'oubli λ , par exemple :

$$W^T \cdot W = \begin{bmatrix} \lambda^{N-1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \lambda^0 \end{bmatrix} \text{ avec } : \lambda \leq 1 \text{ (souvent } : 0 \leq \lambda \leq 0.99) \quad (5.13)$$

À partir de l'Équation 5.12 et l'Équation 5.13 on obtient :

$$\overset{\Lambda}{\theta}_k = P_k \cdot \sum_{i=1}^k \Psi^T(i) \cdot \lambda^{k-1} \cdot y(i) \quad (5.14)$$

La matrice P_{k+1} se calcule de façon récurrente comme suit :

$$P_{k+1}^{-1} = \lambda \cdot P_k^{-1} + \Psi^T(k+1) \cdot \Psi(k+1) \quad (5.15)$$

L'introduction d'un facteur d'oubli permet d'éviter une diminution trop rapide des éléments de la matrice P_{k+1} ce qui résulterait en un gain $P_{k+1} \cdot \Psi^T(k+1)$ faible et ainsi en une correction trop petite de $\overset{\Lambda}{\theta}_k$. En suivant la même démarche pour l'algorithme des moindres carrés récurrente et le lemme d'inversion matricielle, on obtient la formulation suivante avec le facteur d'oubli :

$$P_{k+1} = \frac{1}{\lambda} \cdot \left[P_k - \frac{P_k \cdot \Psi^T(k+1) \cdot \Psi(k+1) \cdot P_k}{1 + \Psi(k+1) \cdot P_k \cdot \Psi^T(k+1)} \right] \quad (5.16)$$

$$\overset{\Lambda}{\theta}_{k+1} = \overset{\Lambda}{\theta}_k + P_{k+1} \cdot \Psi^T(k+1) \cdot \left[y(k+1) - \Psi(k+1) \cdot \overset{\Lambda}{\theta}_k \right] \quad (5.17)$$

L'introduction du facteur d'oubli λ permet également d'identifier des paramètres qui varient lentement dans le temps. Un λ plus petit permet une meilleure "poursuite" de paramètres variables, alors qu'un λ plus grand permet une meilleure élimination des perturbations par effet de lissage.

Exemple :

Soit un système de premier ordre donné par :

$$y(k) + a_0 \cdot y(k - 1) = b_0 \cdot u(k - 1) + e_0(k) \quad (5.18)$$

Avec $\theta_0 = [a_0 \ b_0]$ les vrais paramètres du système, $u(k)$ l'entrée spécifiée, $y(k)$ la sortie mesurée et $e_0(k)$ un bruit blanc de moyenne nulle. Remarquons que dans cet exemple, $e_0(k)$ ne correspond pas au traditionnel bruit additif sur la sortie (“*bruit de sortie*”), mais plutôt à un “*bruit d'équation*” blanc.

L'objectif est d'estimer le vecteur de paramètres θ sur la base des séquences d'entrée et de sortie $u(k)$ et $y(k)$. La minimisation du critère quadratique basée sur l'erreur d'équation sera utilisée.

Système stationnaire sans facteur d'oubli :

Le cas d'un système stationnaire est d'abord considéré : $\theta_0 = [0.5 \ 0.5]^T$.

La [Figure 5.1](#) représente l'entrée $u(k)$ excitant le système et la [Figure 5.2](#) représente la réponse bruitée $y(k)$.

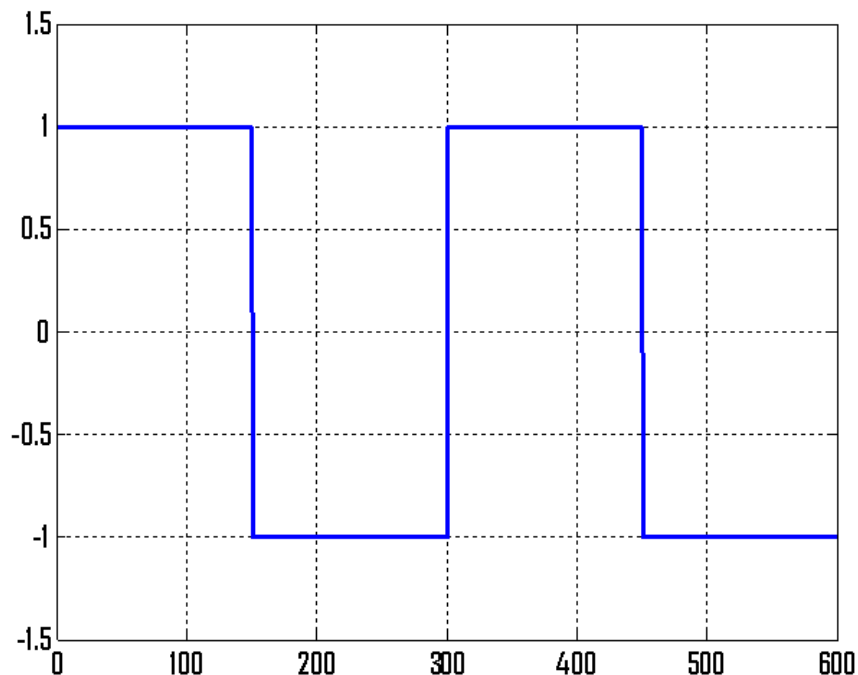


FIG. 5.1 – Signal d'entrée.

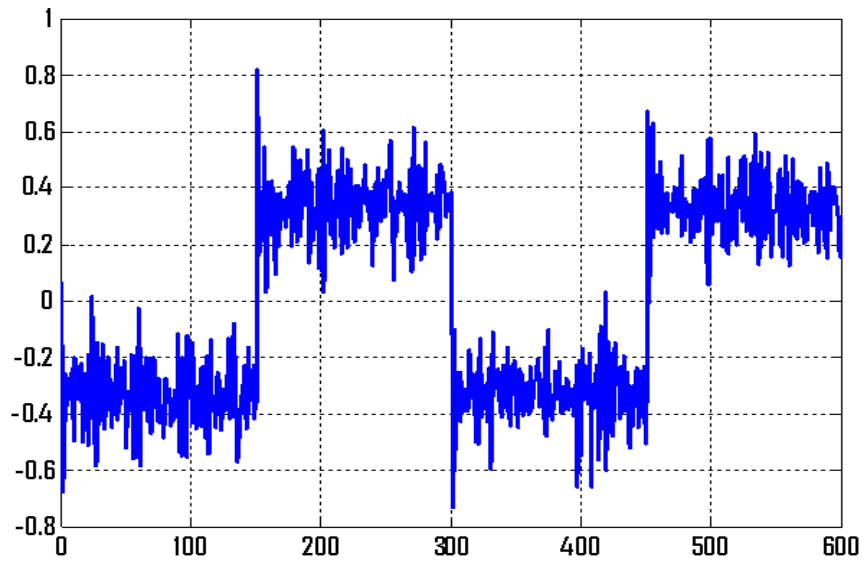


FIG. 5.2 – Réponse bruitée.

Pour $\hat{\theta}_0 = [0 \ 0]^T$ et $P = I$, l'Équation 5.3 et l'Équation 5.4 permettent d'estimer les paramètres du système (voir Figure 5.2) et la matrice P appelée communément la matrice de covariance de l'erreur d'estimation. On remarque ici que les paramètres estimés convergent rapidement vers les vraies valeurs et que les éléments de la matrice P ($P_{21} = P_{12} = 1000$ puisque la matrice P est symétrique).

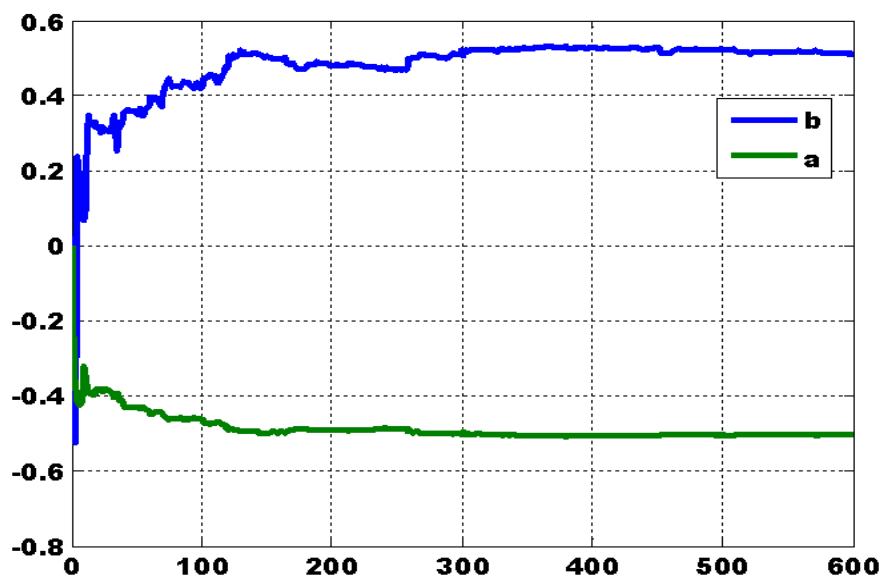
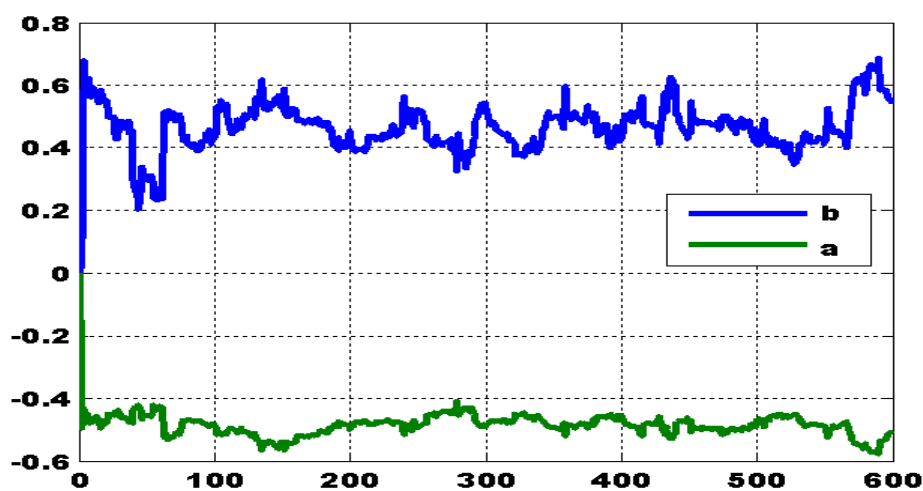
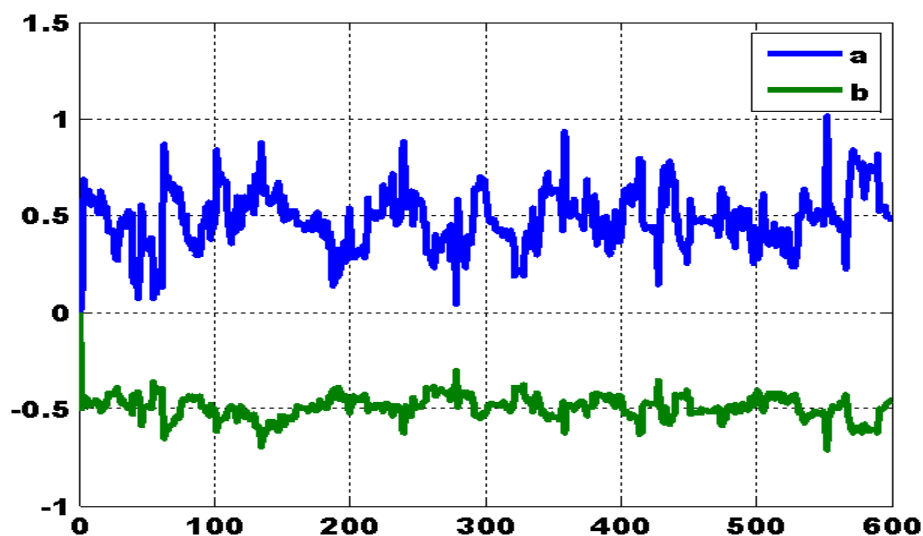
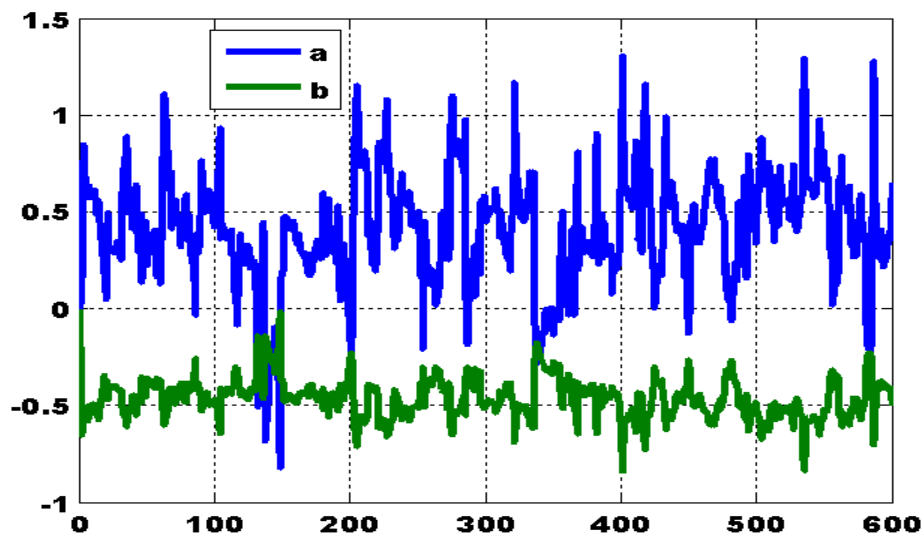


FIG. 5.3 – Paramètres estimés sans un facteur d'oubli.

Système stationnaire avec facteur d'oubli :

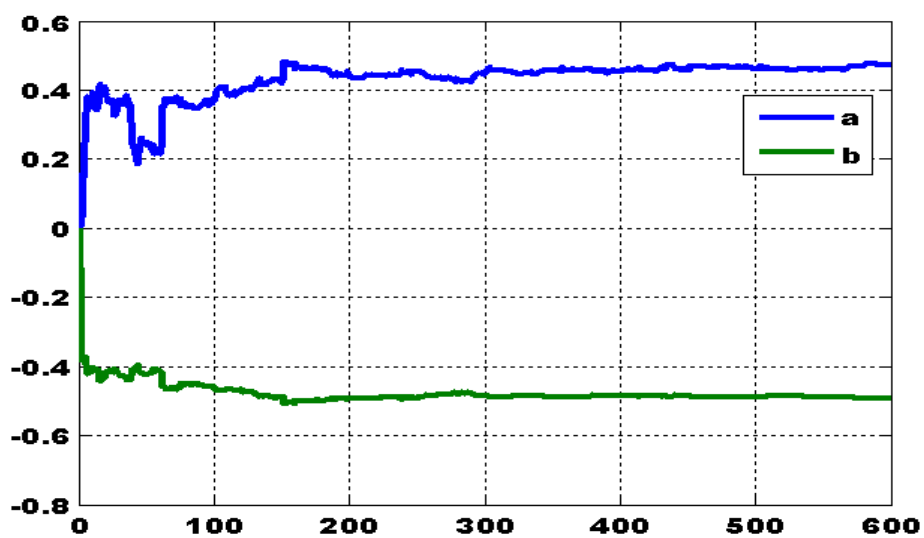
Pour le système stationnaire précédent, l'utilisation d'un facteur d'oubli ($\lambda = 0.97, 0.9, 0.8$) permet de conserver un gain suffisamment important pour suivre la variation des paramètres. Les éléments de P sont toujours les ceux utilisés précédemment (grâce à $\lambda < 1$) lorsque l'excitation du système est faible. Mais, comme auparavant, ils diminuent rapidement lorsque le système est fortement excité.

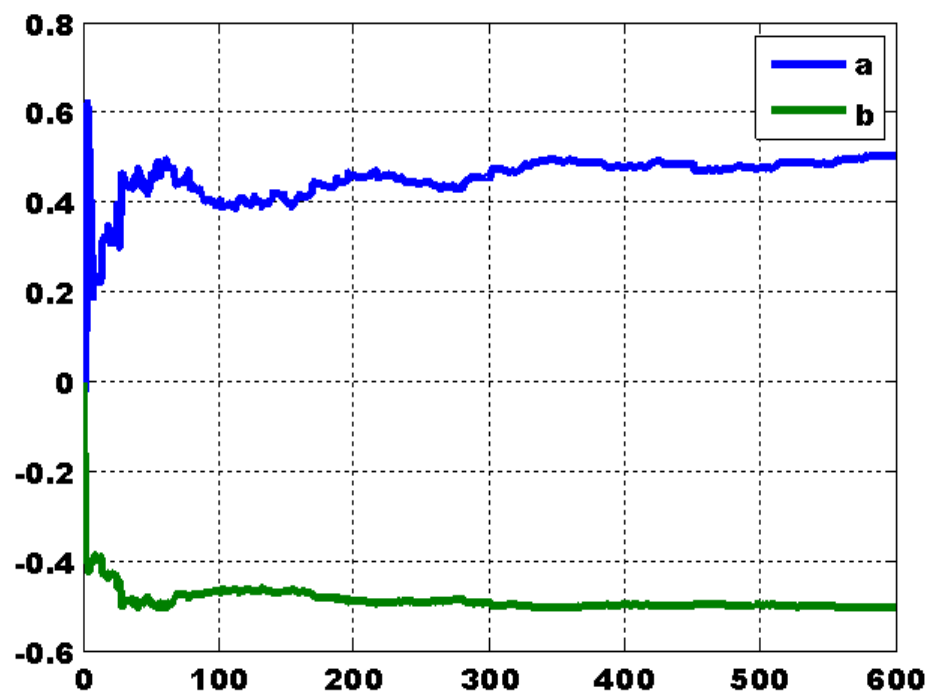
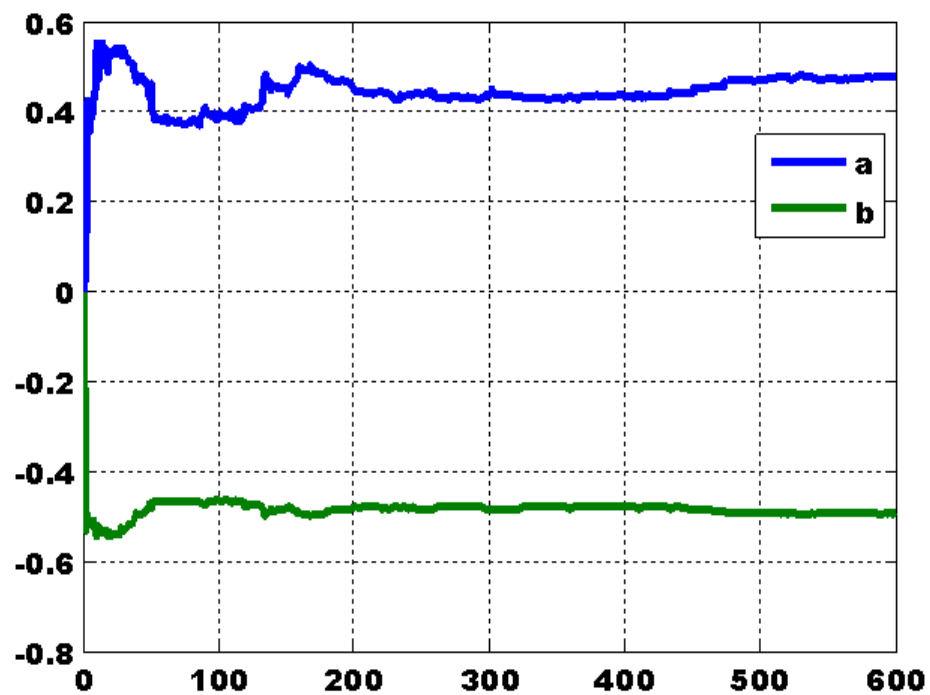
FIG. 5.4 – Paramètres estimés avec un facteur d'oubli ($\lambda = 0.97$).FIG. 5.5 – Paramètres estimés avec un facteur d'oubli ($\lambda = 0.90$)

FIG. 5.6 – Paramètres estimés avec un facteur d’oubli ($\lambda = 0.80$)

Système stationnaire sans facteur d’oubli avec variation de la matrice P :

1. $P > 1000$:

FIG. 5.7 – Paramètres estimés sans un facteur d’oubli pour $P = 1100$

FIG. 5.8 – Paramètres estimés sans un facteur d'oubli pour $P = 1500$ FIG. 5.9 – Paramètres estimés sans un facteur d'oubli pour $P = 5000$

1. $P < 1000$:

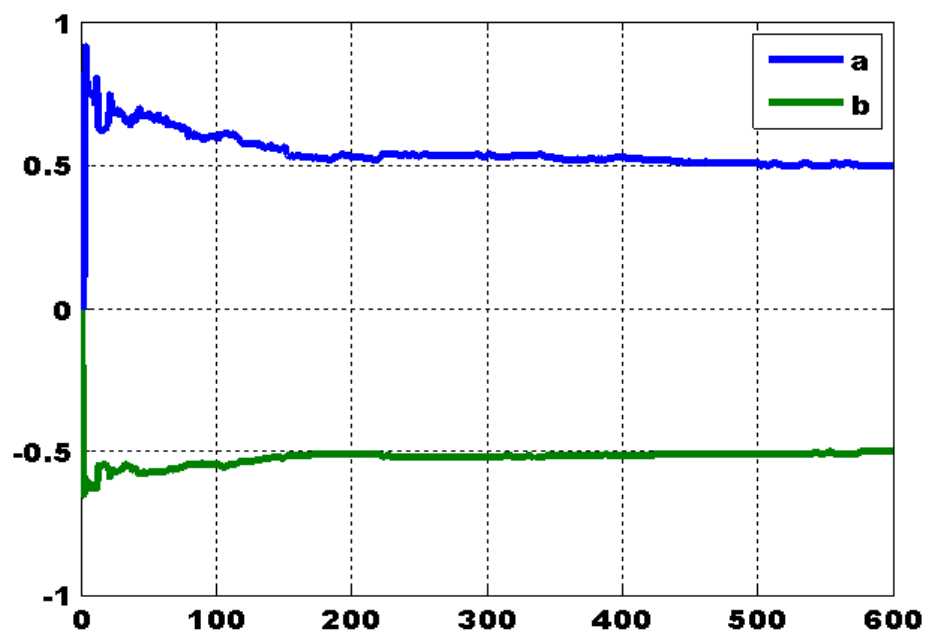


FIG. 5.10 – Paramètres estimés sans un facteur d’oubli pour $P = 500$

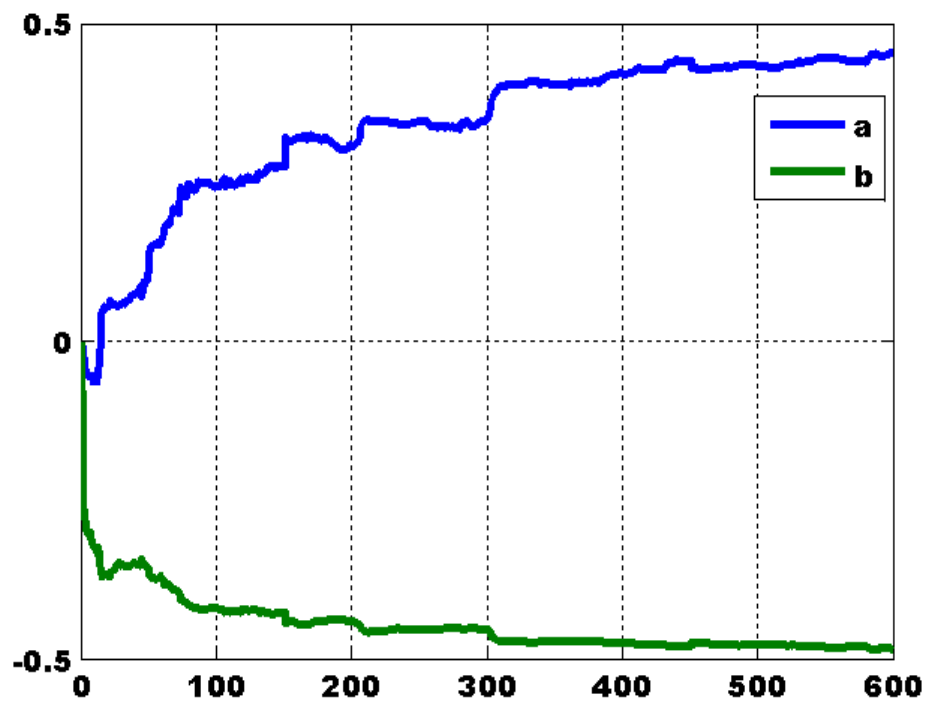


FIG. 5.11 – Paramètres estimés sans un facteur d’oubli pour $P = 1$

Système stationnaire avec facteur d'oubli et avec variation de la matrice

P :

1. $P > 1000$:

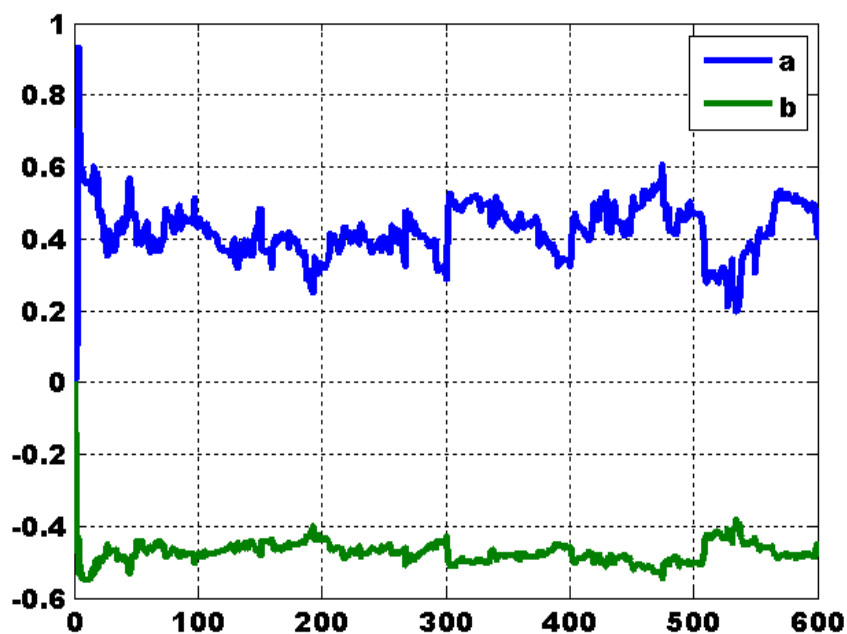


FIG. 5.12 – Paramètres estimés avec un facteur d'oubli ($\lambda = 0.97$) pour $P = 1100$

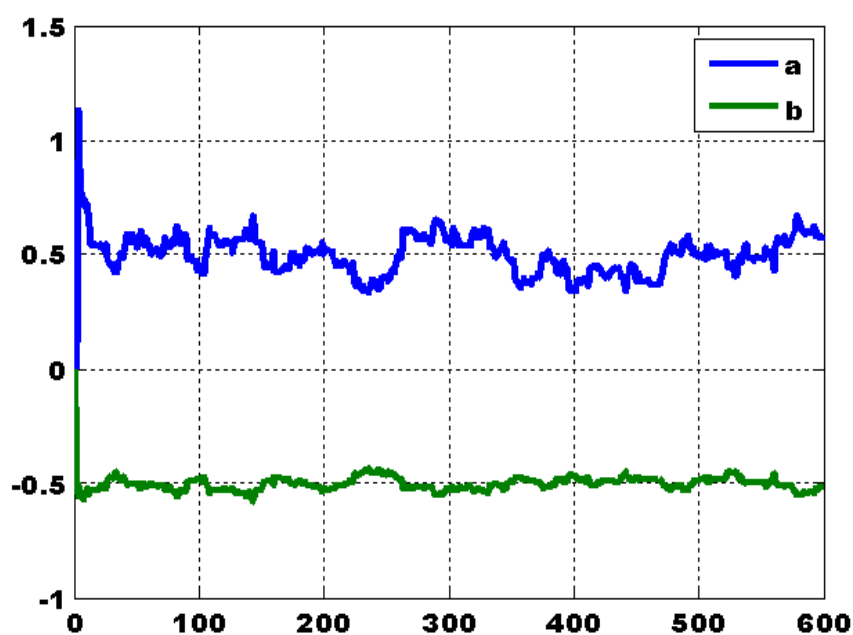


FIG. 5.13 – paramètres estimés avec facteur d'oubli ($\lambda = 0.97$) pour $P = 5000$

2. $P < 1000$:

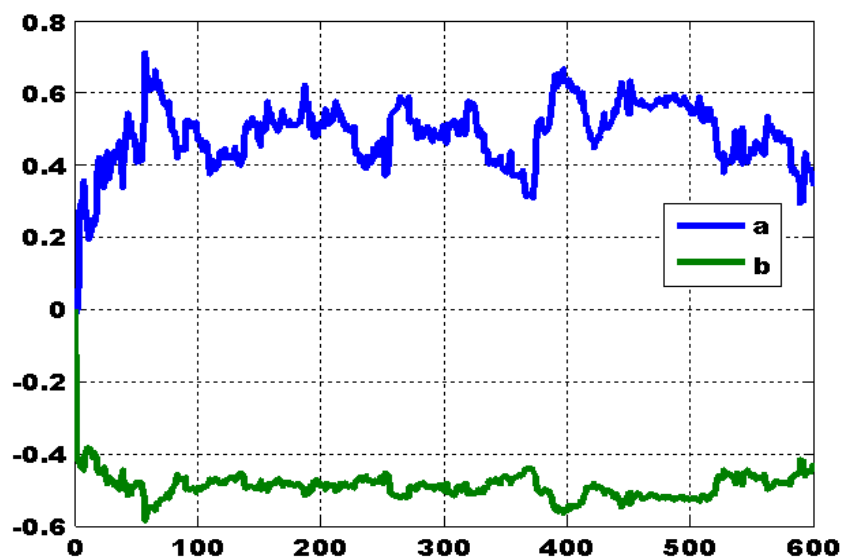


FIG. 5.14 – Paramètres estimés avec un facteur d'oubli ($\lambda = 0.97$) pour $P = 500$

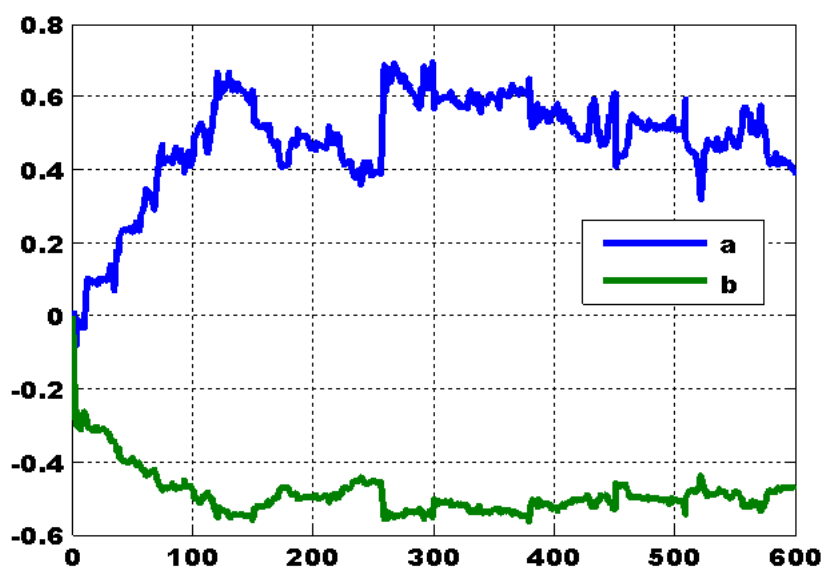


FIG. 5.15 – Paramètres estimés avec un facteur d'oubli ($\lambda = 0.97$) pour $P = 1$

En comparant le comportement des éléments de P dans les figures précédentes, on remarque que :

- Comme le gain d'adaptation reste suffisant ;
- L'algorithme peut ajuster correctement la valeur des paramètres.

Travaux pratiques

TP1 : Prise en main de SIMULINK

1. LES OBJECTIFS DE CE TP

- Maîtriser en main le logiciel de simulation SIMULINK de Matlab ,
- Simuler et de comparer les performances d'un système en boucle ouverte et en boucle fermée.

2. PRÉSENTATION DE SIMULINK

SIMULINK est une extension du logiciel Matlab. C'est un logiciel de simulation des systèmes dynamiques munis d'une interface graphique qui facilite les deux phases d'utilisation du logiciel :

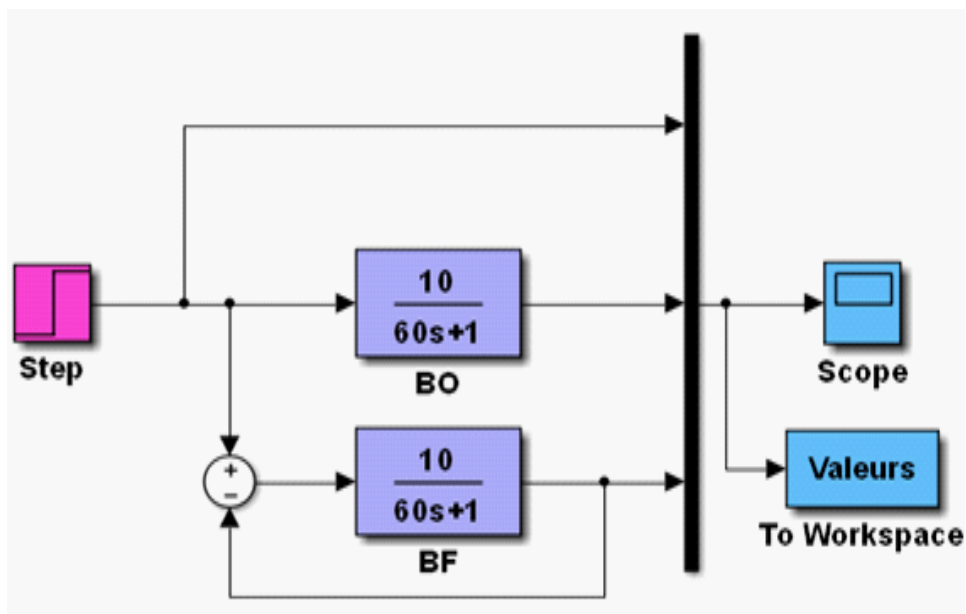
- Saisie du modèle,
- Simulation du modèle.

3. SIMULATION D'UN SYSTÈME EN BOUCLE OUVERTE VERSUS BOUCLE FERMÉE

Ouvrez un nouveau document et construisez un modèle pour avoir la réponse à un échelon en *BO* (Boule Ouverte) et en *BF* (Boule Fermée) d'un filtre du premier ordre ayant la fonction de transfert suivante :

$$\frac{10}{60 \cdot P + 1}$$

Le schéma-bloc doit avoir la forme de celui présenté sur la figure ci-dessous :



Les éléments de ce schéma-bloc sont :

Step : se trouve dans **Sources**, générateur de l'échelon. Choisissez la date 0 et d'amplitude 1.

Transfert fcn : se trouve dans **Continuous**. Permet de définir une fonction de transfert. Changez les paramètres pour qu'elle corresponde à la fonction de transfert demandée.

Mux : se trouve dans **Signal routing**. Permet de multiplexer plusieurs signaux dans un fil. Si on veut avoir l'entrée (échelon), la sortie en *BO* et en *BF* dans le même graphique, il faut avoir 3 entrées au **Mux**.

Sum : se trouve dans **Math Operations**. Permet de réaliser le comparateur. Il faut choisir les signes + et -.

Scope : se trouve dans **Sinks**. C'est un scope rudimentaire pour avoir rapidement un tracé des courbes.

L'icône permet d'adapter automatiquement les échelles.

ToWorkspace : se trouve dans **Sinks**. Permet de récupérer le résultat de la simulation dans une variable exploitable sur Matlab (ligne de commande). Paramétrez ce bloc pour avoir le résultat au format array.

4. TRAVAIL DEMANDÉ

a- Lancez la simulation.

b- Pour avoir un tracé exportable et imprimable, tapez dans Matlab (commande en ligne) l'instruction : `plot(tout,valeurs)` où *valeurs* est le nom de la variable entrée dans le bloc To Workspace. La variable *tout* est le vecteur qui contient les temps des points de calcul de la simulation.

TP2 : Simulation d'un système décrit par l'équation d'état

1. Système mécanique

L'équation d'état d'un système Masse-Ressort-Amortisseur est la suivante :

$$\dot{X} = \begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{K}{M} & -\frac{b}{M} \end{bmatrix} \cdot \begin{bmatrix} x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M} \end{bmatrix} \cdot F(t)$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$$

Masse ($M = 1.0\text{kg}$),

Constante de rappel ($K = 1.0\text{N/m}$),

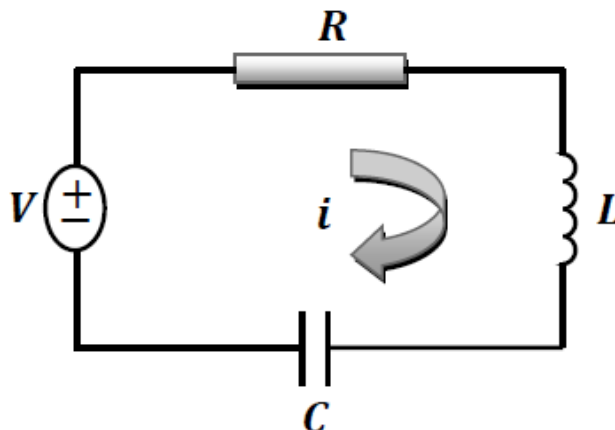
Constante d'amortissement ($b = 0.2\text{Ns/m}$),

La force d'entrée ($F = 1.0\text{N}$).

- Entrer le modèle d'espace d'état en *MATLAB* "où la commande principale utilisée est : *ss*".
- Entrer le modèle d'espace d'état en Simulink "à l'aide le bloc *state-space* trouvé en *continuous*".
- Donner la réponse indicielle de ce système.

2. Système électrique

Trouver la représentation d'état du circuit ci-dessous, et modéliser en *Matlab* puis Simulink ce circuit en donnant la réponse indicielle ?



Alimentation ($V = 10Volt$),

Résistance ($R = 400\Omega$),

Inductance ($L = 10H$),

Capacité ($C = 100\mu F$).

TP3 : Identification non paramétrique par la méthode de Strejc

1. Présentation de la méthode

Cette méthode peut s'appliquer aux systèmes dont la réponse indicielle ne présente pas de dépassement. On identifie à une fonction de la forme :

$$\frac{K \cdot e^{-r \cdot P}}{(1 + \tau \cdot P)^n}$$

Les paramètres à identifier sont donc : le gain statique K , le retard r , la constante de temps τ et l'ordre n .

Pour identifier le système, la méthode peut se décomposer en :

Le gain statique est mesuré directement par la valeur finale de la sortie. Celle-ci vaut $K \cdot E_0$ où E_0 est l'amplitude de l'échelon d'entrée.

On trace la tangente au point d'inflexion I pour déterminer deux valeurs : T_u et T_a "voir la figure pour la mesure de ces deux temps".

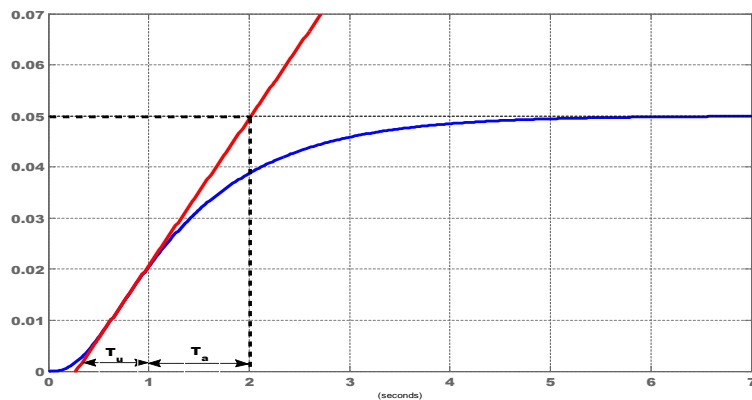
Dans la colonne T_u/T_a du tableau trouver la valeur immédiatement inférieure à ce rapport.

Sur la ligne de ce ratio déterminer n .

Toujours à l'aide des valeurs numériques de cette ligne, calculer τ avec $\frac{T_a}{\tau}$.

Calculer la nouvelle valeur de $T_{u1} = \frac{T_u}{\tau}$.

En déduire r avec $r = T_{u1} - T_u$.



Méthode de Strejc : Réponse indicielle

n	T_u/T_a	T_u/τ	T_a/τ
1	0	0	1
2	0.104	0.282	2.718
3	0.218	0.805	3.695
4	0.319	1.425	4.465
5	0.410	2.1000	5.119
6	0.493	2.811	5.699
7	0.570	3.549	6.226
8	0.642	4.307	6.711
9	0.709	5.081	7.164
10	0.773	5.869	7.590

Tableau de Strejc

2. Travail à faire

Considérons un système de 3^{ème} ordre caractérisé par sa fonction de transfert suivante :

$$\frac{1}{(P+1)(P+4)(P+5)}$$

- a- Faire la construction graphique permettant de déterminer les paramètres : *le gain statique K, le retard r, la constante de temps τ et l'ordre n* pour obtenir le modèle de *Strejc*.
- b- Tracer la réponse indicielle du modèle et la superposer sur celle du système dans la même figure.
- c- Calculer l'erreur entre la réponse du système et celle du modèle.

Code source Matlab (courbe+la tangente dans le point d'inflexion) :

```
clc;clear all; close all
s=tf('s'); H=1/((s+1)*(s+4)*(s+5)); [y,t]=step(H);
d1y = gradient(y,t); % La première dérivée numérique
d2y = del2(y,t); % La dérivée2 numérique
t_infl = interp1(d1y, t, max(d1y)); % Trouver 't' au maximum de la première dérivée
y_infl = interp1(t, y, t_infl); % Trouver 'y' au maximum de la première dérivée
```

```
slope = interp1(t, d1y, t_infl); % Pente définie ici comme maximum de 1ere dérivée
intcpt = y_infl - slope*t_infl; % Calculer l'interception
tngt = slope*t + intcpt; % Calculer Ligne tangente
figure(1)
step(H)
hold on
plot(t, tngt, 'r', 'LineWidth',0.5)
hold off
grid on,axis([0 7 0 0.1])
```

TP4 : Identification non paramétrique par la méthode de Broida

1. Présentation de la méthode

Le modèle proposé pour approcher le comportement du système est un premier ordre avec un retard pur. Sa fonction de transfert est :

$$\frac{K \cdot e^{-r \cdot P}}{1 + \tau \cdot P}$$

Le gain K est déterminé comme dans la méthode de Strejc avec la valeur finale de la sortie.

Soient t_1 et t_2 les temps au bout desquels la réponse expérimentale atteint respectivement 28% et 40% de la valeur finale. On va simplement résoudre le système donné par :

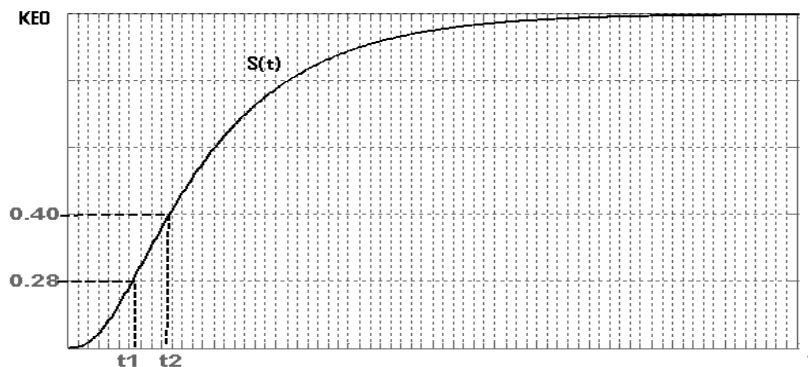
$$\frac{s(t)}{K \cdot E_0} = 0.28 \implies t_1 - r = 0.328 \cdot \tau$$

$$\frac{s(t)}{K \cdot E_0} = 0.40 \implies t_2 - r = 0.510 \cdot \tau$$

La résolution de ces équations donne :

$$\tau = 5.5 \cdot (t_2 - t_1)$$

$$r = 2.8 \cdot t_1 - 1.8 \cdot t_2$$



Méthode de Broida : Réponse indicielle

2. Travail à faire

Considérons un système de 3^{ème} ordre caractérisé par sa fonction de transfert suivante :

$$\frac{1}{(P+1)(P+4)(P+5)}$$

- a- Faire la construction graphique permettant de déterminer les paramètres : le gain statique K , le retard r et la constante de temps τ pour obtenir le modèle de *Broida*.
- b- Tracer la réponse indicielle du modèle et la superposer sur celle du système dans la même figure.
- c- Calculer l'erreur entre la réponse du système et celle du modèle.
- d- Comparer le modèle de *Broida* et le modèle de *Strejc*.

TP5 : Identification paramétrique par la méthode des moindres carrés

1. Identification d'un système modélisé par la méthode ARMA

– Introduction

La méthode Autorégressive à Moyenne Mobile Ajustée ARMA utilise un modèle de représentation très universel, traduisant une relation entrée-sortie du processus du type récurrence en tenant compte d'un éventuel retard. L'ajustage des coefficients se fait par la mesure des signaux entrée-sortie sur le processus réel pour une large gamme de signaux d'entrée.

Le modèle de comportement défini dans la méthode ARMA est du type discret : Ce choix facilite son utilisation dans les algorithmes de commande complexes qui seront établis sur une machine numérique. De plus, la simulation des résultats (validation du modèle) est directement effectuée à l'aide du calcul numérique. Tandis que l'ajustement des paramètres du modèle se fait par une méthode de moindres carrés récursifs, ce qui minimise la taille des données intervenant dans les calculs. Cette technique permet en plus de modéliser des systèmes dont les paramètres évoluent dans le temps : le modèle suit ces variations avec un effet de "filtrage" réglable.

– Le modèle ARMA

Le modèle ARMA correspond à une fonction de transfert discrète développée sous forme d'une récurrence qui établit une relation linéaire entre la suite des entrées $\{u(k)\}$ et celle des sorties modélisées $\{y_m(k)\}$ de la forme :

$$a_0 \cdot y_m(n) + \dots + a_p \cdot y_m(n - p) = b_0 \cdot u(n) + \dots + b_q \cdot u(n - q)$$

2. Ajustement des paramètres du modèle ARMA

L'identification des coefficients du modèle ARMA s'appuie sur l'observation des signaux $u(t)$ et $y(t)$. Il est à remarquer que la forme du signal $u(t)$ n'intervient pas directement dans la phase d'identification, contrairement aux méthodes qui s'appuient sur une excitation particulière du système (*impulsion, échelon, SBPA,...*).

– Méthode des moindres carrés

La méthode la plus utilisée pour déterminer les paramètres a_i et b_i de ce modèle consiste à évaluer l'écart quadratique moyen entre la sortie réelle du processus $\{y(k)\}$ et celle du modèle $\{y_m(k)\}$ et à ajuster les paramètres du modèle pour minimiser cet écart quadratique.

La sortie y_m du modèle à l'instant k , connaissant la suite des entrées, peut s'écrire (en supposant que $a_0 = 1$) :

$$y_m(n) = -a_1 \cdot y_m(n-1) \cdots - a_p \cdot y_m(n-p) + b_0 \cdot u(n) + \cdots + b_q \cdot u(n-q)$$

Soit v l'écart entre la sortie réelle y du processus et la sortie y_m du modèle à l'instant i :

$$v(i) = y(i) - y_m(i)$$

L'équation précédente devient :

$$y(n) = -a_1 \cdot y(n-1) \cdots - a_p \cdot y(n-p) + b_0 \cdot u(n) + \cdots + b_q \cdot u(n-q) + e(n)$$

$$e(n) = - (v(i) + a_1 \cdot v(n-1) + \dots + a_p \cdot v(n-p))$$

$e(n)$ est appelé résidu ou erreur de prédiction. C'est l'écart entre la sortie réelle et la sortie prédite à l'instant k .

Supposons que l'on fasse N mesures successives sur le processus du couple entrée-sortie. On peut écrire $(N-p)$ fois l'équation précédente, l'ensemble des relations est regroupé sous forme matricielle :

$$\begin{bmatrix} y(n) \\ \vdots \\ y(p+1) \end{bmatrix} = \begin{bmatrix} -y(n-1) & \dots & u(n-q) \\ \dots & \dots & \dots \\ -y(p) & \dots & u(p+1-q) \end{bmatrix} \cdot \begin{bmatrix} a_1 \\ \vdots \\ b_q \end{bmatrix} + \begin{bmatrix} e(n) \\ \vdots \\ e(p+1) \end{bmatrix}$$

De la forme $Y = H \cdot \theta + \varepsilon$ avec θ vecteur des paramètres à estimer.

Remarque :

Les équations peuvent s'écrire pour un instant $n < p + 1$ en affectant éventuellement des valeurs nulles aux signaux $u(j)$ et $y(j)$ pour lesquels $j < 0$.

– **Estimation des paramètres**

La méthode d'estimation des meilleures valeurs des paramètres est la méthode des moindres-carrés dont les étapes principales sont rappelées ici :

Pour estimer θ mise un critère quadratique J somme des carrés des erreurs de prédiction :

$$J(\theta) = \sum_{i=p+1}^n \varepsilon_i^2 = \varepsilon^T \cdot \varepsilon = (Y - H \cdot \theta)^T \cdot (Y - H \cdot \theta)$$

Le minimum de J est obtenu en recherchant la valeur θ qui annule les dérivées partielles par rapport à chacune des composantes de θ soit $\partial J / \partial \theta = 0$ et l'optimum de θ est donné par :

$$\hat{\theta} = (H \cdot H^T)^{-1} \cdot H^T \cdot Y$$

Le vecteur paramètre ainsi obtenu permet de définir les coefficients du modèle ARMA du processus qui minimise l'erreur entre les sorties réelles et celle du modèle au sens des moindres carrés.

3. Travail à faire

Prenons le cas d'un système du premier ordre, type réponse d'un circuit RC à un échelon de tension. Le modèle ARMA est donné par :

$$y_m(n) = -a \cdot y(n-1) + b \cdot u(n-1)$$

avec :

$$a = -e^{-t/\tau}$$

$$b = K \cdot (1 - e^{-t/\tau})$$

- a- A l'aide des valeurs de u et y données, construire la matrice H pour environ 70 lignes.
- b- Calculer le vecteur θ des paramètres identifiés par la méthode des moindres-carrés.
- c- Comparer aux valeurs obtenues directement.

- d- Simuler le système obtenu et superposer le tracé graphique à celui de la réponse directe.
- e- Comparer les différentes valeurs (la valeur finale et le temps de réponse) du système et du modèle obtenu.

Code utilisé pour la simulation :

```
clc;clear all;close all

%génération de mesure

Te=0.01;

t=0 :Te :0.7;

taux=0.1%

N=length(t);

br=rand(1,N);

y=1-exp(-t/taux)+0.05*br;

%estimation des paramètres

H=[-y(1 :N-1)' ones(1, N-1)'];

teta=inv(H'*H)*H'*y (2 :N)';

a=teta(1);

b=teta(2);

tauxest=-Te/log(-a);

kest=b/(1+a);

plot(t,y,'r',t,kest*(1-exp(-t/tauxest)),'b');
```

Bibliographie et Webographie

1. Bourlès, Henri. *Systèmes linéaires : de la modélisation à la commande*. Lavoisier, 2006.
2. Borne, Pierre, and Jean-Pierre Richard. *Modélisation et identification des processus*. Vol. 1. Editions Technip, 1992.
3. Landau, Ioan Doré, and Gianluca Zito. *Digital Control Systems : Design, Identification and Implementation*. Springer Science & Business Media, 2005.
4. Duflos, Emmanuel, and Philippe Vanheeghe. *Estimation prédiction : éléments de cours et exercices résolus*. Vol. 13. Editions Technip, 2000.
5. Abdenmour, Ridha Ben. *Identification et commande numérique des procédés industriels*. Vol. 11. Editions Technip, 2001.
6. Bonvin D, Karimi A. *Identification de systemes dynamiques*. Lausanne, October. 2006.
7. *www.Wékipédia.org*