

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE.
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHE SCIENTIFIQUE.
UNIVERSITE MOULOUD MAMMARI, TIZI-OUZOU
FACULTE DE GENIE ELECTRIQUE ET INFORMATIQUE
DEPARTEMENT AUTOMATIQUE



Spécialité : Automatique et Systèmes

MEMOIRE DE FIN D'ETUDES

THEME

**Identification des Systèmes Dynamiques
Par l'Optimisation Globale**

Présenté par : LOUNES Sarah et OULD MAHAMMED Sylia

Soutenu publiquement, le 07/Octobre/2019, devant le jury composé de :

Encadré par :

M^{me} IDIRI Ghania

MCB

Promotrice

Devant les membres du jury:

M^{me} NAIT ABDESSELAM Aldjia

MAA

Présidente

M^{me} HAMRI Hakima

MCB

Examinatrice

M^{me} MEGHERBI Ouerdia

MAB

Examinatrice

Année universitaire : 2018/2019

Remerciements

Nous remercions Dieu Clément et Miséricordieux, tout puissant d'avoir guidé nos pas vers les portes du savoir tout en illuminant notre chemin, et nous avoir donné suffisamment de courage et de persévérance pour mener notre travail à terme.

*Nous témoignons une reconnaissance particulière pour notre promotrice, que Mme **IDIRI Ghania** reçoive l'expression de toute notre reconnaissance pour ses précieux conseils et orientations.*

*Nos plus sincères remerciements vont aux membres du jury: **M^{me} NAIT ABDESSELAM AIdjia**, **M^{me} HAMRI Hakima** Et **M^{me} MEGHERBI Ouerdiade** nous avoir honoré en acceptant d'évaluer notre travail.*

Nous ne saurons oublier le grand mérite des enseignants qui ont contribué à notre cursus particulièrement ceux du département "Automatique" et qu'ils trouvent ici le témoignage de notre profonde reconnaissance.

Nous remercions aussi tous nos camarades du département Automatique en particulier nos amis de la promotion. On leur exprime notre profonde sympathie et leur souhaitent beaucoup de réussite. Finalement, nous remercions toute personne ayant contribué de près ou de loin à l'élaboration de ce mémoire.

Dédicace

Je dédie ce travail ...

À mes chers parents ma mère FARIDA et mon père HAMID

Aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez consenti pour mon instruction et mon bien être.

Puisse Dieu, le Très Haut, vous accorder santé, bonheur et longue vie et faire en sorte que jamais je ne vous déçoive.

À Mon cher frère MOUHAND ALI

Je te souhaite une vie pleine de bonheur et de succès et que Dieu, le tout puissant, te protège et vous garde.

À la mémoire de mon grand-père maternelle qu'il repose en paix.

À ma chère grande mère et ses enfants

À mon cher oncle ABDE LKRIM, sa femme SABRINA et leurs enfants LYCIA, YOUNES et aussi à ROMAÏSSA et RAYANE

À ma chère tante KAHINA et son époux,

Veillez trouver dans ce travail l'expression de mon respect le plus profond et mon affection la plus sincère.

À MASSI et toute sa famille, je vous souhaite une vie pleine de bonheur et de succès et que dieu, le tout puissant, vous protège et vous garde.

À ma meilleure amie Ourdia

En souvenir de notre sincère et profonde amitié et des moments agréables que nous avons passés ensemble.

À ma chère binôme SYLIA

SARAH

Dédicace

Je dédie ce travail ...

A mes chers parents ma mère MALIKA et mon père AHMED

Aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez consenti pour mon instruction et mon bien être.

Puisse Dieu, le Très Haut, vous accorder santé, bonheur et longue vie et faire en sorte que jamais je ne vous déçoive.

A mon cher frère ADEL et mes chère sœurs FATMA, LYDIA et ma chère petite sœur NAWEL.

Je vous souhaite une vie pleine de bonheur et de succès et que Dieu, le tout puissant, vous protège et vous garde.

A tous mes amis(es) et tous ceux que j'aime

Veillez trouver dans ce travail l'expression de mon respect le plus profond et mon affection la plus sincère.

Ama chère binôme SARAH

SYALIA

Liste des figures

Figure I.1 : Principe de l'identification

Figure I.2 : les étapes de l'identification

Figure I.3 : la représentation du procédé

Figure I.4 : Génération d'une séquence binaire pseudo -aléatoire

Figure I.5 : signal SBPA avec 4 bits

Figure I.6 : réponse indicielle d'un procédé du premier ordre.

Figure I.7 : réponse indicielle d'un procédé du premier ordre retardé

Figure I.8 : réponse indicielle d'un procédé de deuxième ordre non oscillatoire

Figure I.9 : réponse indicielle d'un procédé de deuxième ordre non oscillatoire retardé

Figure I.10 : réponse indicielle d'un procédé de deuxième ordre oscillatoire

Figure I.11 : réponse indicielle d'un procédé de deuxième ordre oscillatoire retardé.

Figure I.12 : la réponse d'un système du premier ordre à un échelon

Figure I.13 : réponse d'un système intégrateur à un échelon

Figure I.14 : la réponse d'un système du premier ordre retardé à un échelon

Figure I.15 : la réponse du strejc

Figure I.16: la réponse d'un système de deuxième ordre oscillatoire

Figure I.17 : schéma synoptique de la méthode du modèle.

Figure III.18 : représentation des sorties (mesurée/prédite)

Figure III.19 : représentation des sorties (mesurée/prédite)

Liste des tableaux

Tableau I.1 : Paramètres obtenus pour différents types d'entrée

Tableau I.2 : Génération des SBPA de longueur maximale

Tableau I.3 : tableau de génération d'une SBPA

Tableau I.4 : les différentes méthodes d'identification

Tableau I.5 : Coefficients de la méthode de strejc

Tableau II.6 : Classification des problèmes d'optimisation

Tableau II.7 : Nature des points critiques

Liste des indices

$*$	grandeur optimal
α	Nombre réel strictement positif
α_i	Nombre réel
α_k	coefficients de la paramétrisation
b	Nombre réel positif
D	Domaine admissible pour un problème d'optimisation
e	Nombre réel
ε	Erreur d'approximation
J	Critère de performance
k	temps discret
x_i	variable de décision
$y(k)$	la sortie du système
$u(k)$	la commande appliquée au système
$g_i(x)$	contraintes d'égalité
$h_j(x)$	contraintes d'inégalité
$f(x)$	fonction quelconque
τ	retard de système
r	le rayon de coordonnée polaire
θ	l'angle de coordonnée polaire
α	la spirale d'Archimède
$\frac{\partial f(x)}{\partial x_i}$	dérivée partielle première de f par rapport à x_i
$\frac{\partial^2 f(x)}{\partial x_i^2}$	dérivée partielle secondaire de f par rapport à x_i
$\nabla_x f(x)$	gradient de f par rapport à x
$\nabla_x^2 f(x)$	hessienne de f par rapport à x

Sommaire

Introduction générale.....	1
Chapitre 1 : l'identification des systèmes dynamiques	
I.1 Introduction.....	3
I.2 Principe de l'identification.....	3
I.3 les étapes de l'identification.....	4
I.3.1 Acquisition des données	5
I.3.1.1 Illustration de l'importance de signal SBPA.....	5
I.3.1.2 La séquence binaire SBPA.....	6
I.3.1.3 Exemple de génération d'une SBPA.....	8
I.3.2 Choix de la structure du modèle d'identification.....	9
I.3.3 Choix de la méthode d'identification	13
I.3.4 La validation du modèle	13
I.4 Les méthodes d'identification	13
I.4.1 Méthodes graphiques	14
I.4.1.1 Méthodes basées sur les réponses temporelles	15
I.4.1.2 Méthode des systèmes oscillatoires	19
I.4.2 Méthode numériques.....	21
I.4.2.1 Méthode de moindre carré simple.....	21
I.4.2.2 Méthodes des moindres carrés récursifs	23
I.4.3 Méthode du modèle.....	26
Conclusion	27
Chapitre 2 : Optimisation Globale	
II.1 Introduction	28
II.2 Formulation d'un problème d'optimisation 2.....	28
II.2.1 Cas de minimisation	28
II.2.3 Cas de maximisation.....	28

II.3 Classification d'un problème d'optimisation.....	29
II.4 Condition d'optimalité.....	31
II.4.1 Condition du 1 ^{er} ordre.....	31
II.4.2 Condition du 2 ^{eme} ordre	31
II.5 Méthode de la résolution d'un problème d'optimisation.....	32
II.5.1 Méthodes numériques.....	33
II.5.1.1 La structure générale d'un algorithme d'une méthode numérique	34
II.5.1.2 La méthode du gradient	34
II.5.1.3 Méthode du gradient conjugué	35
II.5.1.4 Méthode de Newton	37
II.5.1.5 Méthode de Newton modifiée	37
II.5.2 Méthodes globale	39
II.5.2.1 Les méthodes stochastiques	40
II.5.2.2 Méthodes hybrides.....	40
II.5.2.3 Les méthodes exactes	41
Conclusion.....	45

Chapitre III : Application de la méthode d'Aliénor

III.1 Introduction	46
III.2 Identification des systèmes dynamique par l'optimisation Globale.....	46
III.2.1 Le principe de la méthode du modèle	46
III.2.2 Exemple d'application1	48
III.2.3 Exemple d'application2	52
Conclusion.....	53
Conclusion Générale	54

L'automatique est un domaine basé sur l'utilisation d'un modèle mathématique qui est obtenu par la modélisation du système. Ce dernier consiste à représenter les différentes relations existantes entre les variables caractéristiques du système par des équations mathématiques. Cependant plusieurs inconvénients sont rencontrés dans l'obtention d'un modèle complet des processus par la modélisation. En effet, il est très difficile voir impossible, pour certains systèmes, d'écrire toutes les équations mathématiques (lois de la physique). C'est dans l'optique de surmonter cette difficulté que l'identification est apparue pour simplifier l'obtention d'un modèle acceptable.

Identifier un système consiste à proposer une structure entre son entrée et sa sortie, ainsi de déterminer à partir du couple (entrée/sortie) les valeurs des paramètres de son modèle. Ainsi, différentes méthodes d'identification peuvent être utilisées. Les différentes méthodes développées peuvent être classées en trois catégories : les méthodes graphiques (Strejc, Broida,.....), les méthodes non récursives (méthodes des moindres carrés) et méthodes récursives (méthodes des moindres carrés récursifs).

De manière générale, le principe des méthodes d'identification consiste à minimiser l'écart observé entre la sortie prédite par le modèle et celle mesurée en définissant un critère. Ce principe est connu par la méthode du modèle qui est utilisée en pratique car le problème d'identification est formulé sous forme d'un problème d'optimisation dont les paramètres à identifier sont les variables de décision. La méthode du modèle reste l'une des méthodes d'identification les plus efficaces.

Après avoir transformé le problème d'identification à un problème d'optimisation, plusieurs méthodes d'optimisation peuvent être appliquées pour résoudre le problème. Pour avoir la solution globale du problème d'optimisation on doit choisir une méthode d'optimisation globale afin d'avoir les optimums globaux qui sont les paramètres du système.

Afin de surmonter ce problème, nous utilisons dans notre travail une étude pour l'identification d'une manière précise des paramètres d'un système dynamique basée sur la méthode d'Aliénor. Le but de cette dernière consiste à ramener le problème d'optimisation à plusieurs variables de décision (paramètres à identifier) à un problème d'optimisation à une seule variable de décision (facile à résoudre) à l'aide d'une transformation réductrice, ce qui permet de simplifier la résolution du problème d'optimisation et de localiser la solution globale du problème d'optimisation.

Le premier chapitre est consacré à l'identification paramétrique des systèmes dynamiques. Ce chapitre débute par le principe d'identification puis les étapes d'identification et se termine par les différentes méthodes d'identification.

Le deuxième chapitre constitue une introduction à la théorie d'optimisation dont on présente la forme générale d'un problème d'optimisation, puis on citera les conditions d'optimalités et on va classer les différents problèmes d'optimisation, enfin on citera les différentes méthodes d'optimisation en mettant l'accent sur la méthode d'optimisation globale d'Aliénor.

Dans le dernier chapitre on va identifier deux systèmes de types (entrée /sortie) en appliquant la méthode du modèle par l'optimisation globale en utilisant la méthode d'Aliénor.

Enfin on termine notre travail par une conclusion générale sur l'étude menée.

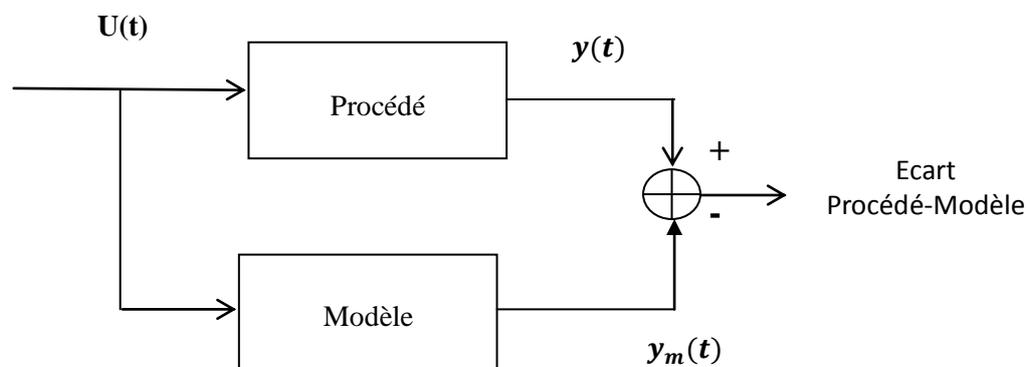
I.1 Introduction :

L'identification est une technique expérimentale qui s'appuie sur l'utilisation des procédures et des algorithmes manipulant les ressources expérimentales [2], et qui a pour but d'ajuster les paramètres du modèle de telle sorte à ce que le comportement du modèle soit identique à celui du système.

Ce chapitre est consacré à des notions de base concernant l'identification paramétrique des systèmes dynamiques et divers méthodes d'identification les plus souvent rencontrées et leurs mises en œuvre.

I.2 Principe de l'identification :

Identifier un système consiste à proposer une structure d'un modèle à partir des mesures obtenues et à déterminer ces paramètres de telle sorte que le comportement du modèle suit celui du système réel lorsque ces derniers sont soumis à une même entrée [2]. Le principe de l'identification est expliqué par la figure (I.1).



Figure(I.1) : Principe de l'identification

I.3 Etapes d'identification :

Pour bien identifier un système dynamique on doit respecter les étapes d'identification illustrées par la figure (I.2)

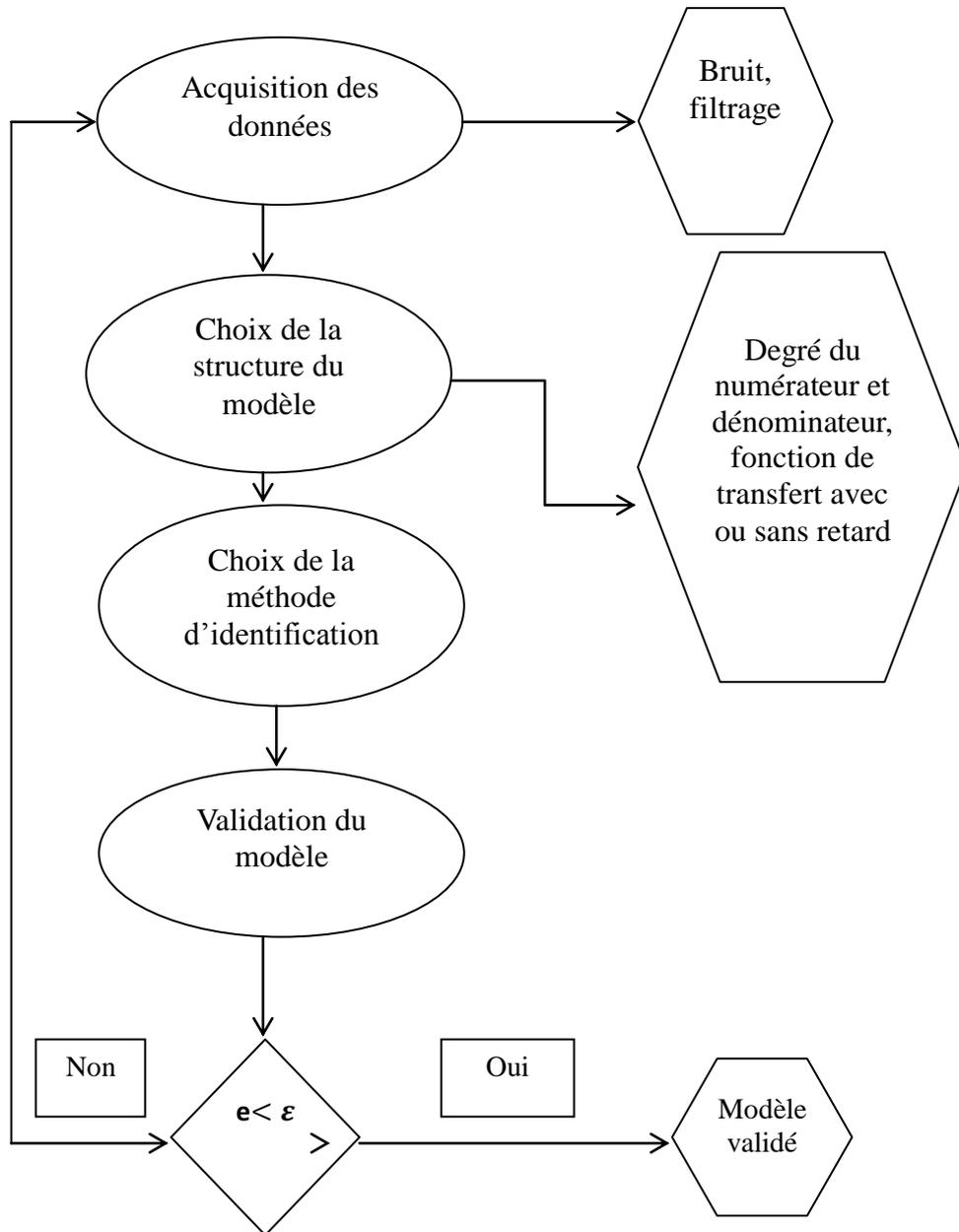


Figure (I.2) : les étapes de l'identification

I.3.1 Acquisition des données

Pour bien identifier un système, il faudra connaître son comportement à chaque instant et cela revient à choisir une entrée considérée comme un signal d'excitation dont ce dernier possède une densité spectrale pouvant exciter toute la dynamique du système à identifier.

Parmi les signaux d'excitations qu'on peut injecter comme une entrée :

- Séquence binaire pseudo-aléatoires « SBPA »,
- Signal constant,
- Signal impulsionnel,
- Echelon unitaire,
- Signal rectangulaire,
- Signal sinusoïdal ...etc.

I.3.1.1 Illustration de l'importance du signal d'excitation :

Pour illustrer l'importance du signal d'entrée pour le système à identifier, on considère un système du premier ordre dont la fonction de transfert est décrite par un gain statique $K = 1$ et une constante de temps $T = 8 \text{ secondes}$ [2].

On fait injecter à ce procédé différents types d'entrée, comme il est représenté dans la figure (I.3) :

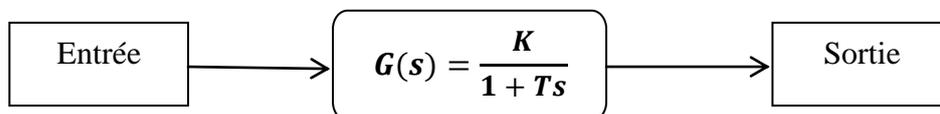


Figure (I.3) : la représentation du procédé

Les résultats obtenus pour chaque signal sont représentés dans le tableau suivant :

Signal	Constante	Impulsion	Echelon	Rectangulaire	SBPA
Gain (k)	1.0048	1.6826	1.0033	0.9980	1
Constante de temps (T)	3.3098	4.4800	7.6879	8.0671	7.98

Tableau (I.1) : Paramètres obtenus pour différents types d'entrée

Le tableau (I.1) présente les valeurs des paramètres K et T identifiés après excitation du système par une constante, une impulsion, un échelon, une entrée rectangulaire et enfin un signal binaire pseudo aléatoire (SBPA).

D'après les résultats obtenus, on remarque que le meilleur signal d'excitation est le signal SBPA car les paramètres obtenus sont très proches des paramètres réels ce qui montre l'importance du choix du signal d'excitation qui doit être riche en fréquence.

I.3.1.2 La séquence binaire pseudo-aléatoire (SBPA)

La SBPA (Séquence Binaire Pseudo Aléatoire) est un signal formé d'impulsions rectangulaires modulées aléatoirement en longueur, qui approxime un bruit blanc discret, donc il est riche en fréquence et de valeur moyenne nulle. [2]

La SBPA est un signal pseudo aléatoire car il est caractérisé par une « Longueur de Séquence » à l'intérieur de laquelle les variations de la largeur des impulsions varient aléatoirement, mais sur un grand horizon de temps. Elles sont périodiques et la période étant définie par la longueur de la séquence.

Ce signal est généré à l'aide de registres à décalage bouclé il est composé de N éléments à valeur binaire 0 ou 1. Lorsque la sortie du registre vaut 0, le signal est pris égale a S_{min} pendant une période T_{SBPA} et lorsque la sortie du registre =1, le signal est égale a S_{max} pendant une même période, il est alimenté par la somme de 2 bits i et j, le numéro de ces bits dépend de la longueur de ce registre. Le principe est expliqué par la figure (I.4).

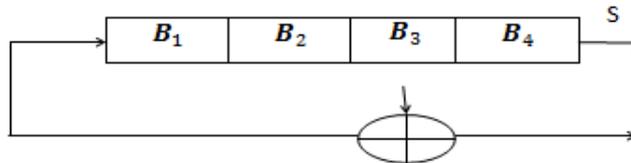


Figure (I.4): Génération d’une séquence binaire pseudo –aléatoire

La longueur d'une séquence est $2^N - 1$ bits, tels que N est le nombre de cellules du registre à décalage. Sa durée est $(2^N - 1)T_{SBPA}$.

Le tableau suivant définit les bits d'alimentation utilisés pour générer une SBPA pour différentes tailles du registre N.

Nombre de cellules N	Longueur $L= 2^N - 1$	Bits additionnés
2	3	1 et 2
3	7	2 et 3
4	15	3 et 4
5	31	3 et 5
6	63	5 et 6
7	127	4 et 7
8	255	2,3,4 et 8
9	511	5 et 9
10	1023	7 et 10

Tableau (I.2) : Génération des SBPA de longueur maximale

I.3.1.3 Exemple de génération d'une SBPA

D'après le principe expliqué par la figure (I.4), on a pour un registre à 4 cellules c'est-à-dire $N = 4$ et la taille du signal est $L = 2^N - 1 = 15$. D'après le tableau les bits d'alimentation sont 3 et 4. On choisit une valeur maximale pour la sortie du registre $S_{max} = 220v$ et une valeur minimale $S_{min} = -220v$ et pour une période $T_{SBPA} = 1s$.

Les différentes étapes sont résumées dans le tableau suivant :

T	b_1	b_2	b_3	b_4	S	SBPA
$0T_{SBPA}$	1	1	1	1	0	S_{min}
$1T_{SBPA}$	0	1	1	1	0	S_{min}
$2T_{SBPA}$	0	0	1	1	0	S_{min}
$3T_{SBPA}$	0	0	0	1	1	S_{max}
$4T_{SBPA}$	1	0	0	0	0	S_{min}
$5T_{SBPA}$	0	1	0	0	0	S_{min}
$6T_{SBPA}$	0	0	1	0	1	S_{max}
$7T_{SBPA}$	1	0	0	1	1	S_{max}
$8T_{SBPA}$	1	1	0	0	0	S_{min}
·						
·						
$15T_{SBPA}$						

Tableau (I.3) : Tableau de génération d'une SBPA

La figure (I.5) suivante représente le signal SBPA avec 4 bits :

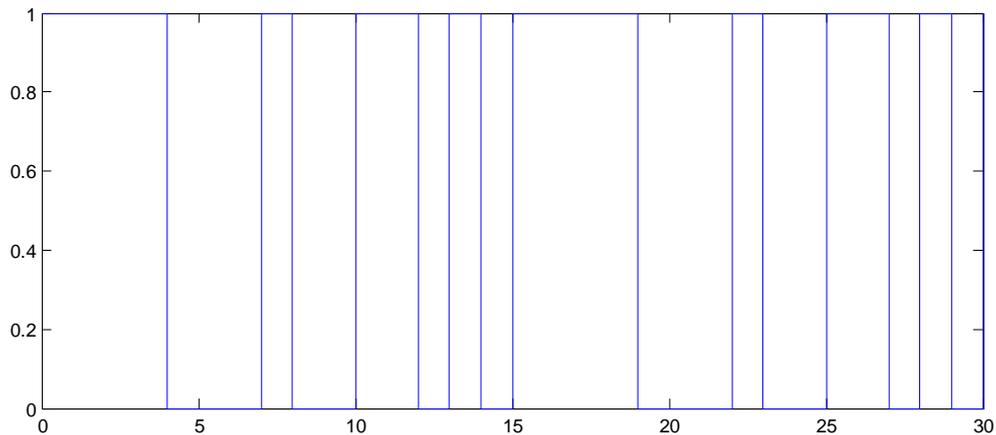


Figure (I.5): signal SBPA généré à l'aide d'un registre à 4 bits

I.3.2 Choix de la structure du modèle

La deuxième étape de l'identification est le choix de la structure du modèle, c'est à dire de choisir l'ordre du dénominateur et du numérateur de la fonction de transfert.

Cette étape est réalisée à partir d'une expérience physique. On injecte une entrée échelon au système réel qu'on veut identifier, et à partir de l'allure de la réponse indicielle, on peut choisir une structure pour le modèle.

Les figures suivantes présentent les réponses indicielles obtenues pour des systèmes d'ordre différents :

- **Le modèle du premier ordre est représenté sous forme**

$$G(s) = \frac{K}{1+Ts} \quad (\text{I.1})$$

{ *k*: est le gain statique
{ *T*: est la constante du temps

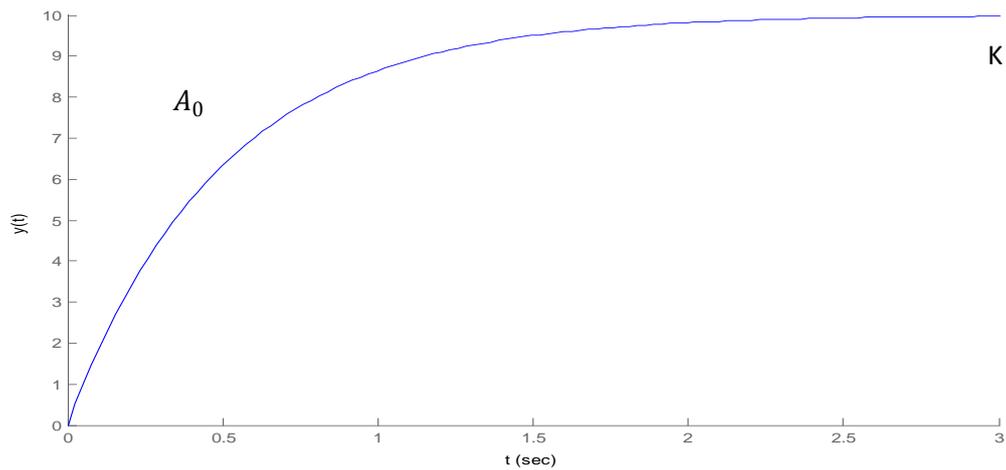


Figure (I.6) : réponse indicielle d'un procédé du premier ordre.

- Le modèle du premier ordre retardé est donné sous forme

$$G(s) = \frac{K}{1+Ts} e^{-\tau s} \tag{I.2}$$

$\left\{ \begin{array}{l} K: \text{est le gain statique} \\ T: \text{la constante du temps} \\ \tau: \text{le retard} \end{array} \right.$

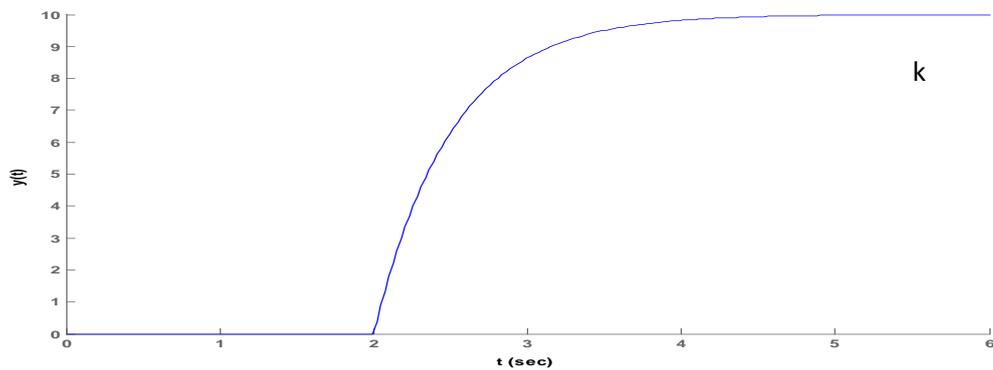


Figure (I.7) : réponse indicielle d'un procédé du premier ordre retardé.

- Cette figure représente le modèle du deuxième ordre non oscillé qui est donné sous forme :

$$G(s) = \frac{K}{(s+a)(s+b)} \tag{I.3}$$

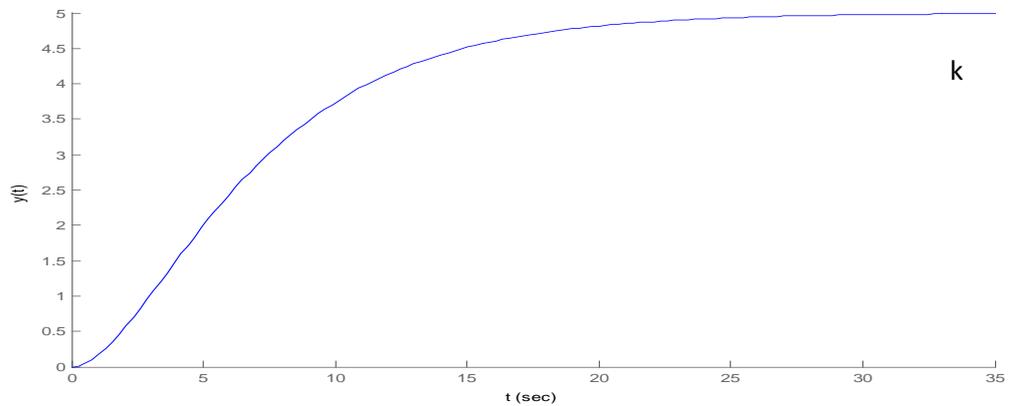


Figure (I.8) : Réponse indicielle d'un procédé du deuxième ordre non oscillatoire.

- Le modèle de deuxième ordre non oscillatoire avec retard est représenté sous forme

$$G(s) = \frac{K}{(s+a)(s+b)} e^{-\tau s} \tag{I.4}$$

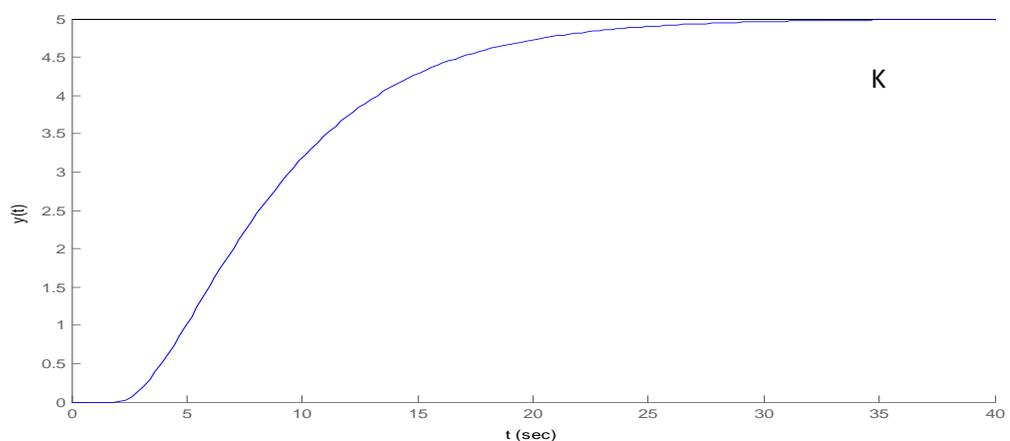


Figure (I.9) : réponse indicielle d'un procédé du deuxième ordre non oscillatoire retardé.

- La forme de représentation du modèle du deuxième ordre est donnée sous cette écriture :

$$G(s) = \frac{Kw_n^2}{s^2 + 2\varepsilon w_n s + w_n^2} \tag{I.5}$$

Où $\left\{ \begin{array}{l} K: \text{est le gain statique} \\ w_n: \text{est la pulsation naturelle non amortie} \\ \varepsilon: \text{facture damortissement} \end{array} \right.$

-

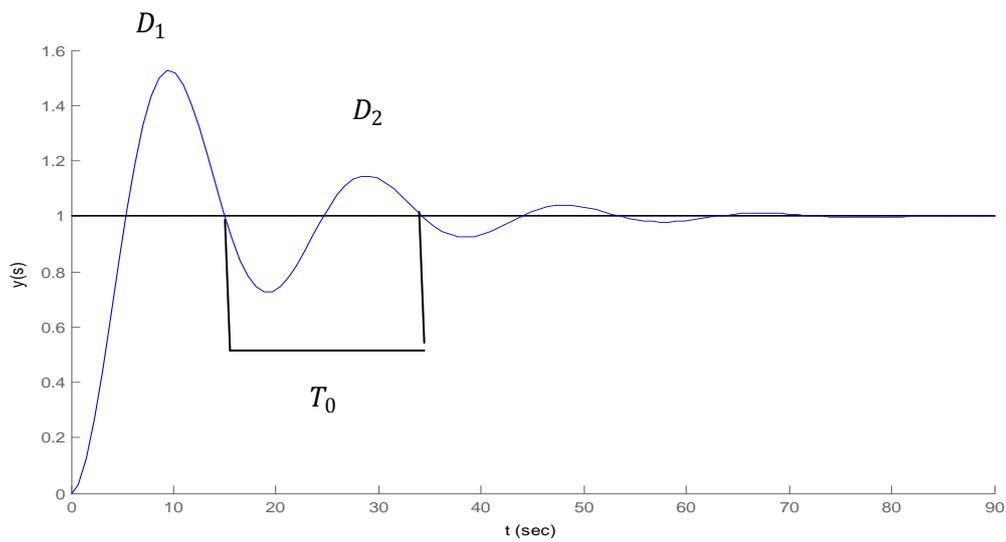


Figure (10) : Réponse indicielle d'un procédé du deuxième ordre oscillatoire

- La forme de représentation du modèle du deuxième ordre oscillatoire retardé est donnée :

$$G(s) = \frac{Kw_n^2}{s^2 + 2\varepsilon w_n s + w_n^2} e^{-\tau s} \tag{I.6}$$

k : le gain statique
 w_n la pulsation naturelle non amortie
 ε le facteur d'amortissement
 τ le retard

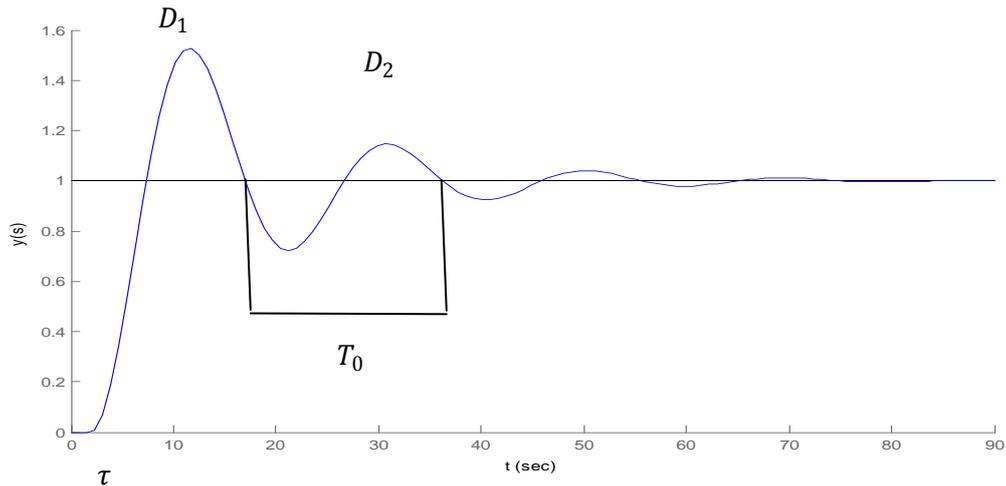


Figure (I.11) : réponse indicielle d'un procédé du deuxième ordre oscillatoire retardé.

I.3.3 Choix de la méthode d'identification :

Pour les signaux d'excitation possédant de faibles amplitudes, la sortie mesurée sera accompagnée d'un bruit [2]. Ce dernier introduit des biais (erreur d'estimation paramétrique) et généralement il est très difficile de caractériser le type du bruit, d'où la nécessité d'introduire à chaque fois une hypothèse sur ces structures puis de choisir un algorithme d'estimation approprié.

I.3.4 La validation du modèle

L'étape finale de l'identification consiste à vérifier si le modèle trouvé peut représenter le système réel. Pour cela, on fixe une tolérance et on calcule l'erreur entre la réponse du système réel et la réponse du système identifié qui sont soumis à la même entrée. Si l'erreur calculée est inférieure à la tolérance qu'on a fixée on valide le modèle, sinon on doit refaire l'identification à partir de la première étape.

I.4 Les méthodes d'identification :

Plusieurs méthodes peuvent être utilisées pour la détermination des paramètres. Ces méthodes sont classées par des catégories telles qu'ils sont représentés dans ce tableau (I.4).

Types de Méthode	Méthodes
Non paramétriques	<ul style="list-style-type: none"> ➤ Méthode basée sur les réponses fréquentielles, ➤ Méthode basées sur des réponses impulsionnelles.
paramétriques	<ul style="list-style-type: none"> ➤ Graphique <ul style="list-style-type: none"> - méthode basée sur des réponses temporelles (Strejc et Broida), - méthode pour des systèmes avec intégration, - méthode des systèmes oscillatoires. ➤ Numérique <ul style="list-style-type: none"> - méthode des moindres carrées simples, - méthode des moindres carrées récursifs, - méthode du modèle.

Tableau (I.4) : les différentes méthodes d'identification

Dans notre mémoire, on s'intéresse aux méthodes paramétriques.

I.4.1 Méthodes graphiques

Cette méthode est basée sur les réponses des graphes dont on peut identifier les paramètres des systèmes, le plus souvent ses réponses sont issues à partir d'un échelon, elles sont utilisées pour identifier des modèles à structure simple, elles ne permettent pas d'avoir des résultats bien précis mais elles donnent seulement des renseignements intéressants sur la dynamique de ce systèmes [2].

On choisit le modèle et la méthode associée par rapport au type de la structure du modèle si le processus est stable ou contient un intégrateur, s'il est retardé ou non, s'il possède un dépassement et s'il est apériodique.

Parmi ces méthodes, on peut citer :

I.4.1.1 méthodes basées sur les réponses temporelles :

➤ Un système du premier ordre sans retard :

Le modèle du premier ordre est donné sous cette forme :

$$G(s) = \frac{k}{1+Ts} \quad (\text{I.7})$$

On a deux paramètres à identifier :

k : est le gain statique, on peut le déduire à partir de graphe de la figure (I.11)

$$k = y(\infty) \quad (\text{I.8})$$

T : est la constante de temps, après la détermination de k , on peut identifier T qui représente le temps de résidence moyen.

$$T = T_{rm} = \frac{A_0}{K} \quad (\text{I.9})$$

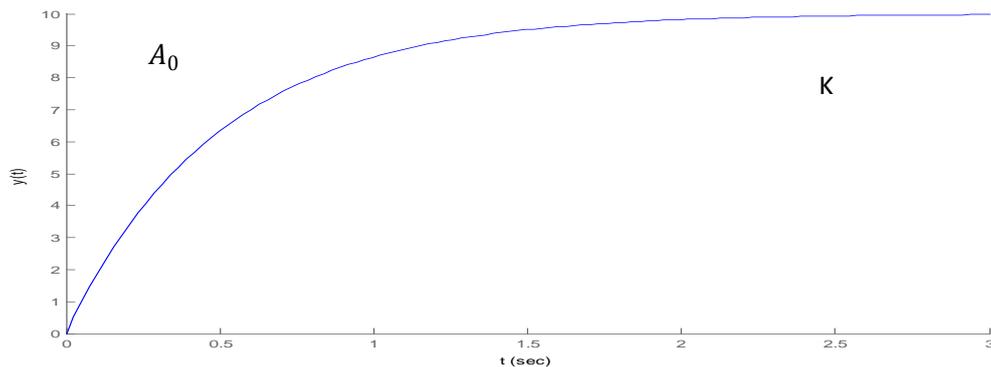


Figure (I.12) : Réponse d'un système du premier ordre à un échelon

Tel que A_0 est la surface située entre la droite K , et la réponse y et l'axe des ordonnées :

$$A_0 = \int_0^{\infty} (y(\infty) - y(t)) dt \quad (\text{I.10})$$

➤ **Système intégrateur :**

La forme du modèle dans un système intégrateur est :

$$G(s) = \frac{\alpha}{\tau s} e^{-\tau s} \tag{I.11}$$

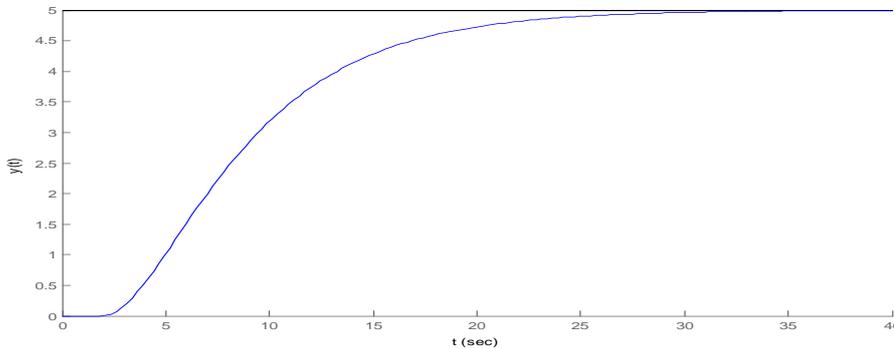


Figure (I.13) : réponse d'un système intégrateur à un échelon

Pour identifier les deux paramètres α et τ , on trace la tangente à la courbe :

τ : est l'intersection de la tangente avec l'axe des abscisses

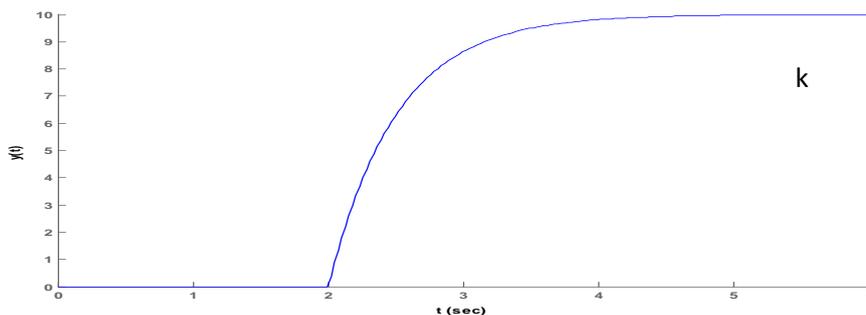
α : est l'intersection de la tangente avec l'axe des ordonnées.

➤ **Système de 1 er ordre avec retard**

Le modèle à trois paramètres est:

$$G(S) = \frac{K}{1+T_s} e^{-\tau s} \tag{I.12}$$

D'après cette figure la réponse de ce système est représentée :



Figure(I.14) : Réponse d'un système

du premier ordre retardé à un échelon

On a trois paramètres à identifier :

k: Le gain statique, on le déduit à partir du graphe $k = y(\infty)$.

τ: Le retard

T : La constante de temps tel que :

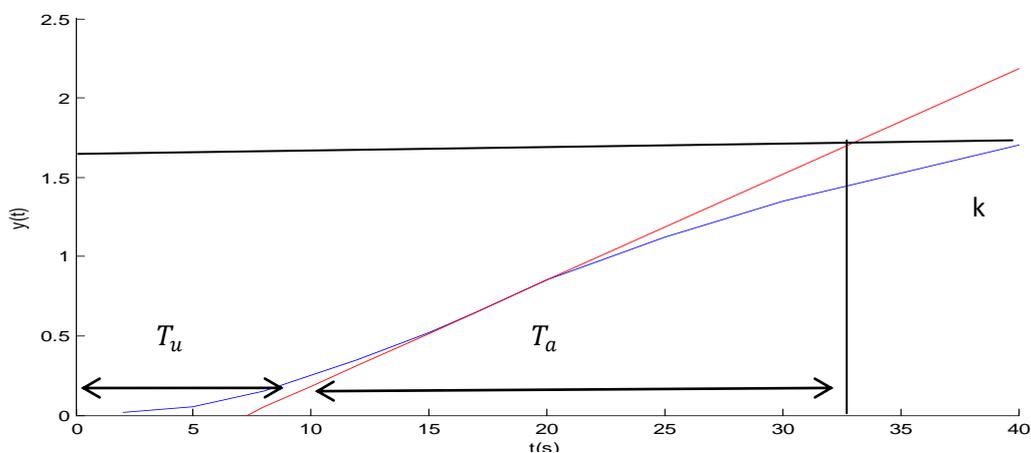
$$T = T_{rm} - \tau \tag{I.13}$$

Où : T_{rm} est le temps de résistance moyenne, son expression est donnée par :

$$T_{rm} = \frac{\int_0^{\infty} (y(\infty) - y(t)) dt}{k} \tag{I.14}$$

➤ **Méthode de strejc-Davoust :**

Cette méthode s'applique aux systèmes dont la réponse indicielle ne présente pas un dépassement, on identifie à une fonction de la forme :



Figure(I.15) : la méthode du strejc

Les paramètres à identifier sont :

k : Le gain statique, on le déduit à partir du graphe,

τ : Le retard,

T : La constante de temps tel que,

n : L'ordre du système.

Pour identifier ces paramètres on suit les étapes suivantes :

- 1- k : Est la valeur finale de la sortie il est mesuré directement à partir du graphe.

$$k = y(\infty).$$

Pour définir les valeurs T_u et T_a on doit tracer la tangente au point d'inflexion.

- 2- On calcule le rapport T_u/T_a pour trouver les valeurs de τ et T , à partir de tableau (I.5)

on cherche la valeur la plus proche (inferieure) au rapport calculé précédemment pour identifier l'ordre n de système.

La constante de temps T se calcule comme suit :

$$T = \frac{T}{T_a} \Big|_{table} * T_{a, mesurée} \quad (I.15)$$

Et le retard τ :

$$\tau = T_{u,mesurée} - \frac{T}{T_a} \Big|_{table} * T_{a, mesurée} \quad (I.16)$$

Le tableau suivant représente les coefficients de la méthode de strejc :

T_u/T_a	Ordre du modèle (n)	T/T_a
0	1	1
0.125	2	0.37
0.220	3	0.27
0.320	4	0.22
0.410	5	0.20
0.490	6	0.18
0.570	7	0.19
0.640	8	0.15
0.71	9	0.14

Tableau (I.5) : Coefficients de la méthode de Strejc

➤ Méthode de Broida

Cette méthode identifie un système de premier ordre avec un retard pur, elle a pour principe d'ajuster le paramètre T et pour que les réponses du modèle et du processus aient deux points communs 1 et c_2 , ces deux points correspondent respectivement à 28% et 40 % de la valeur finale.

Sa fonction de transfert est sous la forme :

$$G(s) = \frac{K}{1+Ts} e^{-\tau s} \quad (\text{I.17})$$

Avec

$$T = 2.8t_{28\%} - 1.8t_{18\%} \quad (\text{I.18})$$

$$\tau = 5.5(t_{40\%} - t_{28\%}) \quad (\text{I.19})$$

I.4.1.2 Méthode des systèmes oscillatoires

Dans ce cas la réponse indicielle du système présente un caractère oscillant, le modèle est de deuxième ordre donné par :

$$G(s) = \frac{Kw_n^2}{s^2 + 2\varepsilon w_n s + w_n^2} e^{-\tau s} \tag{I.20}$$

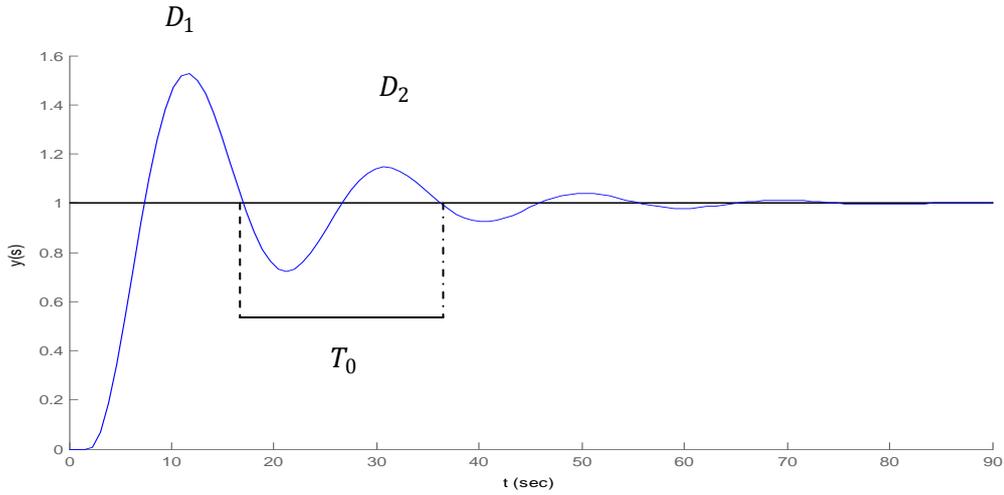


Figure (I.16): la réponse d'un système de deuxième ordre oscillatoire

Les paramètres à identifier sont :

w_n : Pulsation naturelle non amortie,

ε : Facteur d'amortissement,

k : est le gain statique,

τ : Le retard.

En utilisant le graphe, on mesure les valeurs du premier et du deuxième dépassement D_1 et D_2 qui représentent la différence entre le premier (deuxième) maximum de la courbe et la valeur finale de la sortie et la valeur de la période des oscillations est T_0 .

Puis on calcule les coefficients de la fonction de transfert

$$w_n = \frac{2\pi}{T_0\sqrt{1-\varepsilon^2}} \tag{I.21}$$

$$\frac{D_1}{D_2} = e^{-\frac{2\pi\varepsilon}{\sqrt{1-\varepsilon^2}}} \tag{I.22}$$

$$k = y(\infty) \tag{I.23}$$

I.4.2 Méthode numériques

Bien que des fonctions et équations rencontrées en pratique ou en théorie ne peuvent pas être résolues exactement par des méthodes analytiques donc on cherche une solution numérique approchée peut être obtenue en un nombre fini d'opération, alors après l'étude graphique on s'intéresse aux méthodes numériques basées sur l'utilisation des algorithmes pour bien trouver les paramètres de notre modèle.

I.4.2.1 Méthode des moindres carrés simples

Cette méthode est élaborée par « Legendre et Gauss » permet de comparer des données expérimentales entachées d'erreurs de mesure à un modèle mathématique censé décrire ces données

Le principe général de cette méthode est de choisir les paramètres d'un modèle que l'on définira, de telle sorte qu'il minimise la somme des carrés de la différence entre les valeurs prédites par le modèle et les valeurs observées.

Cette méthode peut s'appliquer à n'importe quel type de modèle mais analytiquement elle ne pourra être appliquée que si le modèle est linéaire avec les paramètres [2].

➤ L'application de la méthode des moindres carrés simples :

Pour trouver les paramètres on doit d'abord commencer par la formulation standard des moindres carrés en supposant que la variable calculer

$\hat{y}(x)$ est donné par le modèle suivant :

$$\hat{y}(x) = \theta_1 \phi_1(x) + \theta_2 \phi_2(x) + \dots + \theta_n \phi_n(x) \quad (\text{I. 24})$$

Où :

$\theta_1 \theta_2 \dots \theta_n$: Ce sont des paramètres à déterminer

$\phi_1 \phi_2 \dots \phi_n$: Ce sont des fonctions connues,

$x_1 \dots x_n$: Ce sont des variables explicatives,

N : Le nombre de mesures.

On cherche à trouver les paramètres du modèle de telle sorte que les valeurs \hat{y}_i calculées à partir des variables x_i soient très proches que possible des valeurs mesurées y_i et Dans le cas où la précision est la même pour toutes les mesures, les paramètres doivent vérifier la minimisation des critères suivant :

$$j(\theta) = \frac{1}{2} \sum_{i=1}^N e_i^2, \quad e_i = y_i - \hat{y}_i \quad (\text{I.25})$$

Le modèle identifié est donné sous forme :

$$G(z^{-1}) = \frac{b_0 + b_1 z^{-1} + \dots + b_m z^{-m}}{a_0 + a_1 z^{-1} + \dots + a_n z^{-n}} = \frac{y(z)}{u(z)} \quad (\text{I.26})$$

Pour $a_0 = 1$, l'équation aux différences correspondante sera :

$$y(z) + a_1 y(z)z^{-1} + a_2 y(z)z^{-2} \dots + a_n z^{-n} = b_0 u(z) + \dots + b_m u(z)z^{-m} \quad (\text{I.27})$$

$$y(k) = -a_1 y(k-1) - a_2 y(k-2) - \dots - a_n y(k-n) + b_0 u(k) + \dots + b_m u(k-m) \quad (\text{I.28})$$

$$y(k+1) = -a_1 y(k) - a_2 y(k-1) - \dots - a_n y(k-n+1) + b_0 u(k+1) + \dots + b_m u(k-m+1) \quad (\text{I.29})$$

On applique une séquence d'entrée $\{u(1), u(2), \dots, u(N)\}$ au système et on récupère la séquence de la variable mesurée correspondante $\{y(1), y(2), \dots, y(N)\}$

Les paramètres inconnus sont regroupés dans le vecteur suivant :

$$\theta = [a_1 \dots a_0 \dots b_0 \dots b_m]^T \quad (\text{I.30})$$

En introduisant le vecteur de régression $\phi(k+1)$ tel que :

$$\phi(k+1) = [-y(k) \quad -y(k-1) \quad \dots \quad -y(k-n+1) \quad u(k-1) \quad \dots \quad u(k-m+1)] \quad (\text{I.31})$$

Le modèle dynamique s'écrit sous forme :

$$y(k+1) = \phi(k+1)\theta \quad (\text{I.32})$$

Dans le cadre des moindres carrés, les mesures $y(k) \dots y(k-n+1), u(k) \dots u(k-m+1)$ sont utilisées pour calculer (ou encore prédire) $y(k+1)$:

$$\hat{y}(k+1) = \phi(k+1)\theta \quad (\text{I.33})$$

En considérant les points de 1 à N, on a (on suppose $n > m$ pour simplifier l'écriture) :

$$\hat{y}(n + 1) = -a_1y(n) - a_2y(n - 1) - \dots - a_ny(1) + b_0u(n) + \dots + b_mu(k - m + 1) \tag{I.34}$$

$$\hat{y}(n + 2) = -a_1y(n + 1) - a_2y(n) - \dots - a_ny(2) + b_0u(n + 1) + \dots + b_mu(k - m + 2) \tag{I.35}$$

·
·
·

$$\hat{y}(N) = -a_1y(N - 1) - a_2y(N - 2) - \dots - a_Ny(N - 1 - n) + b_0u(N + 1) + \dots + b_mu(N - m) \tag{I.36}$$

$$\begin{bmatrix} \hat{y}(n + 1) \\ \hat{y}(n + 2) \\ \vdots \\ \hat{y}(n) \end{bmatrix} = \begin{bmatrix} -y(n) & -y(n - 1) & -y(n) & \dots & -y(1) & u(n + 1) & \dots & u(n - m + 1) \\ \dots & -y(n + 1) & -y(n) & \dots & -y(2) & \dots & u(n + 2) & \dots & u(n - m + 2) \\ \dots & \dots \\ -y(N - 1) & -y(N - 2) & \dots & -y(N - 1 - n) & \dots & u(N + 1) & \dots & u(N - m) \end{bmatrix} \times \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \\ b_0 \\ \vdots \\ b_m \end{bmatrix} = \theta_N \tag{I.37}$$

$$\rightarrow y_N = \Phi_N \theta_N \tag{I.38}$$

$$y_N = \begin{bmatrix} \Phi(n + 1) \\ \Phi(n + 2) \\ \vdots \\ \Phi(N) \end{bmatrix} \theta_N \rightarrow y_N = \Phi_N \theta_N \tag{I.39}$$

Remarque :

La matrice $(\Phi_N^T \Phi_N)^{-1} \Phi_N^T$ est une matrice le pseudo inverse de Φ_N .

I.4.2.2 Méthodes des moindres carrés récursifs :

Avec la méthodes des moindres carrés non récursif, si une nouvelle mesure se présente, on doit recommencer entièrement le calcul, c'est pour cela qu'on a fait appel aux méthodes récursives qui permet de calculer, a chaque nouvel instant, les nouvelles valeurs des vecteurs de paramètres d'un modèle

Soit à calculer θ_{N+1} l'estimation des paramètres a (N+1), on à les deux vecteurs suivant :

$$y_N = [Y(1) \ Y(2) \ \dots \ Y(N)]^T \quad (\text{I.40})$$

$$\phi_N = [\phi(1) \ \phi(2) \ \dots \ \phi(N)]^T \quad (\text{I.41})$$

On utilisant la relation non récursive prise sur une durée N donnée, on a :

$$\theta_N = [\theta_N^T \theta_N]^{-1} \phi_N^T y_N \quad (\text{I.42})$$

Ainsi pour un horizon à N+1 mesures, on aura la relation suivantes :

$$\theta_{N+1} = [\phi_{N+1}^T \phi_{N+1}]^{-1} \phi_{N+1}^T Y_{N+1} \quad (\text{I.43})$$

Avec :

$$Y_{N+1} = [y_N \quad y(N+1)]^T \quad (\text{I.44})$$

$$\phi_{(N+1)} = [\phi_N \quad \phi(N+1)]^T \quad (\text{I.45})$$

Afin de simplifier l'écriture, on introduit par la suite la matrice p_{N+1} tel que :

$$p_{(N+1)} = [\phi_{N+1}^T \phi_{N+1}]^{-1} \quad (\text{I.46})$$

Ce qui donne :

$$\begin{aligned} p_{N+1}^{-1} &= [\phi_N^T \phi^T(N+1)] [\phi_N \quad \phi(N+1)]^T \\ &= \phi_N^T \phi_N + \phi^T(N+1) \phi(N+1) \end{aligned} \quad (\text{I.47})$$

Si on pose :

$$p_{N+1}^{-1} = \phi_N^T \phi_N \quad (\text{I.48})$$

On aura la relation suivante :

$$[p_{N+1}^{-1}]^{-1} = [p_N^{-1} + \phi^T(N+1) \phi(N+1)]^{-1} \quad (\text{I.49})$$

On obtient

$$p_{N+1} = p_N - \frac{p_N \phi^T(N+1) \phi(N+1) p_N}{1 + \phi(N+1) p_N \phi^T(N+1)} \quad (\text{I.50})$$

Pour le calcul de θ_{N+1} , on a :

$$\begin{aligned}\theta_{N+1} &= p_{N+1}[\phi_{N+1}^T y_{N+1}] = p_{N+1}[\phi_N^T \phi^T(N+1)][y_N \quad y(N+1)]^T \\ &= p_{N+1}[\phi_N^T y_N + \phi^T(N+1)y(N+1)]\end{aligned}\quad (\text{I.51})$$

Avec :

$$\phi_N = p_N \phi_N^T y_N \quad (\text{I.52})$$

On peut écrire alors :

$$p_N^{-1} \theta_N = \phi_N^T y_N \quad (\text{I.53})$$

D'où :

$$\theta_{N+1} = p_{N+1}[p_N^{-1} \theta_N + \phi^T(N+1)y(N+1)] \quad (\text{I.54})$$

Remplaçons p_N^{-1} par ce qu'on a trouvé précédemment, on aura :

$$\begin{aligned}\theta_{N+1} &= p_{N+1}[p_{N+1}^{-1} - \phi^T(N+1)\phi(N+1)]\theta_N + \phi^T(N+1)y(N+1) \\ &= \theta_N - p_{N+1}\phi^T(N+1)\theta_N + p_{N+1}\phi^T(N+1)y(N+1) \\ &\quad \theta_N + p_{N+1}\phi^T(N+1)[y(N+1) - \phi(N+1)\theta_N]\end{aligned}\quad (\text{I.55})$$

Donc on obtient la relation suivante :

$$\theta_{N+1} = \theta_N + k_{N+1}[y(N+1) - \phi(N+1)\theta_N] \quad (\text{I.56})$$

D'où :

θ_{N+1} : nouvelle estimation des paramètres
 θ_N : Estimation précédente des paramètres
 k_{N+1} : Gain d'adaptation

Avec :

$$k_{N+1} = p_{N+1}\phi^T(N+1) \quad (\text{I.57})$$

$y(N+1)$: vecteur des dérivations à $N+1$

$\phi(N+1)\theta_N$: vecteur des dérivations précédente

I.4.3 Méthode du modèle

La méthode du modèle est l'une des méthodes d'identification les plus efficaces, appliquée aux systèmes non linéaire.

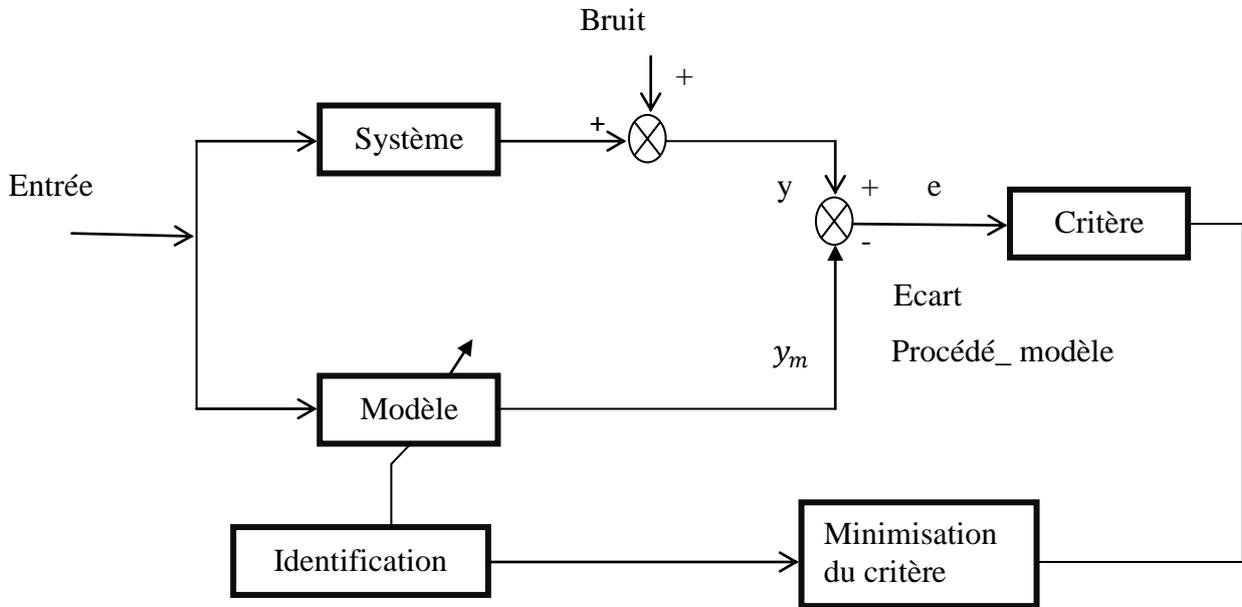


Figure (I.17) : schéma synoptique de la méthode du modèle

Son principe est de proposer une structure mathématique pour le modèle à partir des résultats expérimentaux, et de transformer ce problème à un problème d'optimisation pour minimiser l'écart trouvé entre les mesures expérimentales $y(k)$ et celles prédites par le modèle $y_m(k)$. De telle sorte d'avoir cet écart très proche de zéro.

$$e(k) = y(k) - y_m(k).$$

Avec :

$$y_m(k) = f(\theta, y_m(k - 1), y_m(k - 2), \dots, y_m(k - n), u(k - 2) \dots u(k - m)) \tag{I.58}$$

Donc cette méthode consiste à déterminer les paramètres du modèle θ qui représentent les variables de décision de telle sorte que l'erreur modèle- procédé soit minimale.

Ce problème d'optimisation peut être formulé comme un problème de minimisation du critère suivant :

$$J(\theta) = \frac{1}{2} \sum_{k=1}^N e(k)^2 = \frac{1}{2} \sum_{k=1}^N [y(k) - y_m(k)]^2 \quad (\text{I.60})$$

Conclusion

Ce chapitre présente différentes méthodes pour calculer les paramètres d'un modèle dynamique de comportement, nous les avons classées en plusieurs catégories : les méthodes graphiques basées sur une réponse indicielle qui ne permettent d'avoir des résultats bien précis mais donnent les informations sur la dynamique de ce système, les méthodes basées sur les moindres carrés ne sont pas les meilleurs mais elle peuvent donner une solution analytique, la méthode du modèle permet d'atteindre de façon directe l'objectif nous avons défini pour l'identification. Cette méthode est très utilisée en pratique car le problème d'identification est formulé d'un problème d'optimisation dont les paramètres à identifier par la méthode du modèle en utilisant une technique d'optimisation globale appelé Aliénor.

II.1 Introduction :

Dans le but d'identifier les paramètres qui convergent vers les paramètres réels du système l'étape de l'optimisation est primordiale, car les performances dépendent de la solution du problème d'optimisation. Elles sont meilleures si l'optimum global du critère est atteint. Pour cela plusieurs méthodes sont utilisées (méthodes locales et globales).

Dans ce chapitre on présente les notions de l'optimisation de manière générale et les différentes méthodes d'optimisation local et globale, avec ou sans contraintes, à la fin on s'intéresse à l'étude de la méthode d'Aliéner qui est considéré comme une méthode globale.

II.2 Formulation d'un problème d'optimisation :**II.2.1 Cas de minimisation :**

Le problème décrit ci-dessus est un problème de minimisation :

$$\text{Sujet à : } \begin{cases} \min_x f(x) \\ g_i(x) = 0, i = 1, \dots, p \\ h_j \leq 0, j = 0, j = 1, \dots, q \end{cases} \quad (\text{II.61})$$

Minimiser une fonction revient à chercher son minimum sur un domaine admissible.

II.2.2 Dans le cas de maximisation :

Le problème décrit ci-dessous est un problème de *maximisation* :

$$\text{Sujet à : } \begin{cases} \max_x f(x) \\ g_i(x) = 0, i = 1, \dots, p \\ h_j \leq 0, j = 0, j = 1, \dots, q \end{cases} \quad (\text{II.62})$$

Maximiser une fonction revient à chercher son maximum sur un domaine admissible.

Remarque :

Pour chercher le maximum d'une fonction, on peut appliquer la relation suivante :

$$\min f(x) = - \max[-f(x)] \tag{II.63}$$

Le problème de *maximisation* de la fonction f (à gauche ci-dessus) est *équivalent* au problème de *minimisation* de $-f$. L'équivalence veut dire ici que les solutions sont les mêmes et que les valeurs optimales sont opposées. En particulier, une méthode pour analyser/résoudre un problème de minimisation pourra être utilisée pour analyser/résoudre un problème de maximisation.

II.3 Classification des problèmes d'optimisation

Un problème d'optimisation peut être classé suivant la nature de la fonction objectif et la nature des contraintes. Les cas les plus courants sont résumés dans le tableau suivant [3] :

		Fonction objectif		
		Linéaire	Quadratique (convexe)	Non linéaire
Contrainte	Linéaire	Programmation Linéaire	Programmation quadratique	Programmation Non linéaire
	Convexe	Programmation Non linéaire	Programmation convexe	Programmation Non linéaire
	Non Linéaire	Programmation Non linéaire	Programmation Non linéaire	Programmation Non linéaire

Tableau (II.6): Classification des problèmes d'optimisation

- **Programmation linéaire :**

Il s'agit d'une classe de problèmes d'optimisation où la fonction objectif ou le critère est linéaire

$$f(x) = C^T x$$

(C est un vecteur des coefficients) et où l'ensemble des contraintes est un polyèdre convexe fermé (représenté sous la forme $x \leq b$, avec A et b sont respectivement une matrice et un vecteur constants)

- **Programmation quadratique :**

Il s'agit d'un problème d'optimisation où la fonction objectif est quadratique convexe, c'est-à-dire :

$$f(x) = \frac{1}{2} x^T A x + C^T x \quad (\text{II.64})$$

Avec A est une matrice symétrique semi-définie positive et l'ensemble de contrainte est toujours un polyèdre convexe fermé comme dans le cas de la programmation linéaire.

- **Programmation convexe :**

Ici la fonction objectif est convexe, et l'ensemble admissible D est convexe. Le cas de la programmation quadratique est un cas particulier de la programmation convexe.

Un domaine D est convexe si :

$$\forall (x_1, x_2) \in D \text{ et } 0 \leq \alpha \leq 1 \rightarrow x = \alpha x_1 + (1 - \alpha)x_2 \in D \quad (\text{II.65})$$

Une fonction $f(x)$ est convexe dans un domaine convexe D si :

$$\forall (x_1, x_2) \in D \text{ et } 0 \leq \alpha \leq 1 \rightarrow f(\alpha x_1 + (1 - \alpha)x_2) \leq \alpha f(x_1) + (1 - \alpha)f(x_2) \quad (\text{II.66})$$

- **Programmation non linéaire :**

Toutes les données du problème d'optimisation sont des fonctions différentiable (Continument différentiable), c'est-à-dire la fonction objectif est différentiable

II.4 Condition d'optimalité :

Pour que x soit minimum de f , on doit vérifier certains conditions d'optimalité nécessaires suivantes :

II.4.1 Condition 1^{er} ordre :

Supposons que $f(x)$ est continue et dérivable sur le domaine D_a , et x^* la

Solution de problème. La condition nécessaire dans ce cas consiste l'annulation de gradient en ce point :

$$\nabla_x f(x^*) = 0 \tag{II.67}$$

Le gradient se calcule comme suit :

$$\nabla_x f(x) = \left[\frac{\partial f(x)}{\partial x_1} \quad \frac{\partial f(x)}{\partial x_2} \quad \dots \quad \frac{\partial f(x)}{\partial x_n} \right] \tag{II.68}$$

II.4.2 Condition 2^{ème} ordre :

Soit $f(x)$ une fonction deux fois continument dérivable. La condition suffisante du minimum est d'avoir la matrice hessienne définie positive.

$$\nabla_x^2 f(x) = \begin{bmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_2} & \dots & \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(x)}{\partial x_2 \partial x_1} & \frac{\partial^2 f(x)}{\partial x_2^2} & \dots & \frac{\partial^2 f(x)}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f(x)}{\partial x_n \partial x_1} & \frac{\partial^2 f(x)}{\partial x_n \partial x_2} & \dots & \frac{\partial^2 f(x)}{\partial x_n^2} \end{bmatrix} \tag{II.69}$$

Remarque :

Une matrice est dite positive si l'une des conditions suivantes est satisfaite :

- Les valeurs propres de la matrice sont toutes positives.
- Les déterminants de la matrice pris en chaîne sont tous positifs.

La matrice **Héssienne** nous permet d'étudier la nature des points critiques

Le tableau suivant résume les différents cas possible:

Matrice	Commentaire	Nature du point critique
$\nabla_x f(x^*) > 0$	Définie positive	Minimum
$\nabla_x f(x^*) < 0$	Définie négative	Maximum
$\nabla_x f(x^*) \geq 0$	Semi-définie positive	Point singulier (on ne peut pas préciser sa nature)
$\nabla_x f(x^*) \leq 0$	Semi-définie négative	Point singulier
$\nabla_x f(x^*) = 0$	Possède des valeurs propres négatives et positives	Point selle

Tableau(II.7) : Nature des points critiques

II.5 Méthodes de résolution des problèmes d'optimisation :

De nombreuses méthodes d'optimisation peuvent être appliquées pour résoudre un problème d'optimisation avec ou sans contraintes tout dépend de la fonction objectif et les contraintes.

On peut obtenir une solution d'un problème d'optimisation graphiquement si la fonction objectif est linéaire (la programmation linéaire), ou bien analytiquement si la fonction objectif est convexe ou quadratique (non linéaire). Dans le cas où la fonction objectif à minimiser est fortement non linéaire ou elle est linéaire mais elle dépasse trois variables, on fait appel aux méthodes numériques.

La présence des contraintes dans un problème d'optimisation nécessite la transformation de ce problème à un problème sans contraintes [3], on a recours à la méthode des multiplicateurs de Lagrange dans le cas de la présence des contraintes égalités et à la méthode de

Kuhn-Tucker dans le cas des contraintes inégalités, cette dernière consiste l'ajout des coefficients de pénalisations.

II.5.1 Méthodes numériques

Nous trouvons de nombreuses méthodes d'optimisation sans contraintes.

Nous pouvons les diviser en deux grandes catégories : les méthodes locales, qui cherchent à converger vers l'optimum le plus proche de la solution courante, en utilisant des méthodes d'explorations dans son voisinage. le but de l'optimisation locale d'une fonction à partir d'un élément $x_0 \in D_a$ est de trouver l'élément $x^* \in D_a$ proche de x_0 qui minimise la fonction $f(x)$ parmi les éléments appartenant au voisinage de x_0 , comme l'indique la relation suivante :

$$\begin{cases} \text{chercher } x^* \in D_a \\ \exists V \in V_{x_0} / x^* \in V \text{ et } f(x^*) = \min_{x \in V} f(x) \end{cases} \quad (\text{II.70})$$

On appelle x^* le minimiseur local de f et $f(x^*)$ le minimum local de f autour de x_0 .

La deuxième catégorie est les méthodes globales, qui cherchent à faire converger la solution vers l'optimum global de la fonction.

La méthode de recherche locale la plus célèbre est la descente de gradient, elle génère une suite $(x_k, k \in N)$, qui converge vers un minimum locale x^* de la fonction f suivant le traitement des contraintes utilisées, son inconvénient principale est que la dérivée de la fonction f n'est pas toujours connue alors cette recherche numérique exige l'utilisation des méthodes itératives tel que à partir d'une valeur initiale $x^{(0)}$ on calcule une suite des valeurs $x^{(1)} x^{(2)} \dots x^{(k)}$

de façon à avoir dans le cas d'un minimum une suite bornée par $f(x^*)$ tel que:

$$f(x^{(0)}) > f(x^{(1)}) > f(x^{(2)}) > \dots > f(x^{(k)}) \quad (\text{II.71})$$

Cette suite converge vers l'optimum globale s'il est unique ou vers l'un des optimum locaux s'il en existe plusieurs, d'où la nécessité d'utiliser les méthodes globales.

II.5.1.1 La structure générale d'un algorithme d'une méthode numérique :

Pour calculer le minimum de la fonction f en utilisant une méthode numérique dont son algorithme a la structure suivante :

- Choisir la valeur initiale $x^{(0)}$ et le seuil de tolérance ε ,
- $x^{(k+1)} = x^{(k)} + \Delta x^{(k)}$ (II.72)

Avec $\Delta x^{(k)}$ dépend de la méthode utilisée,

- Test : si $\|\nabla_x f(x^{(k+1)})\| < \varepsilon$, on arrête : $x^{(k+1)}$ est la solution, sinon aller à l'étape (2).

Parmi les méthodes numériques les plus utilisées on trouve :

- La méthode de gradient,
- La méthode de gradient conjugué,
- La méthode de newton,
- La méthode de newton modifié.

II.5.1.2 La méthode du gradient

L'algorithme du gradient désigne un algorithme d'optimisation différentiable. Il est par conséquent destiné à minimiser une fonction réelle différentiable définie sur un espace euclidien R^n .

L'algorithme est itératif et procède donc par amélioration successive.

Au point courant, un déplacement est effectué dans la direction opposée au gradient, de manière à faire décroître la fonction. Le déplacement le long de cette direction est déterminé par technique numérique connue sous le nom de

Recherche linéaire. Cette description montre que l'algorithme fait partie de la famille des algorithmes à directions de descente. Le principe de cette méthode peut être expliqué par l'algorithme suivant:

- Choisir la valeur initiale $x^{(0)}$ et le seuil de tolérance $\varepsilon \geq 0$. L'algorithme du gradient définit une suite d'itération :

$$x^{(1)}, x^{(2)}, \dots, x^{(k)}, \dots$$

Jusqu'à ce que le test d'arrêt soit satisfait, il passe de $x^{(k)}$ à $x^{(k+1)}$ par les étapes suivantes :

1- calcul de $f(x^{(k)})$ et son gradient $\nabla_x f(x^{(k)})$,

2-Vérification de la convergence en utilisant un critère approprié c'est-à-dire :

$$\|\nabla f(x^{(k)})\| \leq \varepsilon \quad (\text{II.73})$$

- Si la convergence est assurée, on arrête et on passe à l'étape (4) , sinon on passe à l'étape 3.

3- Calcul de la nouvelle itération par la relation suivante :

$$x^{(k+1)} = x^{(k)} - \alpha_k \nabla f(x^{(k)}) \quad (\text{II.74})$$

Avec α_k le pas de descente, puis on passe à l'étape 1.

4- La solution du problème d'optimisation (l'optimum) est :

$$x^* = x^{(k)} \quad (\text{II.75})$$

II.5.1.3 Méthode du gradient conjugué

La méthode de gradient conjugué est une variante améliorée de la méthode de la plus grande pente, qui consiste à suivre la direction opposée au gradient, l'inconvénient de cette méthode est de créer des directions orthogonales, ce qui ralentit la convergence de l'algorithme. Le principe de cette méthode peut être explicité par l'algorithme suivant :

- Choisir la valeur initiale $x^{(0)}$ et le seuil de tolérance $\varepsilon \geq 0$.

L'algorithme du gradient définit une suite d'itération :

$$x^{(1)}, x^{(2)}, \dots, x^{(k)}, \dots$$

Jusqu'à ce que le test d'arrêt soit satisfait, il passe de $x^{(k)}$ à $x^{(k+1)}$ en suivant les étapes suivantes :

1- calcul de $f(x^{(k)})$ et son gradient noté: $\nabla_x f(x^{(k)})$.

2-Vérification de la convergence en utilisant critère approprié c'est-à-dire :

$$\|\nabla f(x^{(k)})\| \leq \varepsilon \quad (\text{II.76})$$

- Si la convergence est assurée, on arrête et on passe à l'étape (6), sinon on passe à l'étape 3.

3- calcul de $\beta^{(k)}$ par la relation suivante :

$$\beta^{(k)} = \begin{cases} 0 & \text{si } k = 0 \\ \frac{\|\nabla_x f(x^{(k)})\|^2}{\|\nabla_x f(x^{(k-1)})\|^2} & \text{si } k \geq 1 \end{cases} \quad (\text{II.77})$$

4-Calcul de $p(x^{(k)})$ par la relation suivante :

$$p(x^{(k)}) = -\nabla_x f(x^{(k)}) + \beta^{(k)} p(x^{(k-1)}) \quad (\text{II.78})$$

5- Calcul de la nouvelle itération par l'algorithme :

$$x^{(k+1)} = x^{(k)} + \alpha_k p(x^{(k)}) \quad (\text{II.79})$$

Avec α_k : le pas de descente.

Puis aller a l'étape(1).

6- L'optimum est :

$$x^* = x^{(k)} \quad (\text{II.80})$$

Calcul de pas de descente :

On se déplace toujours dans la direction opposé au gradient $\nabla f(x^{(k)})$ mais on va cette

fois choisir le pas α_k de façon à maximiser la décroissance de f dans cette direction. En fait le α_k choisi va être la valeur de α qui minimise la fonction :

$$g(a) = f(x_k - a\nabla f(x^{(k)}), a \geq 0 \quad (\text{II.81})$$

II.5.1.4 Méthode de Newton :

En optimisation numérique, la méthode de Newton est, dans son application la plus simple, un algorithme efficace pour trouver numériquement une approximation précise un minimum d'une fonction réelle à n variable.

Le principe de cette méthode peut être explicité par l'algorithme suivant :

- choisir la valeur initiale $x^{(0)}$. L'algorithme de Newton définit une suite d'itération :

$$x^{(1)}, x^{(2)}, \dots, x^{(k)}, \dots$$

Jusqu'à ce que le test d'arrêt soit satisfait il passe de $x^{(k)}$, à $x^{(k+1)}$, par les étapes suivant :

1- calcul de $f(x^{(k)})$ et son gradient : $\nabla f(x^{(k)})$

2- calcul de la matrice Hessienne de $f(x^{(k)})$: $\nabla_x^2 f(x^{(k)})$

Vérifiant si :

$$\nabla_x^2 f(x^{(k)}) = 0 \quad (\text{II.82})$$

-si la condition est vérifiée, on arrête et on passe à l'étape 5 sinon on passe à 3.

3- calcul de : $p(x^{(k+1)})$

$$p(x^{(k+1)}) = [\nabla_x^2 f(x^{(k)})]^{-1} \nabla_x f(x^{(k)}) \quad (\text{II.83})$$

4- calcul de la nouvelle itération :

$$x^{(k+1)} = x^{(k)} + p(x^{(k)}) \quad (\text{II.84})$$

Puis aller à l'étape 1.

5- l'optimum est :

$$x^* = x^{(k)} \quad (\text{II.85})$$

II.5. 1.5 Méthode de Newton modifiée

Le principe de cette méthode peut être explicité par l'algorithme suivant :

- choisir la valeur initiale $x^{(0)}$. L'algorithme de Newton définit une suite d'itération :

$$x^{(1)}, x^{(2)}, \dots, x^{(k)}, \dots$$

- jusqu'à ce que le test d'arrêt soit satisfait il passe de $x^{(k)}$, à $x^{(k+1)}$, par les étapes suivant :

1- calcul de $f(x^{(k)})$ et son gradient : $\nabla_x f(x^{(k)})$

2- calcul de la matrice Hessienne de : $f(x^{(k)})$: $\nabla_x^2 f(x^{(k)})$

Vérifiant si :

$$\nabla_x^2 f(x^{(k)}) = 0 \tag{II.86}$$

Si la condition est vérifiée, on arrête et on passe à l'étape 5

3- calcul de $p(x^{(k+1)})$:

$$p(x^{(k+1)}) = -[\nabla_x^2 f(x^{(k)})]^{-1} \nabla_x f(x^{(k)}) \tag{II.87}$$

4- calcul de la nouvelle itération :

$$x^{(k+1)} = x^{(k)} + \alpha_k p(x^{(k)}) \tag{II.88}$$

Puis aller a l'étape 1.

5 - l'optimum est :

$$x^* = x^{(k)} \tag{II.89}$$

Détermination de α_k :

Avant d'entamer l'algorithme de Newton modifié, il faut d'abord calculer le paramètre α_k . Il existe deux méthodes pour le déterminer :

- Méthode 1 : méthode du pas de descente :

On génère des vecteurs $x^{(1)}, x^{(2)}, \dots, x^{(k)}, \dots$ tels que la valeur de la fonction f décroît à chaque itération :

$$f(x^{(k+1)}) < f(x^{(k)}) \tag{II.90}$$

On remplace $x^{(k+1)}$ avec sa valeur et on aura :

$$f(x^{(k)} + \alpha_k p(x^{(k)})) < f(x^{(k)}) \tag{II.91}$$

α_k : Solution de l'équation (II.91) par conséquent plusieurs valeurs peuvent être considérées

pour α_k .

- **Méthode 2 : Méthode du pas de descente optimal**

Dans ce cas α_k est déterminé en résolvant le problème d'optimisation suivant :

$$\min_{\alpha_k} f(x^{(k+1)}) = \min_{\alpha_k} f(x^{(k)} + \alpha_k p(x^{(k)})) \quad (\text{II.92})$$

Remarque

Les méthodes présentées dans ce chapitre sont facile à traiter et à programmer

Elles convergent vers les solutions locales en quelques itération, mais aussi d'une efficacité limitée, lorsque le nombre de variable de décision est élevée la résolution devient compliqué.

De plus elles sont très difficiles à mettre en œuvre généralement elles convergent vers des optimums locaux, puisque la convergence vers un minimum globale n'est pas toujours assuré avec l'utilisation des méthodes locales ce qui nécessite l'utilisation des méthodes globales qui sont considérés généralement très efficaces quand l'évaluation de la fonction objectif est très rapide.

Dans le cas des méthodes locales, la solution trouvée dépend de l'estimé de départ choisi. Et la convergence de ces méthodes impose à la fonction objectif f de vérifier certaines hypothèses d'applicabilité, souvent fortes comme la différentiabilité, qui est très difficile à garantir dans les problèmes concrets. En plus l'estimation du temps de convergence (calcul d'un optimum) n'est souvent pas possible.

II.5. 2 Méthodes globales :

La convergence vers un minimum globale n'est pas toujours assurée avec l'utilisation des méthodes locales se qui nécessite l'utilisation des méthodes globales qui sont considérés généralement très efficaces quand l'évaluation de la fonction objectif est très rapide.

L'optimisation globale a pour but de chercher l'élément $x^* \in D_a$ tel que $f(x^*)$ soit la valeur minimale que prend f sur tout le domaine D_a .

$$\begin{cases} \text{chercher } x^* \in D_a \\ f(x^*) = \min_{x \in D_a} f(x) \end{cases} \quad (\text{II.93})$$

De nombreux problèmes issus de la physique, de la mécanique, de la biomathématique ou du monde industriel conduisent à des problèmes d'optimisation globale. Les méthodes permettant de résoudre cette classe de problèmes se divisent en deux catégories : les méthodes stochastiques et les méthodes déterministes.

II.5.2.1 Les méthodes stochastiques

Les méthodes stochastiques (heuristiques) sont un ensemble d'algorithmes d'optimisation visant à résoudre des problèmes d'optimisation difficiles.

Elles font intervenir le hasard, elles procèdent par voisinage successif. Ces méthodes convergent vers le voisinage d'un optimum global sans l'atteindre avec la précision souhaitée.

Ces méthodes présentent aussi l'avantage de ne pas exiger des hypothèses sur la fonction objectif f par exemple la différentiabilité. Parmi ces méthodes on a les algorithmes génétiques et les algorithmes d'essaim de particule (PSO).

II.5.2.2 Méthodes hybrides

Avant de passer aux méthodes exactes on peut citer une troisième classe qui relie les méthodes stochastiques et les méthodes déterministes nommée les méthodes hybrides, ces méthodes profitent des avantages des méthodes exactes et heuristiques, leur principe consiste à démarrer par une méthode stochastique pour converger vers un point au voisinage de l'optimum global, et pour améliorer davantage la précision on continue avec une des méthodes déterministes dont l'estimé de départ de la solution est le point trouvé par la méthode stochastique, par exemple si la fonction objectif f est deux fois différentiable, on peut utiliser les algorithmes génétiques, et après la convergence on continue la recherche avec la méthode de Newton en considérant comme estimé de départ le point trouvé par les algorithmes génétiques. L'inconvénient de cette classe est le temps de calcul qui peut être important [6,7].

Remarque

L'inconvénient majeur de ces méthodes citées précédemment est qu'on ne peut garantir leur convergence vers la solution globale que d'une manière asymptotique, bien que ces méthodes puissent être adaptées à tout type de problème d'optimisation, mais elles sont souvent moins puissantes que les méthodes exactes sur certains types de problèmes.

II.5.2.3 Les méthodes exactes

Ces méthodes requièrent des propriétés mathématiques restrictives de la fonction objectif à optimiser telle que la continuité, la différentiabilité, et la convexité.

Ces méthodes permettent de localiser la solution avec une très grande précision au prix, généralement, d'un nombre d'itérations important. Parmi ces méthodes on peut mentionner la méthode d'Aliénor.

La méthode d'Aliénor est une méthode récente qui consiste à approcher l'espace R^n par R à l'aide d'une transformation réductrice. C'est-à-dire ramener un problème d'optimisation de plusieurs variables à un problème d'optimisation d'une seule variable.

Les inventeurs des méthodes proposent une première transformation réductrice qui se base sur la spirale d'Archimède :

Soit le point $(x, y) \in R^2$ qui s'écrit en coordonnées polaires :

$$\begin{cases} x = r \cos \theta \\ y = r \sin \theta \end{cases} \quad (\text{II.99})$$

On peut alors relier r et θ grâce à la spirale d'Archimède d'équation :

$$r = \alpha \theta \quad (\text{II.100})$$

Où α est un paramètre fixe, destiné à tendre vers 0, les relations deviennent :

$$\begin{aligned} x &= \alpha \theta \cos \theta = h_1(\theta) \\ y &= \alpha \theta \sin \theta = h_2(\theta) \end{aligned} \quad (\text{II.101})$$

Par conséquent, elle permet d'exprimer x et y en fonction d'une seule variable.

Pour trois variables x, y, z on relie d'abord les variables x et y à l'aide d'une spirale d'angle θ_1 ce qui donne :

$$\begin{aligned} x &= \alpha \theta_1 \cos \theta_1 = h_1(\theta) \\ y &= \alpha \theta_1 \sin \theta_1 = h_2(\theta) \end{aligned} \quad (\text{II.102})$$

Puis on relie θ_1 et z à l'aide d'une spirale d'angle θ on posant :

$$\begin{aligned} \theta_1 &= \alpha\theta\cos\theta \\ z &= \alpha\theta_1\sin\theta = h_3(\theta) \end{aligned} \tag{II.103}$$

Ainsi, on obtient la courbe paramétrée $h(\theta) = (h_1(\theta), h_2(\theta), h_3(\theta))$ définie par :

$$\begin{cases} h_1(\theta) = \alpha^2\theta\cos\theta\cos(\alpha\theta\cos\theta) \\ h_2(\theta) = \alpha^2\theta\cos\theta\sin(\alpha\theta\cos\theta) \\ h_3(\theta) = \alpha\theta\sin\theta \end{cases} \tag{II.104}$$

On peut généraliser ce procédé à n variable $x_1, x_2, x_3, \dots, x_n$ en les reliant deux par deux par les spirales d'angle θ_i . À la fin du processus on obtient la variable θ qui permet d'exprimer tous les x_i :

$$x_i(\theta), i = 1, \dots, n \quad ; \theta \geq 0 \tag{II.105}$$

Nous avons approché l'espace R^n par R grâce à une transformation réductrice utilisant la spirale d'Archimède. La précision de cette approximation dépend du coefficient α . Plus α est petit, plus la précision est meilleure.

La méthode d'Aliénor peut être exploitée pour la résolution des problèmes d'optimisations suivant :

$$\min_{x_1, \dots, x_n} f(x_1, \dots, x_n) \tag{II.106}$$

Où f est une fonction non linéaire continue.

La transformation permet de remplacer la fonction $f(x_1, \dots, x_n)$ par la nouvelle fonction $f^*(\alpha)$ donnée comme suit :

$$f^*(\alpha) = f(h_1(\theta), \dots, h_n(\theta)) \tag{II.107}$$

Qui est une fonction d'une seule variable.

Le problème d'optimisation est alors remplacé par le problème suivant :

$$\min_{\theta \geq 0} f^*(\theta) \quad (\text{II.108})$$

Pour simplifier les calculs, d'autres transformations réductrices ont été proposées pour exprimer toutes les variables en fonction de la variable α en une seule étape au lieu de réduire deux à deux les variables par la méthode d'Aliénor de base. Ceci impose de choisir correctement les fonctions h_i et la transformation réductrice. Par conséquent, les efforts se sont focalisés sur un point dont l'objectif est de développer des transformations réductrices simples qui demandent moins de calculs et surtout de bien approximer le problème d'optimisation à n variables.

Parmi les transformations réductrices proposées dans la littérature on retrouve les transformations suivantes :

Transformation 1 :

$$x_i = \beta \alpha \cos(\rho_i \alpha) \quad (\text{II.109})$$

Où les paramètres ρ_i forment une suite croissante et vérifient la condition suivante :

$$\frac{\rho_i}{\rho_{i+1}} < 1 \text{ et } \beta > 0 \quad (\text{II.110})$$

Transformation 2

$$x_i = \beta \cos(m^i \pi \alpha) \quad (\text{II.111})$$

Avec $m > 1$

Transformation 3

$$x_i = \beta \alpha \cos(\rho_i \alpha) \quad (\text{II.112})$$

Les paramètres $\rho_i, \dots, \rho_{i+1}$ sont choisis proches les uns des autres, tout en constituant une suite lentement croissante, par exemple :

$$\rho_{i+1} = \rho_i + \varepsilon \quad (\text{II.113})$$

Avec $\varepsilon > 0$ est choisi très petit. Le paramètre ρ_n est choisi comme suit :

$$\rho_n = \pi \sqrt{n-1} \frac{\rho_{n-1}}{\beta} \quad (\text{II.114})$$

Transformation 4 :

$$x_i = \beta \alpha \cos(\omega_i \alpha + \varphi_i) \quad (\text{II.115})$$

Où les $\omega_i > 0$ forment une suite croissante. φ_i est une suite lentement croissante dont les termes sont proches l'un des autres, par exemple

$$\varphi_{i+1} = \varphi_i + \varepsilon \quad (\text{II.116})$$

Avec $\varepsilon > 0$ est choisi très petit.

La dernière transformation réductrice permet une exploration globale du plan paramétré (la spirale d'Archimède) par α en assurant une très bonne précision par rapport aux autres transformations et une convergence rapide vers l'optimum global. Aussi d'autres variantes ont été proposées pour lier r et α afin d'améliorer la précision de l'approximation. Parmi ces relations, on peut citer le suivant :

$$\begin{aligned} r &= e^{-\beta \alpha} e \\ r &= e^{-\beta \alpha^2 + b \alpha}, b > 0 \end{aligned} \quad (\text{II.117})$$

Pour illustrer la transformation de la fonction à plusieurs variables à une fonction $f^*(\alpha)$ à une seule variable, considérons le problème d'optimisation à deux variables de décision suivant :

$$\min_x \cos(x_1) \sin(x_1^2 - x_1) + \cos(x_2) \sin(x_2^2 - x_2)$$

Sujet à :

$$\begin{aligned} -10 &\leq x_1 \\ x_2 &\leq 10 \end{aligned} \quad (\text{II.118})$$

On propose d'utiliser la transformation réductrice (4), et pour respecter les contraintes il suffit de prendre $\beta = 10, \omega_1 = 100, \omega_2 = 101, \varphi_1 = 1$ et $\varepsilon = 0.00001$

Ce qui donne :

$$\begin{aligned} x_1 &= 10 \cos(100\alpha + 1) \\ x_2 &= 10 \cos(101\alpha + 1.00001) \end{aligned} \quad (\text{II.119})$$

On aura :

$$\begin{aligned} f^*(\alpha) &= \cos 10 \cos(100\alpha + 1) \sin((10 \cos(100\alpha + 1))^2 - 10 \cos(100\alpha + 1)) + \\ &\cos(10 \cos(101\alpha + 1.00001)) \sin((10 \cos(101\alpha + 1.00001))^2 - 10 \cos(100\alpha + \\ &1.00001)) \end{aligned} \quad (\text{II.120})$$

En résolvant l'équation algébrique suivante :

$$\nabla_{\alpha} f^*(\alpha) = 0 \quad (\text{II.121})$$

On détermine facilement le minimum global α^* . Ensuite on détermine la solution globale comme suit :

$$\begin{aligned} x_1^* &= 10 \cos(100\alpha^* + 1) \\ x_2^* &= 10 \cos(101\alpha^* + 100001) \end{aligned} \quad (\text{II.122})$$

Conclusion

Dans ce chapitre, on a donné un aperçu général sur l'optimisation. Après la formulation mathématique d'un problème d'optimisation, et la présentation des différentes classe des problèmes d'optimisation, on s'est intéressé à la présentation des différentes méthodes numériques de résolutions des problèmes d'optimisation sans contraintes et on a cité les inconvénients de ces méthodes et la nécessité de l'utilisation des méthodes globale. Dans ce chapitre notre étude est basée sur la méthode d'Aliénor qui est classée parmi les méthodes d'optimisation exacte, dont son avantage est de localiser l'optimum global. Cette méthode sera exploitée dans le prochain chapitre pour l'identification des paramètres des systèmes dynamique, après avoir reformulé le problème sous la forme d'un problème d'optimisation.

III.1 Introduction

La méthode d'Aliénor consiste à approcher l'espace R^n par R à l'aide d'une transformation réductrice. C'est-à-dire ramener un problème d'optimisation de plusieurs variables de décision à un problème d'optimisation d'une seule variable [1].

Cette méthode permet de simplifier davantage la recherche de la solution et de localiser l'optimum global. Pour montrer l'efficacité de la méthode d'Aliénor appliquée pour l'identification des systèmes, deux exemples d'application ont été étudiés.

III.2 l'identification des systèmes dynamiques par l'optimisation globale

La méthode du modèle est basée sur la comparaison des comportements du modèle et du système à identifier suite à une même excitation. C'est l'une des méthodes puissantes appliquées aux systèmes non linéaires.

III.2.1 Le principe La méthode du modèle

Le principe de la méthode du modèle est donné par la Figure (I.15).

L'idée consiste à minimiser l'écart observé entre les mesures expérimentales $y(k)$ et celles prédites par le modèle $y(\hat{k})$ en utilisant un algorithme d'optimisation qui adapte les paramètres du modèle d'une manière à avoir l'erreur $e(k) = y(k) - y(\hat{k})$ proche de zéro pour toute valeur de k qui désigne l'instant d'échantillonnage [2,3].

Dans ce cas le modèle et le système physique sont soumis à une même entrée $u(k)$ supposée suffisamment excitante pour toutes les dynamiques du système. Par conséquent, pour l'identification des paramètres, on considère la minimisation d'une fonction objectif (critère) qui dépend explicitement de l'erreur $e(k)$. Généralement, cette fonction objectif prend la forme suivante :

$$J = \sum_{k=0}^{+\infty} f(e(k)) = \sum_{k=0}^{+\infty} f(y(k) - \hat{y}(k)) \quad (\text{III.123})$$

où $f(e(k))$ est une fonction non linéaire qui mesure l'écart entre les deux sorties mesurée et prédite $y(k)$ et $\hat{y}(k)$. La sortie prédite $\hat{y}(k)$ du modèle, dans le cas discret, est donnée comme suit :

$$\hat{y}(k) = g(x(k), \theta) \quad (\text{III.124})$$

Avec :

$$x(k) = [u(k), \dots, u(k - n_u), y(k), \dots, y(k - n_y)] \quad (\text{III.125})$$

Où :

$u(k)$ et $y(k)$: représentent respectivement la commande et sortie du système.

$x(k)$: représente le vecteur de régression.

n_u et n_y : représentent respectivement les retards tolérés pour l'entrée et la sortie du système

θ : représente le vecteur des paramètres du modèle à identifier. Ce vecteur est de dimension n .

Ainsi, en remplaçant $\hat{y}(k)$ par son expression (III.124) dans la fonction objectif (III.123) on aura :

$$J = \sum_{k=0}^{+\infty} f(y(k) - g(x(k), \theta)) \quad (\text{III.126})$$

On aura le critère suivant : Comme les mesures $y(k)$ et les commandes $u(k)$ sont disponibles, il est clair que le critère J dépend explicitement de vecteur des paramètres θ à identifier.

$$J(\theta) = \sum_{k=0}^{+\infty} f(y(k) - g(x(k), \theta)) \quad (\text{III.127})$$

Le problème d'identification des paramètres regroupés dans le vecteur θ se ramène à la résolution du problème d'optimisation globale suivant :

$$\min_{\theta} J(\theta) = \min_{\theta} \sum_{k=0}^{+\infty} f(y(k) - g(x(k), \theta)) \quad (\text{III.128})$$

La complexité du problème d'optimisation dépend de la fonction objectif f considérée. Généralement, cette fonction prend la forme quadratique telle que :

$$f(e(k)) = e^2(k) = [y(k) - \hat{y}(k)]^2 \quad (\text{III.129})$$

D'où le critère à minimiser est :

$$J(\theta) = \sum_{k=0}^{+\infty} [y(k) - g(x(k), \theta)]^2 \quad (\text{III.130})$$

Ce problème d'optimisation peut être résolu par plusieurs méthodes qui sont généralement itératives. Cependant, ces méthodes numériques peuvent tomber dans un minimum local et deviennent très délicates lorsque le nombre de variables est très élevé.

Afin d'y remédier à tous ces problèmes, nous proposons de chercher le minimum global du critère (III.130) par la méthode d'Aliénor.

Exemple d'application 1 :

Le modèle du système linéaire à identifier est donné comme suit :

$$y(k) = -a * y(k - 1) - b * y(k - 2) + c * u(k) \quad (\text{III.131})$$

On a fixé les paramètres du modèle a, b, c pour obtenir les valeurs expérimentales tel que :

$$a = -0.4232, \quad b = 0.5390, \quad c = 0.3456,$$

θ est le vecteur de paramètres à identifier

On aura :

$$\theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \quad (\text{III.132})$$

Pour faire la simulation on doit injecter une entrée considérée comme un échelon unitaire $u = 1$ et trente huit valeurs pour faire l'identification, tels que $y(1) = y(2) = 1$,
A l'aide de la transformation du konfé, on aura les fonctions suivantes :

$$\begin{aligned} \theta_1 = a &= 2 \cos(5000\alpha + 1) \\ \theta_2 = b &= 2 \cos(5001\alpha + 1.00005) \\ \theta_3 = c &= 2 \cos(5002\alpha + 1.0001) \end{aligned} \quad (\text{III.133})$$

Avec :

$$\beta = 2; \quad \omega_1 = 5000; \quad \omega_2 = 5001; \quad \omega_3 = 5002; \quad \varphi_1 = 1; \quad \varphi_2 = 1.00005; \quad \varphi_3 = 1.0001$$

Le problème d'optimisation à résoudre est le suivant :

$$\min_{\alpha} \sum_{k=4}^{13} (y(k) + 2 \cos(5000\alpha + 1) y(k-1) + 2 \cos(5001\alpha + 1.00005) y(k-2) - 2 \cos(5002\alpha + 1.0001) u(k))^2 \quad (\text{III.134})$$

La résolution de ce dernier conduit à l'optimum global $\alpha^* = 3.6411$ En remplaçant $\alpha^* = 3.6411$ dans les relations, (III. 133) on obtient les paramètres du modèle suivants :

$$\theta_1^* = a^* = -1.4069; \quad \theta_2^* = b^* = 0.5541; \quad \theta_3^* = c^* = 0.4342$$

La valeur de critère est $j(\alpha^*) = 0.0055$

La représentation de la sortie des deux modèles prédite et mesuré

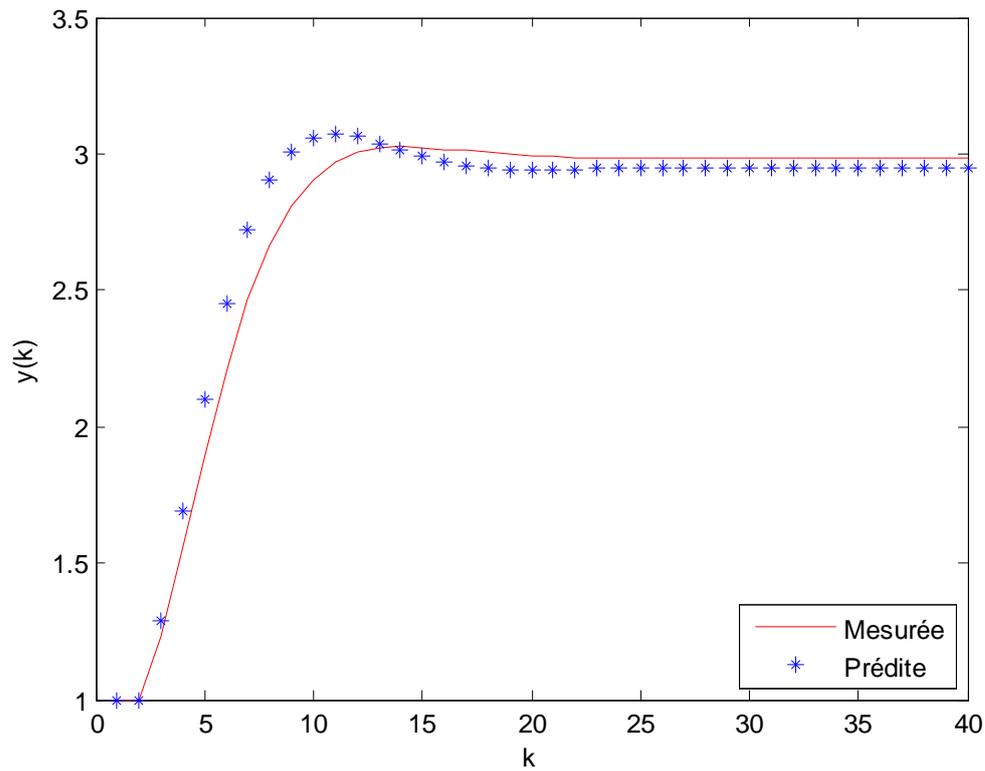


Figure (III.16) ; représentation des sorties (mesurée et prédite)

Le graphe de la fonction $y(k)$ est donné par la figure (III.16) , on remarque que la fonction $Y(k)$ est oscillatoire et les sorties sont identiques

On a obtenue la solution après l'application du la méthodes des évaluations simultanées , on à considérés le pas de discrétisation $\Delta_w = 10^{-5}$ et $w_{max} = 2$.

III.2.2 Exemple d'application 2

Le modèle du système non linéaire à identifier est donné sous forme d'une équation de récurrence (modèle entrée-sortie discret) suivant :

$$y(k + 2) = ay(k + 1) + by(k)u(k + 1) + cu(k + 1) \quad (\text{III.135})$$

Pour générer les mesures, on fixe d'abord des valeurs pour les paramètres du modèle, et on choisit un horizon de temps déterminé pour faire la simulation, et on récupère les valeurs de la sortie correspondante à une entrée bien choisie.

Les valeurs obtenues pour la sortie seront considérées comme les mesures expérimentales et seront utilisées pour l'identification.

Pour obtenir les mesures expérimentales, les valeurs suivantes
 $a = 0.9, b = -0.7, c = 0.5$

Ont été fixées pour les paramètres du modèle. Ainsi, le vecteur de paramètres à Identifier est :

$$\theta = (\theta_1, \theta_2, \theta_3)^T = (a, b, c)^T \quad (\text{III.136})$$

Pour l'identification de système, treize mesures ont été considérées en prenant $y(0) = y(1) = 0$ et une entrée $u(k)$ aléatoire.

Pour l'identification, en utilisant la méthode d'Aliénor pour la résolution du problème d'optimisation (III.130), on utilise la transformation réductrice (III.115) avec les paramètres suivants $w_0 = 100, \varphi_0 = 1$ et $\varepsilon = 5 * 10^{-5}$

Ainsi, en explicitant les variables de décision θ_i regroupé dans le vecteur θ , en fonction de la seule variable de décision α , à l'aide de la transformation réductrice (III.115), le problème d'identification (III.130) se réduit au problème d'optimisation à une seule variable de décision α suivant :

$$\min_{\alpha} \hat{j}(\theta) = \min_{\alpha} \sum_{k=0}^{+\infty} (y(k) - g(x(k), h(\alpha)))^2 \quad (\text{III.137})$$

Où $h(\alpha)$ est une fonction vectorielle donnée comme suit :

$$h(\alpha) = \begin{bmatrix} \theta_1 = h_1(\alpha) \\ \theta_2 = h_2(\alpha) \\ \theta_3 = h_3(\alpha) \end{bmatrix} = \begin{bmatrix} \cos(100\alpha + 1.0000) \\ \cos(101\alpha + 1.00005) \\ \cos(102\alpha + 1.0001) \end{bmatrix} \quad (\text{III.138})$$

Le problème d'optimisation à résoudre est le suivant :

$$\min_{\alpha} \sum_{k=0}^{11} (y(k+2) - ay(k+1) - by(k)u(k+1) - cu(k+1))^2 \quad (\text{III.139})$$

La solution de problème a été réalisée en utilisant la méthode des évaluations simultanées.

Dans cette méthode, on fixe, à priori, le nombre de points équidistants auxquels on évaluera la valeur du critère (III.139) (points supports de l'évaluation). L'application de la méthode conduit à l'optimum global $\alpha^* = 3.0344$.

En remplaçant α^* dans les relations (III.138), on obtient les Paramètres du modèle suivants

$$\theta_1^* = a^* = 0.9027 ; \theta_2^* = b^* = -0.7366 ; \theta_3^* = c^* = 0.5024$$

La valeur de critère est $j(\alpha^*) = 6.3782 * 10^{-6}$

Cet exemple montre que la méthode d'Aliénor permet d'identifier de manière précise les paramètres du système, puisque les valeurs estimées sont très proches des valeurs réelles. **La**

représentation du la sortie des deux modèles prédite et mesur

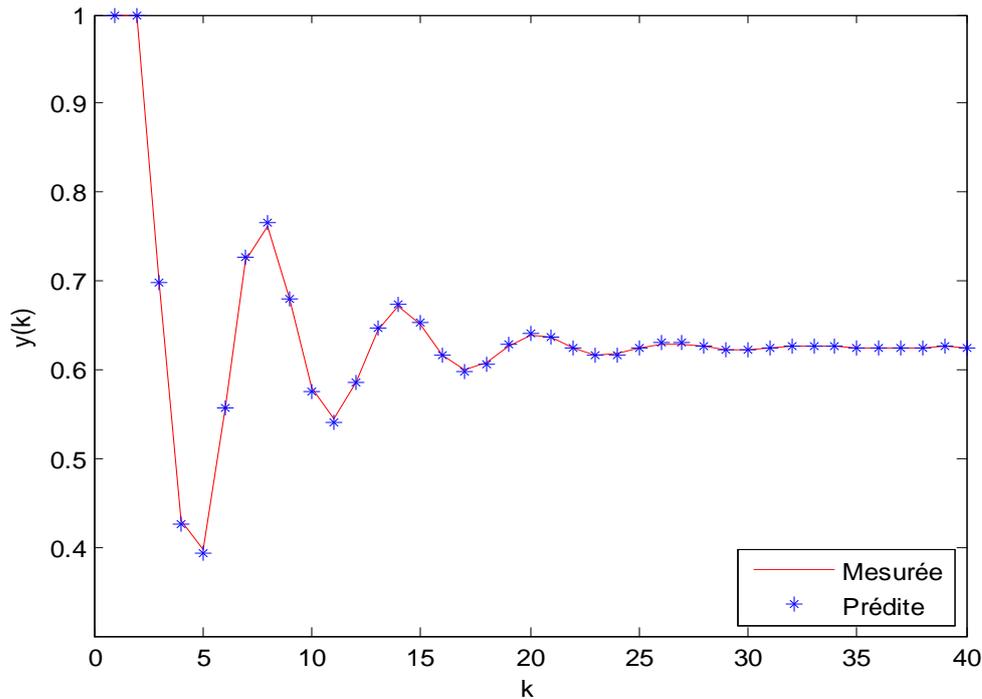


Figure (III.17) ; représentation des sorties (mesurée et prédite)

Le graphe de la fonction $y(k)$ est donné par la figure (III.17) , on remarque que la fonction $Y(k)$ est oscillatoire et les sorties sont identique

On a obtenue la solution après l'application du la méthodes des évaluations simultanées, on à considérés le pas de discrétisation $\Delta_w = 10^{-5}$ et $w_{max} = 2$.

Conclusion

Dans ce chapitre nous avons identifié les paramètres d'un système non linéaire par la méthode du modèle qui a pour objectif de minimiser l'écart entre les mesures expérimentales et celles prédites par le modèle à identifier.

La résolution de ce problème revient à utiliser la méthode d'Aliénor qui est une méthode exacte dont le principe consiste à utiliser une transformation réductrice pour ramener le problème d'optimisation initiale à Plusieurs variables à un problème d'optimisation à une seule variable dont la résolution est simple, et l'obtention de l'optimum global avec une très grande précision est assurée .

Le travail représenté dans ce mémoire porte sur l'identification paramétrique d'un système par la méthode du modèle basé sur la minimisation d'un critère en fonction de l'écart entre la sortie réelle du procédé et celle prédite par le modèle. La résolution du problème d'optimisation est réalisée par la méthode d'optimisation globale d'Aliénor.

Ainsi, après avoir donné quelques généralités sur l'identification, on a présentée les différentes méthodes d'identification. par la suite on s'est intéressé aux méthodes d'optimisation en particulier la méthodes d'optimisation globale Aliénor.

L'identification paramétrique par la méthode du modèle permet l'obtention des résultats efficaces qui converge vers les paramètres réels du système. La méthodes d'optimisation globale Aliénor réduit le temps de calcul en ramenant le problème d'optimisation de plusieurs variables a un problème d'optimisation d'une seule variable facile à résoudre et converge vers l'optimum global. Ce constat est démontré lors de l'étape de validation du modèle. L'efficacité est démontrée même en présence du bruit de mesure.