

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mouloud MAMMERI, Tizi-Ouzou



Faculté de Génie Electrique et d'Informatique
Département d'Automatique

Mémoire de Fin d'Etudes

En vue de l'obtention du diplôme

Master académique en Automatique
Option : commande des systèmes

Thème

*Acquisition des données par LabView en
vue de l'Identification d'une station de
pression*

Proposé et Dirigé par :

Mr. MAIDIA

Présenté par :

Mr. LOUHI Yacine
Mr. EDJEKOUANE Essaid

Soutenu le : / /2010

Promotion 2010

Ce travail a été préparé au : Laboratoire d'automatique industrielle et de régulation au niveau du département Automatique, de l'Université Mouloud MAMMERI de Tizi-Ouzou.

Remerciement



Nous tenons à remercier nos parents, notre encadreur Mr MAIDI Ahmed pour ses conseils, son orientation et son aide le long de notre projet de fin d'étude.

Nous remerciment aussi Mr CHARIF et Mr HESNAOUI pour leur aide précieuse.

Nos remerciements à tous nos enseignants et à toute personne ayant contribué de près ou de loin à la réalisation de ce projet.



Sommaire

Sommaire

Introduction générale	1
------------------------------------	---

CHAPITRE 1 : GENERALITES SUR LES MODELES

Introduction	3
1.1 Notion générale sur les systèmes.....	3
1.2 Modèles	5
1.2.1 Définition du modèle	6
1.3 Utilisation du modèle	6
1.4 Différents types de modèles	7
1.4.1 Modèle de connaissance	7
1.4.2 Modèle de représentation et modèle de conduite.....	8
1.4.2.1 Modèle de règles.....	9
1.4.2.2 Modèle fichiers	9
1.4.2.3 Modèle entrées-sorties	9
1.4.2.4 Modèle d'état	10
1.4.2.5 Modèle graphiques.....	11
1.5 Obtention du modèle	14
1.5.1 Modélisation mathématique.....	14
1.5.1.1 Modèle à temps continu.....	15
1.5.1.2 Modèles à temps discontinu ou discret.....	15
1.5.1.3 Modèle linéaire	16
1.5.1.4 Modèle non linéaire	17
1.5.2 Modélisation expérimentale.....	18
Conclusion.....	19

CHAPITRE 2 : IDENTIFICATION DES SYSTEMES

Introduction	20
2.1 Généralités.....	20
2.1.1 Définition	20
2.1.2 Objectif	20
2.2 Méthodes d'identification	21
2.2.1 Méthodes graphique.....	21
2.2.1.1 Objectif	21
2.2.1.2 Système apériodique sans retard.....	22
2.2.1.3 Systèmes apériodiques avec retard	24
2.2.1.4 Méthode des moments	26
2.2.1.5 Système oscillant	26
2.2.1.6 Systèmes intégrateurs	27
2.2.2 Méthodes numériques	28
2.2.2.1 Méthode des moindres carrés	28

Sommaire

2.3 Etapes et mise en œuvre de l'identification.....	36
2.3.1 Mise au point du protocole d'identification.....	37
2.3.2 Réalisation de l'expérience	41
2.3.3 Prétraitement des données	41
2.3.4 Calcul des paramètres du modèle	43
2.3.5 Validation des résultats	45
Conclusion.....	46

CHAPITRE 3 : ACQUISITION DES DONNEES AVEC LABVIEW

Introduction	47
3.1 C'est quoi l'acquisition ?.....	47
3.2 Traitement des mesures	48
3.3 Acquisition de données avec LabView	49
3.3.1 Présentation du logiciel LabView	49
3.3.1.1 Les éléments principaux d'un VI.....	50
3.3.1.2 Structure de données dans LabView.....	53
3.3.1.3 Traitement numérique.....	54
3.3.1.4 Quel type de programmation utilise-t-on pour LabView ?.....	55
3.3.1.5 Structure de programme	55
3.3.2 Démarche d'acquisition	55
Conclusion.....	56

CHAPITRE 4 : IDENTIFICATION D'UNE STATION DE PRESSION

Introduction	57
4.1 Présentation (description) du procédé	57
4.1.1 Bloc d'alimentation.....	58
4.1.2 Station de pression	58
4.1.2.1 Eléments de la station	58
4.1.2.2 Fonctionnement de la station	59
4.1.3 PC et carte d'acquisition	60
4.2 Acquisition de données par LabView	61
4.3 L'identification du procédé	63
4.3.1 Méthode d'identification utilisée	64
4.4 Validation du modèle	66
Conclusion.....	67
Conclusion générale	68
Bibliographie.....	69

Introduction générale

Introduction générale

L'obtention d'un modèle mathématique d'un système dynamique joue un rôle très important dans le domaine d'automatique. Un modèle mathématique est utilisé pour l'analyse, la synthèse d'un système de commande, la prédiction et le diagnostic.

Ainsi, on ne peut pas mettre en œuvre un système de commande ou de régulation sans passer par la modélisation. En effet, le modèle obtenu ne doit pas être trop simple au risque de ne pas représenter la réalité, mais doit être suffisamment simple pour ne pas rendre inutilement complexes les étapes d'analyse des propriétés du système et de synthèse des régulateurs. La phase de modélisation est donc essentielle dans le processus d'analyse et de synthèse d'un système de commande en automatique. Cependant, l'obtention du modèle peut se faire de deux manières différentes :

✓ Par modélisation : les modèles de connaissance sont construits à partir d'une analyse physique, chimique, biologique ou autre, en appliquant soit les lois générales, fondées sur des principes (lois de la mécanique, de l'électromagnétisme, de la thermodynamique, de la physique quantique, etc.), soit les lois empiriques (finance, économie), qui régissent les phénomènes intervenant au sein des processus étudiés. Cette méthode est limitée, car il arrive fréquemment que le processus soit trop complexe, ou que les phénomènes qui le régissent soient trop mal connus, pour qu'il soit possible d'établir un modèle de connaissance suffisamment précis pour l'application considérée.

✓ Par identification : les modèles de représentation sont construits essentiellement sur la base de mesures effectuées sur les entrées et les sorties du processus à modéliser. Plusieurs méthodes existent classées en deux catégories, les méthodes graphiques et les méthodes numériques, qui sont toujours obtenues en suivant une méthodologie bien déterminée à savoir : une acquisition des données Entrées/Sorties (prétraitement de données), estimation de la complexité du modèle (ou choix), calcul des paramètres et validation du modèle.

L'objectif de ce travail est de mettre en œuvre une identification d'une station de pression existant dans le laboratoire d'automatique industrielle et de régulation au niveau du département Automatique, de l'Université Mouloud MAMMERI de Tizi-Ouzou, en utilisant le langage de programmation graphique LabView pour l'acquisition de données en temps réel.

Introduction générale

Le mémoire est organisé comme suit :

- Le premier chapitre est consacré à des notions générales sur les systèmes et le modèle mathématique tout en détaillant les différents types de ce dernier, et les différentes étapes à suivre pour son obtention.
- Le deuxième chapitre sera réservé à l'identification des procédés et sa mise en œuvre.
- Le troisième chapitre abordera l'acquisition des mesures par le logiciel LabView.
- L'objectif du quatrième est d'identifier graphiquement les paramètres du modèle de la station de pression en utilisant les mesures acquises par LabView.

Les chapitres se terminent par une conclusion et des perspectives de continuité.

CHAPITRE 1:

Généralités sur les modèles

Introduction

Pour concevoir un système de commande, il est utile de modéliser le comportement du procédé à commander. Celui-ci pourra être utilisé pour concevoir l'algorithme de commande et/ou calculer son réglage, ou encore pour la mise au point d'estimateur d'état. Ce chapitre a pour but de mieux comprendre la notion de système, de modèle ainsi que la méthodologie pour la mise au point de différents modèles.

1.1 Notion générale sur les systèmes [1]

Un **système**, agrégation d'éléments interconnectés, est constitué naturellement ou artificiellement afin d'accomplir une tâche prédéfinie. Un **système** est, de manière générale, un ensemble de composantes agencées de façon à produire un certain résultat. Son état est affecté par plusieurs variables, qui sont les **entrées** du système. Le résultat de l'action des entrées sur le système constitue sa **réponse**, caractérisée par le comportement d'une ou de plusieurs variables de **sortie**. Si entrées et sorties sont des fonctions du temps définies à chaque instant t , on parle de système à fonctionnement continu ou, plus brièvement, de **système continu** ; si elles sont seulement définies à certains instants discrets (numéro k d'étape de fonctionnement), on parle de **système discret** ou **récurrent**. Si l'on considère seulement une entrée $e(t)$ ou $e(k)$ et une sortie $s(t)$ ou $s(k)$, on a affaire à un système **monovarié**. Au contraire, un système **multivarié** possède plusieurs entrées :

$e_1(t), \dots, e_m(t)$ ou $e_1(k), \dots, e_m(k)$ que l'on groupe souvent en un **vecteur d'entrée**: $\mathbf{e} = [e_1 \dots e_m]^T$ (T désignant la transposée) et il possède plusieurs sorties : $s_1(t), \dots, s_n(t)$ ou $s_1(k) \dots s_n(k)$ souvent groupées en un **vecteur de sortie** : $\mathbf{s} = [s_1 \dots s_n]^T$.

On représente schématiquement un système par un **diagramme fonctionnel** consistant en un rectangle dans lequel entrent une ou plusieurs flèches représentant la ou les entrées, et duquel sortent une ou plusieurs flèches représentant la ou les sorties (**Figures 1.1 et 1.2**). La direction des flèches exprime que les sorties sont causées par les entrées. La notion de système est extrêmement générale. La relation par laquelle les entrées déterminent les sorties est l'expression des lois physiques (de la mécanique et de l'électricité) qui régissent le fonctionnement.

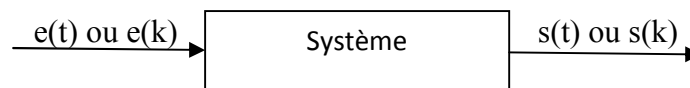


Figure 1.1 Représentation schématique d'un système monovariante

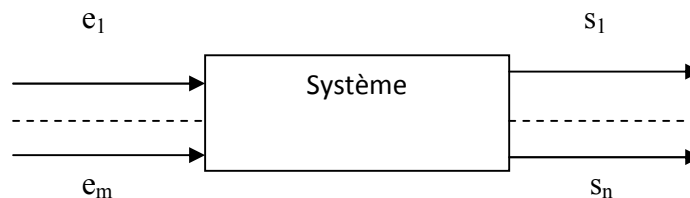


Figure 1.2 Représentation schématique d'un système multivariable.

Les entrées affectant un système peuvent être de nature différentes. Les unes ont pour but d'exercer des actions entraînant le fonctionnement souhaité du système ; ce sont **les commandes**. Les autres entrées troublent le fonctionnement désiré et sont définies comme **des perturbations**.

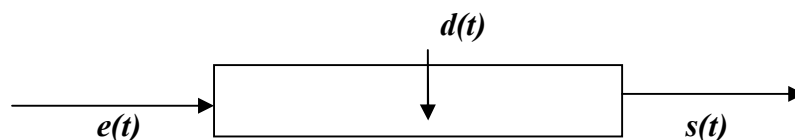


Figure 1.3 Commandes $e(t)$ et perturbations $d(t)$.

Chaque élément constitutif de l'ensemble système peut être caractérisé par un nombre fini de variables et l'interdépendance des variables caractérisant chaque élément peut être exprimée sous la forme d'une loi mathématique. Ainsi, la relation entre les entrées et les sorties du système est l'expression des lois de la physique associées au système. L'ensemble des lois mathématiques régissant la causalité entre les entrées et les sorties du système constitue **le modèle mathématique** du système.

1.2 Modèles [3]

Tout d'abord, il est important de préciser que les modèles à étudier sont en général de type *dynamique*, permettant de représenter l'évolution d'un système dans le temps, par opposition aux modèles de type *statique*, qui décrivent un état stationnaire du système. Ces derniers (modèles statiques) pourront être utilisés pour :

- La recherche du point de fonctionnement (installation continue)
- Le dimensionnement d'une installation.

Alors que les premiers (modèles dynamiques) seront nécessaires pour analyser :

- La compensation des perturbations (régulation)
- Les changements de point de consigne (asservissement)
- L'évolution du système à partir d'un point de départ donné.
- L'optimisation de paramètres et de trajectoire (systèmes batch).

Remarque 1.1 *Dans le cadre des modèles dynamiques, il est important de distinguer différents types qui ont pour objectifs de décrire le système avec plus ou moins de détails.*

1.2.1 Définition du modèle

Le modèle d'un système est une structure mathématique qui, soumise aux mêmes signaux d'action que le système réel, reproduit le comportement d'une expérience. C'est-à-dire que les variables calculées à partir du modèle sont voisines des valeurs mesurées sur le processus. Il faut toujours rester prudent sur l'aptitude d'un modèle à reproduire la réalité. En effet, si le modèle a été validé à partir d'une expérience, il se peut que, pour d'autres conditions d'expérience, le modèle ne fournisse plus des résultats voisins des mesures. Il est d'usage de parler d'un modèle établi pour un domaine de fonctionnement donné et dans un cadre d'hypothèses. Un modèle, quelle que soit la valeur du signal appliqué à l'entrée, donnera un résultat ; le système réel, lui, peut exploser ! La réalité physique d'un phénomène est généralement très complexe et bien souvent on est amené à faire des hypothèses simplificatrices pour traduire cette réalité en équations. L'aptitude du modèle à décrire une expérience sera sa qualité essentielle, mais dès à présent, retenons que le modèle exact

n'existe pas. Le modèle doit satisfaire deux exigences opposées : reproduire la réalité le plus précisément possible et avoir une structure mathématique la plus simple possible (pour une utilisation aisée).

1.3 Utilisation du modèle

Le développement d'un modèle pour un système physique peut être réalisé pour différentes raisons. Citons par exemple le dimensionnement d'une installation, la formation des opérateurs ou encore dans l'objectif d'une meilleure compréhension des phénomènes. Dans le cas de la commande, on peut distinguer les aspects suivants :

- La mise au point de la stratégie de commande,
- La conception de la loi de commande proprement dite et son réglage,
- La conception de capteurs logiciels ou estimateurs d'états du système.

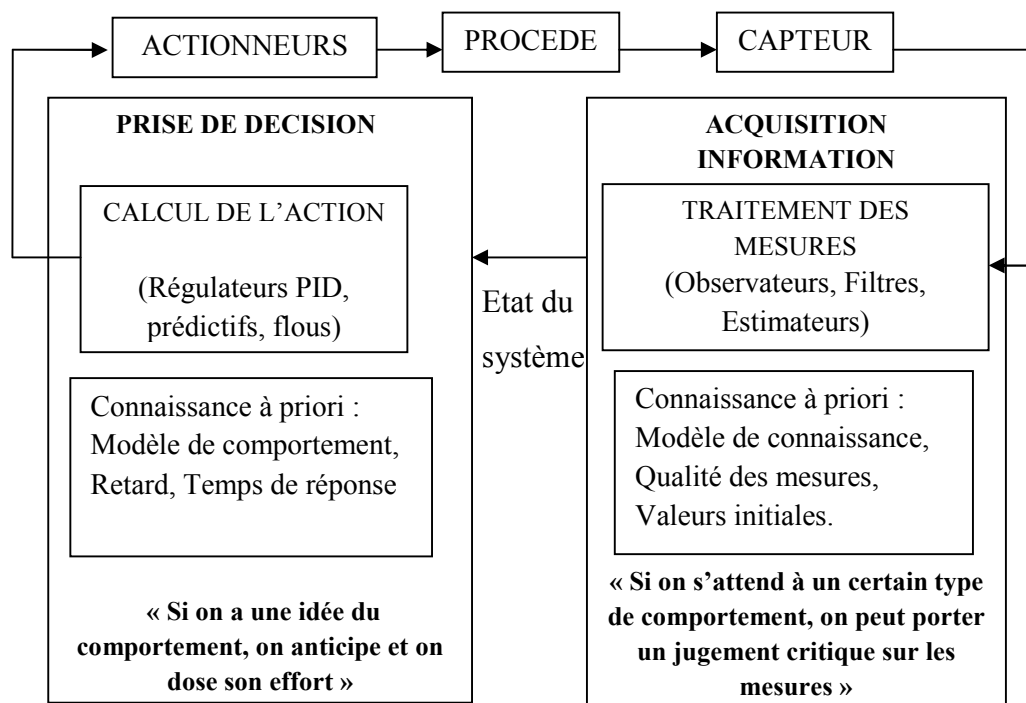


Figure 1.4 Utilisation des modèles dans la boucle de commande.

Exemple

Pour un modèle d'automobile. Si le but du modèle est d'étudier la résistance au vent du véhicule, il suffit de prendre en compte la forme extérieure du véhicule et l'état des diverses surfaces qui le caractérisent. Si on s'intéresse à la résistance du véhicule en cas de choc, le modèle sera plus complexe et devra faire intervenir les caractéristiques géométriques et mécaniques de la carrosserie, mais également du châssis et de l'ensemble de la structure interne de la voiture.

Lorsque l'objectif est de prévoir le comportement du véhicule en fonctionnement, le modèle pourra être simplifié du point de vue de la plupart des structures qui pourront être supposées rigides, mais par contre on devra tenir compte des caractéristiques du moteur, des amortisseurs, etc.

On voit que les modèles utilisés dans les divers cas envisagés peuvent être très différents.

1.4 Différents types de modèles [2]

La modélisation d'un phénomène peut s'effectuer selon des degrés de connaissances divers. L'utilisateur souhaite ce degré de connaissance le plus élevé possible, mais la complexité de la réalité physique permet au mieux d'obtenir un modèle approximatif ; ceci nous conduit à classer les modèles.

1.4.1 Modèles de connaissance

Les modèles les plus riches d'information sont appelés des **modèles de connaissance** ou bien modèles « boîte blanche ». Ils sont obtenus à partir des lois de la physique et apportent une connaissance complète sur les variables internes au système. Par exemple, pour un circuit électrique constitué d'éléments passifs tels que résistances, condensateurs et bobines d'induction, les lois d'Ohm permettent de connaître les intensités des courants ainsi que les différences de potentiel pour tous les composants, quelle que soit la structure du circuit. Les paramètres du modèle ont une interprétation physique : résistance en ohm, capacité en microfarad et inductance en milli henry. On serait tenté dans ce cas précis de

parler de modèles exacts. Il faut bien s'en garder, car les lois d'Ohm sont établies dans un cadre d'hypothèses. Il ne faut pas oublier que la valeur ohmique d'une résistance varie avec sa température, qui dépend elle-même de l'intensité parcourue ainsi que des conditions de dissipation. Une bobine d'induction a une valeur résistive en plus de son coefficient d'induction, les liaisons entre les composants présentent nécessairement une faible résistance électrique.

1.4.2 Modèles de représentation et modèles de conduite

Autour d'un point de fonctionnement, de petites variations de l'entrée d'un système peuvent être reliées à de petites variations de la sortie par un modèle dynamique linéaire qui s'écrit :

$$\begin{aligned} a_n \frac{d^n y(t)}{dt^n} + a_{n-1} \frac{d^{n-1} y(t)}{dt^{n-1}} + \dots + a_1 \frac{d y(t)}{dt} + a_0 y(t) \\ = b_m \frac{d^m u(t)}{dt^m} + b_{m-1} \frac{d^{m-1} u(t)}{dt^{m-1}} + \dots + b_1 \frac{d u(t)}{dt} + b_0 u(t) \end{aligned} \quad (1.0)$$

Où le coefficient n est l'ordre du système.

Cette équation est linéaire à coefficients constants. Elle permet de modéliser la réaction de la sortie d'un système en fonction de l'action qui lui est appliquée. Avec cette approche, on ne cherche pas à expliquer les phénomènes physiques ou chimiques qui interviennent.

L'objectif est simplement de représenter le comportement dynamique entrée/sortie d'un système avec un outil mathématique qu'on sait exploiter : on parle de modèles de comportement ou boîte noire. Ce modèle peut être obtenu par deux approches :

- La première consiste à écrire le modèle exact du système à partir des lois de la physique. Si celui-ci est linéaire, il est possible par substitution de se ramener à une équation de la forme de celle de (1.0). Si celui-ci est non linéaire, on peut le linéariser autour d'un point de fonctionnement pour obtenir une équation linéaire.

- La seconde méthode suppose la structure du modèle connu à priori. Les coefficients a_i et b_i sont calculés à partir d'une observation expérimentale de la réponse du système à une séquence d'entrée donnée $u(t)$. Cette approche, appelée *identification du modèle*.

1.4.2.1 Modèles de règles

Ils correspondent à une description par règles de conduite, comme « si on actionne A deux fois, B avance » et sont issus de l'observation du système en fonctionnement. Souvent lourds à manier et limités du point de vue possibilités, ils se prêtent bien à une exploitation en œuvre les techniques de l'intelligence artificielle.

1.4.2.2 Modèles fichiers

Ils sont constitués d'informations, sous forme de tableaux de données, reliant l'évolution des sorties du processus à l'évolution des entrées pour diverses classes d'entrées. Ces modèles constituent le plus souvent le point de départ en vue de l'élaboration de modèles plus évolués (en analyse structurale, ce niveau s'appelle le « système source »).

1.4.2.3 Modèles entrées-sorties

Dans ce type de représentation, les entrées et sorties du processus sont liées par un ensemble de relations mathématiques (en analyse structurale, c'est le niveau « système générateur »). Cet ensemble peut être composé de relations algébriques, d'équations différentielles, et de relations récurrentes. A ces relations de type égalité, peuvent se rajouter des relations de type inégalité ou inclusion, permettant de tenir compte de contraintes ou d'éventuelles saturations. Dans le cas de processus linéaires stationnaires à état continu, les relations entrées-sorties peuvent être définies par des **matrices de transfert** (en s ou en z). Les modèles entrées/sorties correspondent aux représentations du type « **boîte noire** ».

1.4.2.4 Modèles d'état

Ils sont caractérisés par un ensemble de variables, en nombre minimum, regroupées dans un vecteur \mathbf{x} de \mathbb{R}^n appelé **vecteur état**, dont la connaissance à l'instant $t_0 \in \mathbb{R}$ associée à la connaissance de l'évolution des entrées $\mathbf{u} \in \mathbb{R}^l$ sur l'intervalle $T = [t_0, T]$ permet, à partir du modèle, de prévoir l'évolution de $\mathbf{x}(t)$ sur T . dans l'hypothèse déterministe, il existe alors une fonction $\varphi(t, t_0, \mathbf{x}(t_0), \mathbf{u}[t_0, t])$ appelée **fonction de transfert d'état** telle que :

$$\mathbf{x}(t) = \varphi(t, t_0, \mathbf{x}(t_0), \mathbf{u}[t_0, t]), \quad (1.1)$$

Avec les propriétés :

$$\begin{aligned} \varphi(t, t, \mathbf{x}(t), \mathbf{u}[t_0, t]) &= \mathbf{x}(t), \\ \varphi(t, t_0, \mathbf{x}(t_0), \mathbf{u}[t_0, t]) &= \varphi(t, t_1, \varphi(t_1, t_0, \mathbf{x}(t_0), \mathbf{u}[t_0, t]), \mathbf{u}[t_1, t]), \forall t_1 \in [t_0, t], \end{aligned} \quad (1.2)$$

Un tel modèle est appelé **équation d'état sous forme explicite**, par opposition aux modèles de types équations différentielles ou récurrentes qui sont des équations d'état sous forme implicite. Les systèmes à paramètres distribués peuvent s'interpréter comme des systèmes de vecteur état de dimension infinie. Il est important de noter que les systèmes physiques sont non anticipatifs et ne peuvent pas dépendre d'événements futurs.

Dans l'ensemble \mathbb{R}^n , certains états peuvent ne pas exister, ce qui conduit à définir l'**ensemble des états admissible** $X \subset \mathbb{R}^n$ comme l'ensemble des états X possibles. Par exemple, pour un registre de 4 bits, 16 états seulement sont possibles. Si cet ensemble X est compact, le système est à **état continu**. Si X est dénombrable (donc non compact), on a un **système à état discret**. Si le nombre d'éléments de X est fini, le système est à **états finis**. C'est le cas en pratique pour tous les systèmes informatiques à états discrets, du fait des saturations de capacité mémoire.

Enfin, les **systèmes échantillonnés** correspondent à des systèmes à états continus observés ou commandés à des instants dénombrables. En pratique, ces instants sont très souvent générés par une (ou plusieurs) horloge(s) régulière(s), ce qui rend l'échantillonnage périodique. Ceci peut se résumer par tableau suivant :

$t \backslash x$	Système à états continus	Système à états discrets
Temps continu	Système continu Exemple : Equation différentielle	Exemple : Système asynchrone
Temps discret	Système discret Exemple : Equation de récurrence	Exemple : Système synchrone

« automatique »
« logique »

Table 1.1 Différents types de modèle d'état.

1.4.2.5 Modèles graphiques :

Les modèles graphiques constituent un mode de représentation en général assez aisé à manipuler et possédant des propriétés remarquables d'aider à la modélisation et parfois de permettre de la valider. Les principaux types de modèles graphiques sont :

1.4.2.5.1 Les schémas fonctionnels

Ce mode de description appelé également représentation par schéma-blocs est souvent utilisé de façon intuitive. Il correspond à une description directe des divers éléments du processus étudié (**Figure. 1.5** et **1.6**), faisant ou non intervenir les diverses relations mathématiques mises en œuvre.

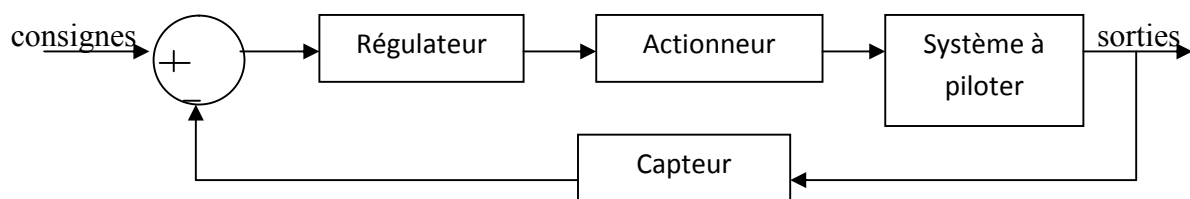


Figure 1.5 Représentation par schéma-blocs.

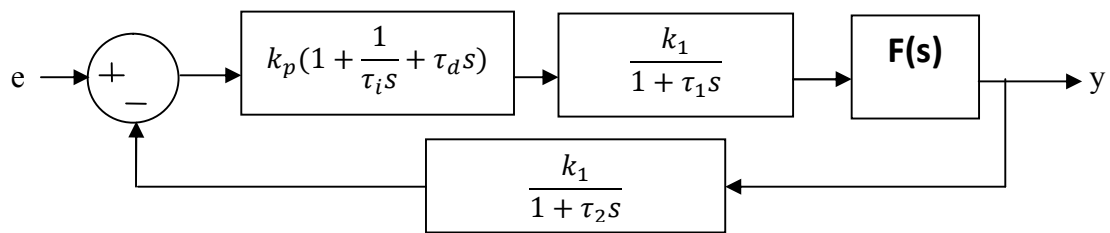


Figure 1.6 Représentation de processus.

1.4.2.5.2 Les graphes de fluence

Ils correspondent à une représentation proche de celle des schémas fonctionnels (**Figure 1.6**) (ces deux représentations sont en effet duales l'une de l'autre).

L'intérêt des graphes de fluence apparaît principalement au niveau de la modélisation des systèmes linéaires. C'est-à-dire pour lesquels il est possible d'appliquer le théorème de superposition défini plus loin. (**P 16**)

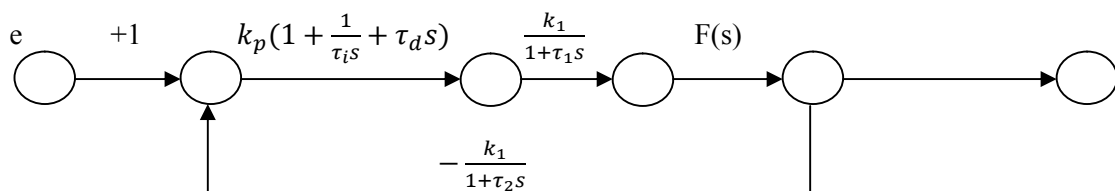


Figure 1.7 Graphe de fluence.

1.4.2.5.3 Les réseaux de Petri et les Grafccets

Ces représentations sont particulièrement adaptées aux processus à événements discrets dont le nombre d'états est fini. Très importants, principalement dans la caractérisation des systèmes logiques, de fabrication automatisée et des ateliers flexibles.

1.4.2.5.4 Les bond-graphs

Ils permettent une description des systèmes physiques (mécaniques, électriques, hydrauliques....) très bien adaptée à la modélisation des transferts de puissance, avec un langage unique quelque soit le domaine physique concerné.

Exemple (modèle de connaissance)

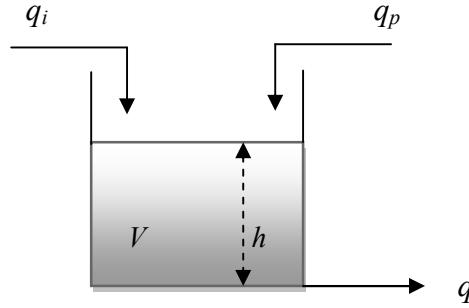


Figure 1.8: Bac de stockage avec entrée perturbante

Considérons le système de la **Figure 1.8**. Si on suppose que le débit q est exprimé par la loi de Bernoulli, c'est-à-dire lié à la hauteur du liquide. Le modèle de connaissance de ce système s'écrit comme suit :

$$S \frac{dh}{dt} = -C_v \sqrt{\rho \frac{g}{g_c}} h + q_i + q_p \quad (1.4)$$

Avec S est la surface de la base du réservoir, h est la hauteur du liquide, ρ c'est sa masse volumique, g est la force de gravité et C qui est la constante de Bernoulli.

Ce modèle peut se linéariser autour d'un point de fonctionnement (h_0, q_0, p_0) :

$$S \frac{d(\delta h)}{dt} = -C_v \sqrt{\rho \frac{g}{g_c}} \frac{1}{2\sqrt{h_0}} \delta h + \delta q_i + \delta q_p \quad (1.5)$$

On s'intéressant qu'à l'effet de l'entrée q seulement, et considérant qu'il n'y a pas de perturbation. Le modèle boîte noire de l'action d'une variation de débit sur une variation de hauteur s'écrit :

$$\frac{dy}{dt} + a.y = u \quad (1.6)$$

en prenant les notations classiques $\delta h = y$ et $\delta q = u$ et en posant :

$$a = C_v \sqrt{\rho \frac{g}{g_c} \frac{1}{2\sqrt{h_0}}} \quad (1.7)$$

Remarque 1.2

*La modélisation d'un système complexe conduit quelquefois à un **modèle mixte**. Certaines parties sont décrites par un modèle de connaissance, mais pour d'autres un modèle de comportement est utilisé. Prenons le cas, dans l'industrie automobile, d'une fonction appelée « liaison au sol ». Les lois de la mécanique permettent de décrire les efforts, les couples et les vitesses dans les parties roue et suspension. Il faut enfin décrire l'interaction entre la roue et le sol pour expliquer le comportement du véhicule. Cette interaction dépend des caractéristiques du pneu (état d'usure, gonflage) ainsi que de celles du sol (granulométrie, adhérence, humidité). Cette partie reste souvent un modèle de comportement. On parle alors de « boîte grise » pour qualifier le modèle obtenu.*

1.5 Obtention du modèle [2]

1.5.1 Modélisation mathématique

La détermination d'un modèle mathématique d'un processus nécessite en général diverses hypothèses simplificatrices afin de limiter sa complexité. L'étude s'intéresse essentiellement aux processus pour lesquels les variables caractéristiques sont susceptibles de prendre un ensemble continu de valeurs appartenant à des intervalles fixés. Comme elle traite que des modèles de processus à paramètres localisés, c'est-à-dire, décrit par des équations

différentielles ordinaires ou des équations récurrentes. Il est important de toujours se souvenir que le choix du modèle d'un processus dépend de l'utilisation prévue pour ce modèle.

1.5.1.1 Modèle à temps continu

Dans ce type de modèle, le temps est une variable qui évolue continûment sur un intervalle $\tau \subset \mathfrak{R}$ prenant en croissant toutes les valeurs situées dans cet intervalle. C'est le cas, par exemple d'un modèle dans l'espace d'état de la forme :

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, t, \mathbf{v}), \quad (1.8)$$

$$\mathbf{y} = h(\mathbf{x}, \mathbf{u}, t, \mathbf{v}),$$

Modèle dans lequel $\dot{\mathbf{x}}$ représente la dérivée totale $\frac{d\mathbf{x}}{dt}$ du vecteur état $\mathbf{x} \in \mathfrak{R}^n$ par rapport au temps, $\mathbf{u} \in \mathfrak{R}^l$ le vecteur des commandes, $t \in \tau$ le temps, $\mathbf{v} \in \mathfrak{R}^{n_v}$ un vecteur de perturbations, et $\mathbf{y} \in \mathfrak{R}^m$ le vecteur des sorties.

Dans cette représentation, on a :

$$\begin{aligned} f: \mathfrak{R}^n \times \mathfrak{R}^l \times \tau \times \mathfrak{R}^{n_v} &\longrightarrow \mathfrak{R}^n, \\ h: \mathfrak{R}^n \times \mathfrak{R}^l \times \tau \times \mathfrak{R}^{n_v} &\longrightarrow \mathfrak{R}^m, \\ \tau = (t_0, t_0 + T_e), t_0 \in \mathfrak{R}, t_0 \ll \infty, T_e \in \mathfrak{R}^+. \end{aligned}$$

Dans beaucoup de problèmes, on prend $t_0 = 0$ ou $t_0 > 0$ mais cette restriction n'est en aucun cas nécessaire a priori.

1.5.1.2 Modèles à temps discontinu ou discret

Ce type de modèle est utilisé soit lorsque le processus étudié a naturellement un mode d'évolution séquentiel, soit lorsqu'on a adopté une représentation du processus qui correspond à une observation des variables d'état ou de sortie à des instants discrets t_k du temps avec k

$\in \mathbb{Z}$. On a dans ce cas une représentation dite **échantillonnée**, particulièrement bien adaptée aux divers types de calculs à effectuer dans les problèmes de commandes de processus par calculateur numérique.

En notant :

$$\mathbf{x}_k = \mathbf{x}(t_k), \quad (1.9)$$

Il vient une description de la forme :

$$\begin{aligned} \mathbf{x}_{k+1} &= f(\mathbf{x}_k, \mathbf{u}_k, k, \mathbf{v}_k), \\ \mathbf{y}_k &= h(\mathbf{x}_k, \mathbf{u}_k, k, \mathbf{v}_k), \end{aligned} \quad (1.10)$$

1.5.1.3 Modèle linéaire

Un tel modèle caractérise un processus susceptible d'être décrit par une équation différentielle ordinaire ou une équation récurrente à coefficients constants (**stationnaire**) ou fonction du temps (non **stationnaire**) ou par un ensemble d'équations de ce type.

La propriété fondamentale des systèmes linéaires s'exprime par le **principe de superposition** : si $\mathbf{y}_1(t)$ et $\mathbf{y}_2(t)$ représentent respectivement l'évolution des sorties du processus pour les entrées $\mathbf{u}_1(t)$ et $\mathbf{u}_2(t)$ sur un horizon et pour un état initial donné, à l'entrée $\mathbf{u}(t)$:

$$\mathbf{u}(t) = \alpha_1 \mathbf{u}_1(t) + \alpha_2 \mathbf{u}_2(t) \quad (1.11)$$

Où α_1 et α_2 sont réels, correspond la sortie :

$$\mathbf{y}(t) = \alpha_1 \mathbf{y}_1(t) + \alpha_2 \mathbf{y}_2(t) \quad (1.12)$$

Sur le même horizon et pour les mêmes conditions initiales. Les systèmes linéaires stationnaires admettent une représentation entrée-sortie par fonction ou matrice de transfert, ou selon qu'il s'agit d'un modèle continu ou discret.

1.5.1.4 Modèle non linéaire

Dans un modèle non linéaire, le théorème de superposition n'est plus valable. L'aspect non linéaire peut être intrinsèque et quasiment irréductible, comme dans la modélisation de la loi d'action de masse en chimie, ou correspondre à l'association d'éléments à caractéristiques non linéaire à des systèmes pour lesquels un modèle est utilisable comme c'est le cas des systèmes à non linéarités séparables (**Figure 1.9**).

Dans certains cas, lorsque le processus non linéaire est utilisé dans une plage de variation limitée de ses variables d'état, l'évolution s'effectuant autour de ce point de fonctionnement en prenant comme nouvelle variable l'expression $\delta x = x - x_0$.

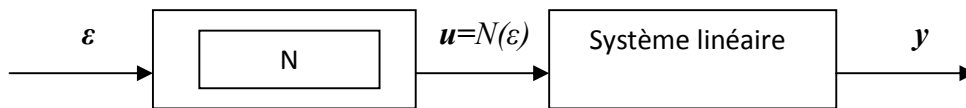


Figure 1.9 Système à non linéarité séparable.

On néglige alors dans un développement limité de la fonction $f(\cdot)$ les termes d'ordre supérieur à 1. Il vient par exemple pour la relation :

$$\dot{x} = f(x, u, t), \quad (1.13)$$

Si la relation f admet des dérivées partielles continues au premier ordre, le modèle linéarisé (1.14) valable pour (x, u) évoluant dans un voisinage de (x_0, u_0) :

$$\frac{d}{dt}(\delta x) = F_x(x_0, u_0, t)\delta x + F_u(x_0, u_0, t)\delta u, \quad (1.14)$$

Avec $\delta x = x - x_0$, $u = u - u_0$, et où :

$F_x = \left\{ \frac{\partial f_i}{\partial x_j} \right\}$, $F_u = \left\{ \frac{\partial f_i}{\partial u_j} \right\}$ Sont les matrices Jacobiennes de f en x et u .

Il est également possible d'effectuer une linéarisation autour d'une trajectoire d'évolution $(x_N(t), u_N(t))$ donnée solution de (1.13) .

1.5.2 Modélisation expérimentale

L'identification constitue une phase importante dans la définition du modèle, c'est par elle que le choix de la classe de modèles à adopter puis les valeurs des paramètres qui le caractérisent vont se préciser.

Le plus souvent, l'identification s'effectue en optimisant un critère de qualité qui caractérise l'écart entre le comportement du processus (repéré par un ensemble de mesures), et celui de son modèle (étudié par simulation) pour un ensemble de sollicitations données.

De nombreuses méthodes d'identification sont précisées dans la suite (chapitre 2) correspondant le plus souvent à l'un des schémas des (**Figures 1.10 et 1.11**).

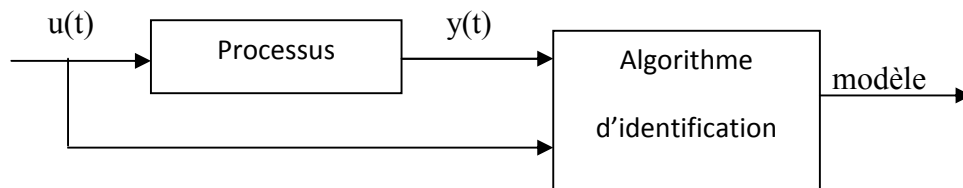


Figure 1.10 Identification à partir du comportement entrée-sortie.

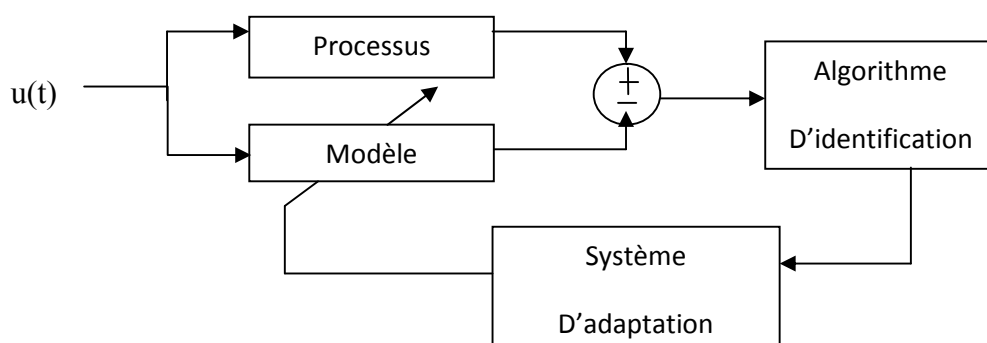


Figure 1.11 Identification de type paramétrique.

Conclusion

Dans ce chapitre, on a présenté des généralités sur les modèles mathématiques et les étapes de la conception d'un tel modèle. Le chapitre suivant sera consacré à la modélisation expérimentale des systèmes, c'est-à-dire à l'identification des processus et à sa mise en œuvre dans le cas du modèle de comportement.

CHAPITRE 2:

Identification des systèmes

Introduction

Ce chapitre donne d'abord les principes de bases de l'identification des modèles dynamiques de procédés. Il classe ensuite les différentes méthodes d'identification. Dans la dernière partie, le chapitre présente les étapes et l'aspect pratique de mise en œuvre dans la commande des procédés dynamiques industriels.

2.1 Généralités [4]

2.1.1 Définition

L'identification, c'est l'opération de détermination des caractéristiques dynamiques d'un procédé (système) dont la connaissance est nécessaire pour la conception et la mise en œuvre d'un système de commande performant. L'identification est une technique expérimentale qui s'appuie sur l'utilisation de procédures et algorithmes issus d'études théoriques.

2.1.2 Objectif

Lors de l'écriture d'un modèle, qu'il soit de connaissance ou de comportement, on est amené à introduire certains coefficients appelés paramètres. Ceux-ci peuvent avoir une signification physique, comme dans les modèles de connaissance, ou ne pas en avoir, comme c'est le cas pour les modèles de comportement. Dans les deux cas, leur valeur numérique doit souvent être déterminée à partir de données expérimentales.

Cependant, la détermination des paramètres a des objectifs différents suivant qu'on considère un modèle de connaissance ou un modèle « boîte noire ». Les termes utilisés pour désigner cette étape sont d'ailleurs différents : on parle plutôt *d'estimation de paramètres* dans le premier cas, et *d'identification* dans le second. En effet, l'objectif de l'identification est de calculer les paramètres d'un modèle du procédé à partir de données expérimentales, de façon à ce que le comportement du procédé et celui du modèle soient identiques, et ceci, pour toutes les séquences de variables d'entrées habituellement utilisées. Par conséquent :

- il ne s'agit pas de trouver un modèle du procédé ayant une signification physique, mais seulement un *modèle qui représente son comportement au mieux*.
- les *paramètres* ont pour seul rôle de permettre un ajustement du modèle sur les données expérimentales. Il doit être possible de les déterminer uniquement à partir du comportement entrées-sorties du procédé.
- les données expérimentales doivent être collectées dans des conditions représentatives, ce qui signifie en particulier que les *séquences appliquées sur la ou les variables manipulées* doivent être choisies de façon à obtenir une réponse dynamique du comportement du système.

2.2 Méthodes d'identification [3]

Cette partie présente les différentes méthodes pour calculer les paramètres d'un modèle dynamique de comportement. Elles sont classées en deux catégories : les méthodes graphiques ou bien classiques basées sur une réponse indicielle, et les méthodes numériques basées sur les moindres carrés.

2.2.1 Méthodes graphiques

2.2.1.1 Objectifs

La réponse d'un système à un échelon ou à une impulsion sur la variable manipulée fournit d'intéressants renseignements sur sa dynamique. Cela permet en particulier d'obtenir une valeur approximative du retard, du gain et du temps de réponse. Si en théorie la réponse indicielle et la réponse impulsionnelle sont équivalentes, ce n'est pas le cas en pratique. En effet, il est difficile de distinguer parmi les perturbations diverses, l'effet d'une impulsion sur la variable manipulée. En règle générale, on préférera donc la réponse indicielle. Le choix de la structure du modèle à partir de la réponse indicielle s'effectue comme suit :

1. La réponse présente-t-elle un retard ?
2. Le processus est-il stable ou de type intégrateur ?
3. La réponse présente-t-elle un dépassement ou est-elle apériodique ?

2.2.1.2 Systèmes apériodiques sans retard

2.2.1.2.1 Recherche d'un modèle du premier ordre

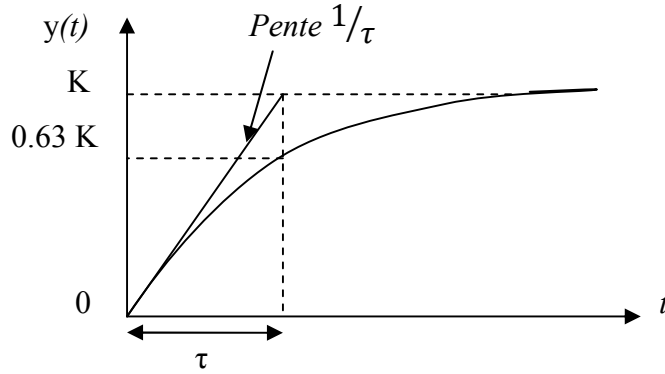


Figure 2.1 Identification directe de la réponse d'un système du premier ordre

Face à une réponse indicielle stable, sans retard et sans dépassement, l'observation de la tangente à l'origine permet de déterminer si on a affaire à un système du premier ordre ou non. En effet dans le cas d'un premier ordre, la tangente ne doit pas être nulle (**Figure 2.1**), et devrait être idéalement, de pente $1/\tau$. Comme cette pente peut être difficile à déterminer avec précision, on tracera conjointement la droite d'ordonnée $0,63.K$ parallèle à l'axe des abscisses. Cette construction permet de déterminer la constante de temps τ .

Une autre approche pour calculer les paramètres d'un modèle du premier ordre consiste à tracer $y_c(t + T) = K - y(t + T)$ en fonction de $y_c(t) = K - y(t)$, où K est le gain, déterminé en prenant la valeur asymptotique de la réponse, et T est un intervalle de temps donné. Il est facile de montrer que les couples de points $y_c(t), K - y_c(t + T)$ définissent une droite :

$$y(t) = K(1 - e^{-\frac{t}{\tau}}) \quad (2.1)$$

$$y_c(t) = K e^{-\frac{t}{\tau}} \quad \text{et} \quad y_c(t + T) = K e^{-\frac{t+T}{\tau}} \quad (2.2)$$

D'où :

$$y_c(t + T) = e^{-\frac{T}{\tau}} y_c(t) = a y_c(t) \quad (2.3)$$

Par conséquent, en mesurant la pente de la droite, on peut facilement calculer la constante de temps du modèle :

$$\tau = -\frac{T}{\text{Log}(a)} \quad (2.4)$$

2.2.1.2.2 Recherche d'un modèle du deuxième ordre apériodique

De façon analogue, on peut montrer que, pour un modèle du deuxième ordre, la sortie obéit à l'équation :

$$y(t) + a_1 y_c(t - T) + a_2 y_c(t - 2T) = 0 \quad (2.5)$$

Le principe de la méthode consiste donc à tracer $y_c(t - T)/y_c(t)$ en fonction de $y_c(t - 2T)/y_c(t)$. Ces points sont sur la droite d'équation :

$$y = -\frac{a_1}{a_2}x - \frac{1}{a_1} \quad (2.6)$$

Ces constantes de temps τ_1 et τ_2 sont obtenues à partir des racines x_1, x_2 de l'équation :

$$a_2 x^2 + a_1 x + 1 = 0 \quad (2.7)$$

Par les relations :

$$\tau_1 = \frac{Te}{\text{Log}(x_1)} \quad , \quad \tau_2 = \frac{Te}{\text{Log}(x_2)} \quad (2.8)$$

2.2.1.3 Systèmes apériodiques avec retard

2.2.1.3.1 Recherche d'un modèle du premier ordre : méthode de Broïda

Dans un grand nombre de procédés, la réponse présente un certain retard. Dans ce cas, on cherche le plus souvent à identifier un modèle du premier ordre du type :

$$G(p) = \frac{K.e^{-r.p}}{1+\tau.p} \quad (2.9)$$

La première solution consiste à tracer la tangente à la courbe en son point d'inflexion, puis à mesurer le retard r et la constante de temps τ . Une autre approche, appelée méthode de Broïda (**Figure 2.2**), consiste à mesurer les instants t_1 et t_2 auxquels la réponse atteint respectivement 28% et 40% de sa valeur finale. La constante de temps et le retard sont alors obtenus par les relations suivantes :

$$\tau \cong 2,8 t_{28\%} - 1,8 t_{40\%} \quad , \quad r \cong 5,5(t_{40\%} - t_{28\%}) \quad (2.10)$$

Une approche analogue consiste à considérer les instants auxquels la réponse atteint 35,3% et 85,3% qui présentent l'intérêt d'être plus nettement séparés que dans le cas précédent. On a alors :

$$\tau \cong 1,3 t_{35,3\%} - 0,29 t_{85,3\%} \quad , \quad r \cong 0,67(t_{85,3\%} - t_{35,3\%}) \quad (2.11)$$

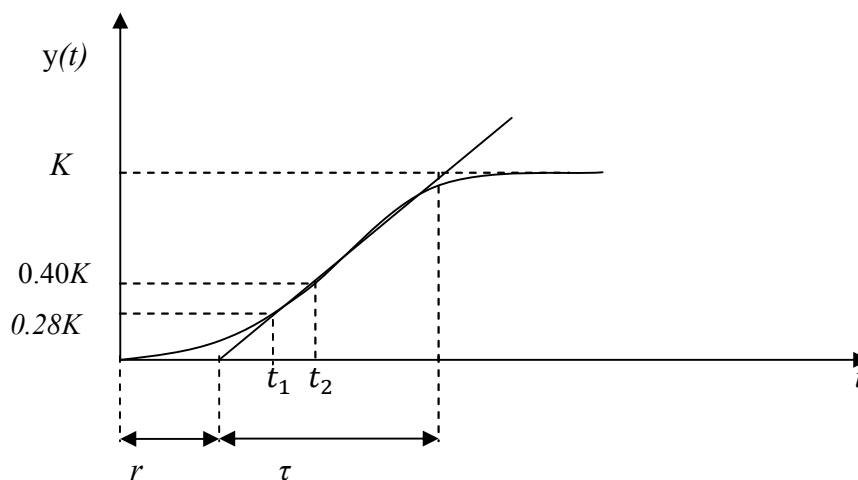


Figure 2.2 Méthode de Broïda

2.2.1.3.2 Recherche d'un modèle d'ordre supérieur (Méthode de Strejc)

La méthode de Strejc consiste à caractériser le procédé par un modèle de la forme :

$$G(p) = \frac{K e^{-rp}}{(1+\tau p)^n} \quad (2.12)$$

Ce type de modèle ne pourra convenir que pour une réponse ayant une allure telle que celle présentée sur la **(Figure 2.3)**. Le principe de la méthode est le suivant :

1. Tracer la tangente au point d'inflexion : cette tangente permet de définir les deux grandeurs T_u et T_a .
2. On calcule le rapport

$$\eta = \frac{T_u}{T_a} \quad (2.13)$$

3. On cherche dans la **Table 2.1** le rapport T_u/T_a immédiatement inférieur à la valeur calculée η . Cette ligne permet d'obtenir l'ordre n du modèle. La constante de temps est calculée à partir de la troisième colonne :

$$\tau = \left. \frac{\tau}{T_a} \right|_{table} T_{a \text{ mesuré}} \quad (2.14)$$

Et le retard est égal à :

$$r = T_{u \text{ mesuré}} - \left. \frac{T_u}{T_a} \right|_{table} T_{a \text{ mesuré}} \quad (2.15)$$

Cette méthode est assez sensible à la détermination graphique du point d'inflexion de la réponse, ce qui n'est pas toujours facile lorsque celle-ci est bruitée.

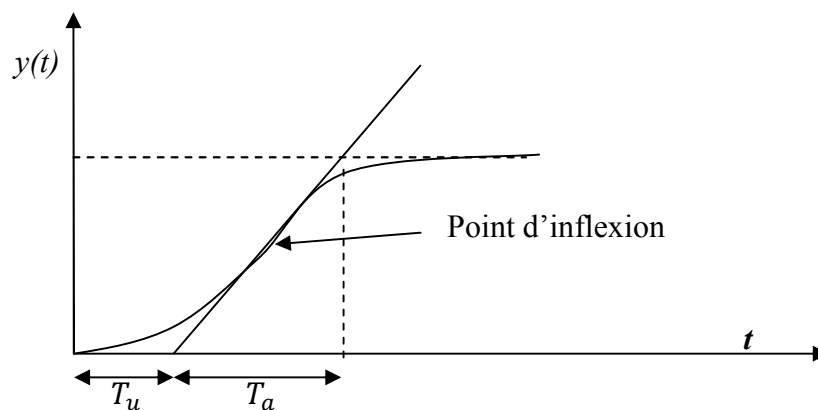


Figure 2.3 Méthode de Strejc

$\frac{T_u}{T_a}$	Ordre du modèle n	$\frac{\tau}{T_a}$
0	1	1
0,105	2	0,37
0,22	3	0,27
0,32	4	0,22
0,41	5	0,20
0,49	6	0,19
0,57	7	0,18
0,64	8	0,15
0,71	9	0,14
0,77	10	0,13

Table 2.1 Coefficients de la méthode de Strejc.

2.2.1.4 Méthode des moments

Dans cette méthode, on transforme toujours la fonction de transfert du procédé sous la forme suivante :

$$G(p) = \frac{K}{1+Tp} e^{-Lp} \quad (2.16)$$

Puis on calcule les paramètres à l'aide des relations suivantes :

$$K = G(0), \quad T_{ar} = -\frac{G'(0)}{G(0)} = L + T, \quad T^2 = \frac{G''(0)}{G(0)} - T_{ar}^2 \quad (2.17)$$

2.2.1.5 Systèmes oscillants

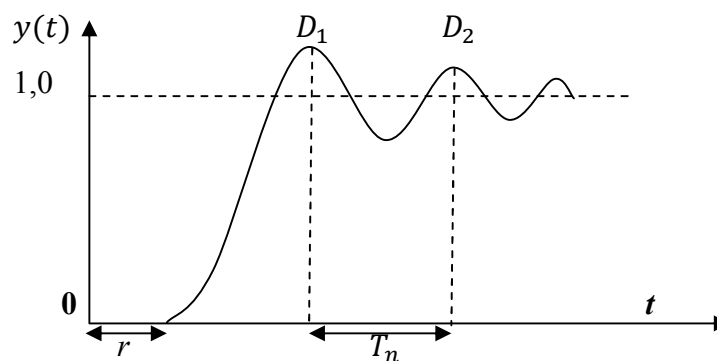


Figure 2.4 Réponse indicielle oscillante.

Si la réponse présente un caractère oscillant, on pourra chercher à identifier un modèle de type deuxième ordre avec retard :

$$G(p) = \frac{K e^{-r p} \omega_n^2}{p^2 + 2\zeta \omega_n p + \omega_n^2} \quad (2.18)$$

On cherchera à mesurer le premier et le deuxième dépassement D_1 et D_2 (différence entre le maximum de la courbe et la valeur finale), ainsi que les instants auxquels ils ont lieu t_1 et t_2 (**Figure 2.4**). On pourra ensuite calculer les coefficients de transfert :

$$\omega_n = \frac{2\pi}{t_1 - t_2} \frac{1}{\sqrt{1 - \zeta^2}} \quad \text{et} \quad \frac{D_1}{D_2} = e^{-\frac{2\pi\zeta}{\sqrt{1 - \zeta^2}}} \quad (2.19)$$

Cette méthode est relativement sensible au bruit de mesure et on doit réaliser un filtrage des mesures.

2.2.1.6 Systèmes intégrateurs

2.2.1.6.1 Système intégrateur pur

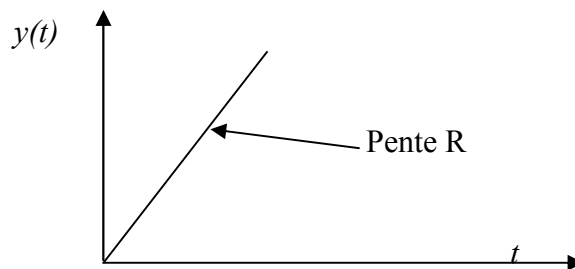


Figure 2.5 Système intégrateur d'ordre 1

Un système intégrateur est caractérisé par le fait que sa sortie augmente jusqu'à saturation de façon linéaire, alors que l'entrée reste constante (**Figure 2.5**). Un intégrateur pur est décrit par l'équation :

$$G(p) = \frac{K}{p} \quad (2.20)$$

L'identification graphique d'un tel modèle se réduit à la détermination du gain, qui est égal à la pente R de la réponse. Pour rappel, on considère la réponse à une entrée unitaire. Si ce n'est pas le cas, il ne faut pas oublier de diviser la pente par l'amplitude de l'entrée pour obtenir le gain.

2.2.2 Méthodes numériques

Ces méthodes permettent de calculer les paramètres d'un modèle dynamique numériquement, et cela, à partir d'un ensemble de valeurs expérimentales. Donc, cette approche fournit des solutions analytiques. Elles sont représentées en trois classes :

- La méthode des moindres carrés : elle n'est pas la meilleure, mais c'est la plus simple, car elle fournit une solution analytique.
- La méthode du modèle : cette méthode est nettement meilleure que la précédente, et nécessite moins de précautions de la part de l'utilisateur. La contrepartie est qu'elle est plus difficile à mettre en œuvre, car la solution doit être calculée par optimisation numérique.
- Les méthodes récursives : elles sont la base de la plus part des régulations adaptatives modernes. Il est donc intéressant d'en connaître le principe.

2.2.2.1 Méthode des moindres carrés [5]

Avant de passer à l'étude de la méthode, il est important de faire la distinction entre l'identification en temps réel et en temps différé. L'identification en temps différé ne nécessite pas la présence d'un calculateur en ligne, donc l'identification est faite après avoir pris les données sur le procédé, par contre l'identification en temps réel, met à jour le modèle au fur et à mesure que les données sont acquises.

2.2.2.1.1 Optique temps différé

Le principe des moindres carrés, selon Gauss, est de choisir les paramètres d'un modèle de telle sorte que la somme des carrés de la différence entre les valeurs prédites par le modèle

et celles observées, éventuellement multipliées par un coefficient caractérisant le degré de précision, soit minimale. Le modèle retenu a pour équation :

$$y(t) + a_1 y(t-1) + \dots + a_n y(t-n) = b_0 u(t) + \dots + b_p u(t-p) + e(t) \quad (2.21)$$

2.2.2.1.2 Critère à minimiser et résolution

Avec N mesures de l'entrée et de la sortie, entre les instants 1 et N , on peut écrire :

$$\begin{aligned} y(N) &= -a_1 y(N-1) - \dots - a_n y(N-n) + b_0 u(N) + \dots + b_p u(N-p) + e(N) \\ y(N-1) &= -a_1 y(N-2) - \dots - a_n y(N-n-1) + b_0 u(N-1) + \dots \\ &\quad + b_p u(N-p-1) + e(N-1) \\ &\vdots \\ y(n+1) &= -a_1 y(n) - \dots - a_n y(1) + b_0 u(n+1) + \dots + b_p u(n-p+1) + e(n+1) \end{aligned} \quad (2.22)$$

Pour que le système soit physiquement réalisable, on doit avoir : $p < n$. Ces $N-n$ équations peuvent s'écrire sous la forme matricielle :

$$\begin{bmatrix} y(N) \\ y(N-1) \\ \vdots \\ y(n+1) \end{bmatrix} = \begin{bmatrix} -y(N-1) \dots -y(N-n) & u(N) \dots u(N-p) \\ -y(N-2) \dots -y(N-n-1) & u(N-1) \dots u(N-p-1) \\ \vdots & \vdots \\ -y(n) \dots -y(1) & u(n+1) \dots u(n+1-p) \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \\ b_0 \\ \vdots \\ b_p \end{bmatrix} + \begin{bmatrix} e(N) \\ e(N-1) \\ \vdots \\ e(n+1) \end{bmatrix} \quad (2.23)$$

On voit que pour un tel modèle, n mesures de la sortie et $p+1$ mesures de l'entrée interviennent sous forme de conditions initiales et qu'avec N mesures entrées/sorties on obtient $N-n$ équations initiales nulles pour cette méthode. Donc, l'équation matricielle (2.23) peut encore s'écrire :

$$\underline{Y} = X\underline{\theta} + \underline{E} \quad (2.24)$$

Où $\underline{\theta}$ est le vecteur des paramètres à estimer.

Le critère à minimiser est le carré de l'erreur de prédiction :

$$C = \underline{E}^T \underline{E} = \sum_{t=n+1}^N e^2(t) \quad (2.25)$$

Soit :

$$\begin{aligned}
 C &= \underline{E}^T \underline{E} = (\underline{Y} - X \underline{\theta})^T (\underline{Y} - X \underline{\theta}) \\
 &= \underline{Y}^T \underline{Y} - \underline{\theta}^T X^T \underline{Y} - \underline{Y}^T X \underline{\theta} + \underline{\theta}^T X^T X \underline{\theta} \\
 &= \underline{Y}^T \underline{Y} - 2 \underline{Y}^T X \underline{\theta} + \underline{\theta}^T X^T X \underline{\theta}
 \end{aligned} \tag{2.26}$$

L'estimateur $\hat{\theta}$ qui minimise ce critère découle de la condition nécessaire suivante :

$$-\frac{\partial C}{\partial \underline{\theta}} = -2 \underline{Y}^T X + 2 \underline{\theta}^T X^T X = 0 \tag{2.27}$$

Donc :

$$\hat{\underline{\theta}} = (X^T X)^{-1} X^T \underline{Y} \tag{2.28}$$

La matrice des dérivées secondes est :

$$C_{\theta\theta} = \left\{ \frac{\partial^2 C}{\partial \theta_i \partial \theta_j} \right\} = 2 X^T X \tag{2.29}$$

C'est une matrice définie positive, symétrique. La solution $\hat{\theta}$ est donc bien un minimum. L'avantage qu'offre cette méthode sur le plan du calcul est que l'estimateur des paramètres est explicite. Son calcul suppose néanmoins que la matrice $X^T X$ soit régulière.

En effet :

- a) Le modèle choisi est linéaire par rapport aux paramètres ; c'est ce qui importe dans le calcul de l'estimateur. Il pourrait représenter un système non linéaire par rapport aux entrées/sorties, par exemple :

$$y(t) = ay(t-1) + u^2 b(t) + e(t) \tag{2.30}$$

Dans ce cas, pour le calcul de la solution (2.28), il suffit de poser :

$$X = \begin{bmatrix} y(N-1) & \cdots & u^2(N) \\ \vdots & & \\ y(1) & \cdots & u^2(1) \end{bmatrix} \quad \underline{\theta} = \begin{bmatrix} a \\ b \end{bmatrix} \tag{2.31}$$

- b) Si le système n'était pas bruité, il aurait suffi de résoudre un système à $n+p+1$ équations :

$$y(n+1) + a_1 y(n) + \dots + a_n y(1) = b_0 u(n+1) + \dots + b_p u(n-p+1) \quad (2.32)$$

$$y(2n+1+p) + a_1 y(2n+p) + \dots + a_n y(n+1+p) = b_0 u(2n+1+p) + \dots + b_p u(2n)$$

Du fait de la présence du bruit, on est amené à résoudre $N-n$ équations à $n+p+1$ inconnus, ceci afin d'obtenir un effet de moyennage du bruit.

- c) Cette méthode ne nécessite pas une entrée $u(t)$ particulière, mais on a intérêt à la choisir de façon à exciter le plus possible le système sans perturber son fonctionnement.

2.2.2.1.3 Caractéristiques de l'estimateur

Un estimateur est dit non biaisé si :

$$E.M\{\hat{\underline{\theta}}\} = \underline{\theta} \quad (2.33)$$

Il converge alors en moyenne vers la vraie valeur des paramètres, cette qualité est évidemment indispensable pour une bonne identification. On essaiera de voir si elle est vérifiée pour l'estimateur des moindres carrés.

$$\begin{aligned} E.M(\hat{\underline{\theta}}) &= E.M\{(X^T X)^{-1} X^T \underline{Y}\} \\ &= E.M\{(X^T X)^{-1} X^T (X \underline{\theta} + \underline{E})\} \\ &= E.M\{(X^T X)^{-1} X^T X \underline{\theta} + (X^T X)^{-1} X^T \underline{E}\} \\ &= \underline{\theta} + E.M\{(X^T X)^{-1} X^T \underline{E}\} \end{aligned} \quad (2.34)$$

Ainsi, l'estimateur est non biaisé si seulement : $E.M\{\underline{E}\} = 0 \quad (2.35)$

C'est-à-dire si \underline{E} est non corrélé avec X et est centré, alors : $E.M\{\hat{\underline{\theta}}\} = \underline{\theta} \quad (2.36)$

La corrélation des paramètres est donnée par la matrice de covariance de l'estimateur $\Gamma_{\theta\theta}$. Les éléments $\delta_{\theta_i\theta_j}$ de cette matrice indiquent la corrélation entre paramètres et les éléments diagonaux $\delta_{\theta_i\theta_i}$, la variance de chaque paramètre qui est une mesure de la précision de l'estimation.

$$\begin{aligned}\Gamma_{\theta\theta} &= E.M\{(\hat{\underline{\theta}} - \underline{\theta}).(\hat{\underline{\theta}} - \underline{\theta})^T\} \\ &= E.M\{(X^T X)^{-1} X^T \underline{E} \underline{E}^T X (X^T X)^{-1}\}\end{aligned}\quad (2.37)$$

A condition que le bruit soit blanc, de variance σ^2 c'est-à-dire si : $E.M\{\underline{E} \underline{E}^T\} = \sigma^2 I$ (2.38)

On démontre que :

$$\Gamma_{\theta\theta} = \sigma^2 (X^T X)^{-1} \quad (2.39)$$

Si la variance des résidus n'est pas connue, un estimateur non biaisé de cette variance est donné par :

$$\sigma^2 = \frac{1}{M-m} \hat{\underline{E}}^T \hat{\underline{E}} = \frac{1}{M-m} (\underline{Y} - X\hat{\underline{\theta}})^T (\underline{Y} - X\hat{\underline{\theta}}) \quad (2.40)$$

Où $M = N - 1$ et $m = n + p + 1$, sont les dimensions de la matrice X et $\hat{\underline{E}}$ l'estimation des résidus par le modèle. Il est essentiel de remarquer que pour l'étude des propriétés statiques de l'estimateur, il faut exiger deux conditions :

- \underline{E} centré
- \underline{E} non corrélé avec X

On peut toujours vérifier la première hypothèse en considérant un paramètre supplémentaire dans l'équation du modèle, pour centrer les résidus. Par contre, on savait que :

$$e(t) = A(Z^{-1}).b(t) \quad (2.41)$$

Les résidus sont donc toujours corrélés par l'intermédiaire de $A(Z^{-1})$. Il contient dans l'expression de $e(t)$ des termes en $b(t-1)$, $b(t-2)$, ..., $b(t-n)$. Et sur la ligne correspondante de la matrice X , on trouve les termes $y(t-1)$, $y(t-2)$, ..., $y(t-n)$, qui sont fonction de $b(t-1)$, ..., $b(t-n)$. Donc avec le modèle choisi, la corrélation est inévitable. L'estimateur $\hat{\underline{\theta}}$ obtenu par la méthode des moindres carrés est toujours biaisé. La

non corrélation ne peut être respectée que si le résidu $e(t)$ est blanc. L'hypothèse de bruit en sortie du système $b(t)$ blanc ne s'avère donc pas suffisante. Ce problème de biais de l'estimateur est un problème dû au modèle choisi plus qu'à la méthode elle-même.

Dans cette optique temps différé, la solution du problème des moindres carrés présente au moins l'avantage énorme d'être explicite et très facile à mettre en œuvre.

2.2.2.1.2 Optique temps réel

La méthode des moindres carrés en temps réel se déduit facilement de l'optique temps différé. Soit N le nombre de mesures effectuées jusqu'à l'instant $t = NT$ (T unitaire). L'estimateur des moindres carrés présentés précédemment a pour expression (2.28) :

$$\hat{\underline{\theta}}_N = (X_N^T X_N)^{-1} X_N^T \underline{Y}_N \quad (2.42)$$

Si on acquiert une mesure supplémentaire, cet estimateur devient :

$$\hat{\underline{\theta}}_{N+1} = (X_{N+1}^T X_{N+1})^{-1} X_{N+1}^T Y_{N+1} \quad (2.43)$$

$$\text{Avec :} \quad X_{N+1} = \begin{bmatrix} X_N \\ X(N+1) \end{bmatrix} \quad Y_{N+1} = \begin{bmatrix} Y_N \\ Y(N+1) \end{bmatrix} \quad (2.44)$$

$$X(N+1) = [-y(N) \dots -y(N-n+1) \quad u(N+1) \dots u(N+1-p)] \quad (2.45)$$

$$\begin{aligned} X_{N+1}^T X_{N+1} &= \begin{bmatrix} X_N^T & X_{(N+1)}^T \end{bmatrix} \begin{bmatrix} X_N \\ X(N+1) \end{bmatrix} \\ &= X_N^T X_N + X^T(N+1)X(N+1) \end{aligned} \quad (2.46)$$

$$X_{N+1}^T Y_{N+1} = X_N^T Y_N + X^T(N+1)y(N+1) \quad (2.47)$$

Donc :

$$\hat{\underline{\theta}}_{N+1} = [X_N^T X_N + X^T(N+1)X(N+1)]^{-1} [X_N^T Y_N + X^T(N+1)y(N+1)] \quad (2.48)$$

En utilisant la formule de calcul matriciel suivante :

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1} \quad (2.48)$$

$$\text{En posant : } A = X_N^T X_N, \quad B = X^T(N+1), \quad C = I, \quad D = X(N+1) \quad (2.49)$$

On a en omettant l'indice N ou $N+1$, pour commodité de calcul :

$$\begin{aligned} [X_N^T X_N + X^T(N+1)X(N+1)]^{-1} = \\ (X^T X)^{-1} - (X^T X)^{-1} X^T [1 + X(X^T X)^{-1} X^T]^{-1} X(X^T X)^{-1} \end{aligned} \quad (2.50)$$

D'où :

$$\hat{\underline{\theta}}_{N+1} = \hat{\underline{\theta}}_N + \frac{(X^T X)^{-1} X^T(N+1)}{1 + X(N+1)(X_N^T X_N)^{-1} X^T(N+1)} [y(N+1) - x(N+1)\hat{\underline{\theta}}_N] \quad (2.51)$$

Soit $P_N = (X_N^T X_N)^{-1}$ que l'on appelle matrice de gain. En remarquant que cette matrice peut être elle aussi calculée récursivement, on a :

$$P_{N+1} = P_N - \underline{K}_{N+1} X(N+1) P_N \quad (2.52)$$

Où \underline{K}_{N+1} est un gain dont l'expression est :

$$\underline{K}_{N+1} = P_N X^T(N+1) [1 + X(N+1) P_N X^T(N+1)]^{-1} \quad (2.53)$$

L'algorithme final est alors :

$$\begin{aligned} \hat{\underline{\theta}}_{N+1} &= \hat{\underline{\theta}}_N + \underline{K}_{N+1} (y(N+1) - X(N+1)\hat{\underline{\theta}}_N) \\ \underline{K}_{N+1} &= P_N X^T(N+1) [1 + X(N+1) P_N X^T(N+1)]^{-1} \\ P_{N+1} &= P_N - \underline{K}_{N+1} X(N+1) P_N \end{aligned}$$

(2.54)

Les matrices $\hat{\theta}$, \underline{K} et P peuvent être mises à jour à chaque instant d'échantillonnage grâce à l'acquisition de $y(N+1)$ et $u(N+1)$. Il faut noter que le calcul ne nécessite aucune inversion de matrice puisque la quantité $X(N+1)P_N X^T(N+1)$ est un scalaire. Pour initialiser la récurrence (2.56), $n+p+1$ échantillons seraient nécessaires, en principe, pour construire la matrice P_0 de départ. En fait, on montre que la valeur initiale $\hat{\theta}_0$ des paramètres a peu d'importance et qu'une initialisation de P_0 à $1/\varepsilon$ ($\varepsilon \rightarrow 0$), c'est-à-dire une matrice dont tous les éléments sont nuls sauf la diagonale qui est infinie, permet la convergence de l'algorithme.

2.2.2.1.3 Interprétation physique de l'algorithme

Le nouveau vecteur $\hat{\theta}_{N+1}$ est égal à l'ancien $\hat{\theta}_N$ à un terme correctif près. Ce terme est le produit par un gain \underline{K}_{N+1} de l'expression :

$$y(N+1) - X(N+1)\hat{\theta}_N = y(N+1) - \hat{y}(N+1/N) \quad (2.55)$$

Cette expression représente l'erreur de prédiction de l'estimateur $\hat{\theta}_N$, ce à quoi on s'attendait, puisque l'estimateur des moindres carrés minimise l'erreur de prédiction. L'algorithme peut être représenté par le système à contre-réaction suivant (**Figure 2.6**) :

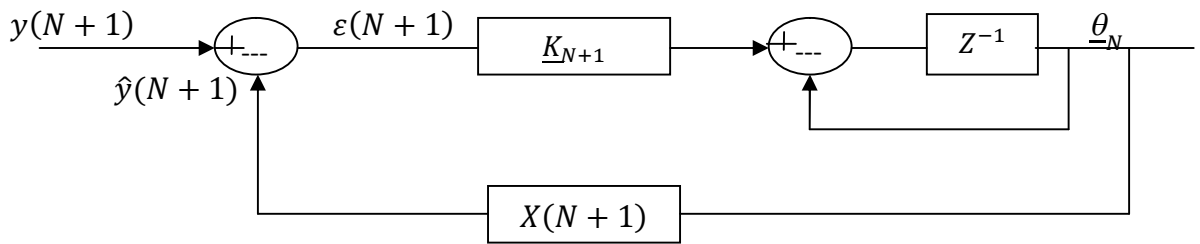


Figure 2.6 Schéma d'interprétation physique de l'algorithme des M.C

La sortie de ce système, $\hat{\theta}_N$, est corrigée avec le gain \underline{K}_{N+1} tant que l'erreur ε_{N+1} entre sortie mesurée et sortie prédite n'est pas nulle.

2.3 Etapes et mise en œuvre de l'identification [6]

L'identification des paramètres d'un modèle de comportement est en général un problème difficile, car la mise en œuvre des méthodes d'identification citées précédemment est délicate. L'objectif est de détailler la méthodologie à suivre pour construire un modèle de type « boîte noire ». Pour ce faire, on examinera successivement chaque étape d'une campagne d'identification.

1. Mise au point du protocole expérimental
2. Réalisation de la campagne de mesures
3. Prétraitement des données
4. Calcul des paramètres du modèle
5. Validation des résultats.

L'étape faisant intervenir la méthode d'identification, ne représente qu'une petite partie du travail. De plus, les outils informatiques existants permettent de mettre facilement en œuvre ces techniques. Par conséquent, la qualité du modèle obtenu dépendra essentiellement du soin apporté aux autres étapes. Il faut donc voir l'identification comme une procédure itérative telle qu'elle est illustrée dans la (**Figure 2.7**).

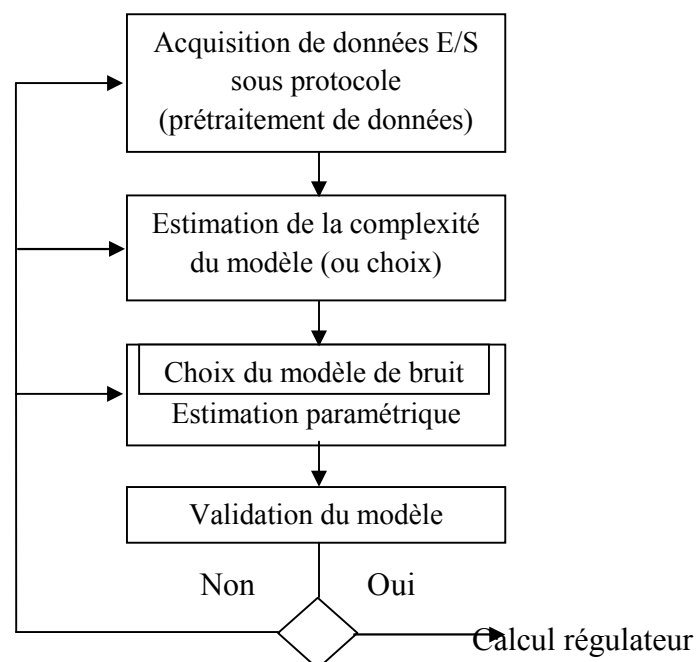


Figure 2.7 Méthodologie de l'identification.

La disponibilité des calculateurs numériques a eu un rôle fondamental dans le développement des techniques d'identification. Le principe de l'estimation des modèles échantillonnés est illustré dans la (**Figure 2.8**).

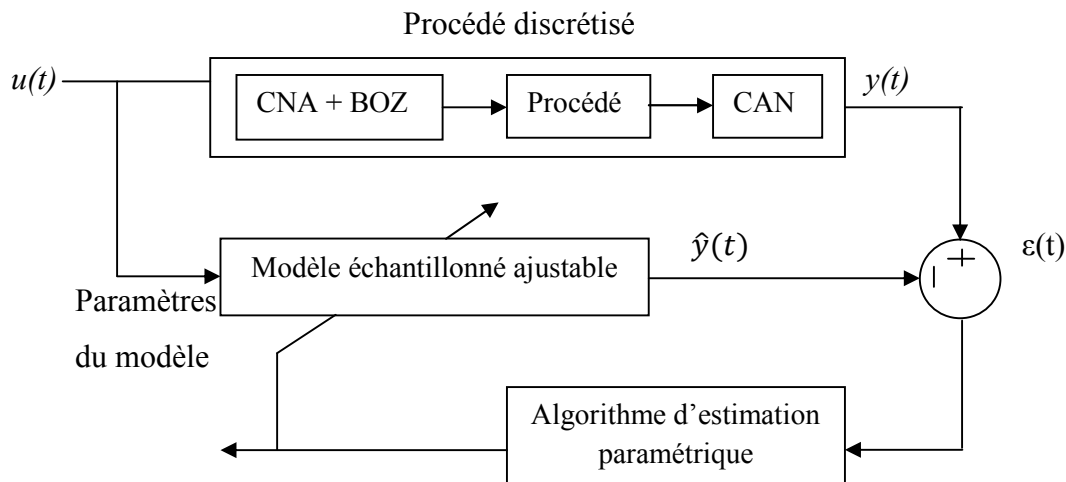


Figure 2.8 Principe de l'identification des modèles échantillonnés

2.3.1 Mise au point du protocole d'identification

2.3.1.1 Définition des objectifs

La première étape consiste à définir les objectifs du modèle, en répondant par exemple aux questions suivantes :

- dans quel domaine de fonctionnement le modèle va-t-il être utilisé ?
- quelles sont les actions tolérées sur la variable manipulée ?
- existe-t-il des perturbations mesurables ?
- quel est le niveau de performance souhaité ?

2.3.1.2 Conditions opératoires

Les objectifs étant fixés, il s'agit maintenant de définir les conditions opératoires, c'est-à-dire :

- Choisir le ou les régimes de fonctionnement auxquels seront réalisées les expériences.
- Choisir la ou les *variables manipulées* (actions).
- Identifier les *perturbations mesurables*.
- Choisir les variables dont l'évolution sera enregistrée, soit pour identifier un modèle dynamique entre celles-ci et les variables manipulées, soit pour vérifier que les conditions de fonctionnement nominales sont bien maintenues.

Il est aussi utile de spécifier comment la séquence d'action sera appliquée. En effet, il n'est pas toujours possible de la générer avec un système informatique, dans ce cas la séquence est appliquée manuellement par l'opérateur.

2.3.1.3 Détermination d'une séquence d'action compatible avec le procédé

Pour montrer l'importance de la séquence d'entrée sur la qualité des résultats de l'identification, deux caractéristiques sont à déterminer :

- sa forme
- et ses dimensions (durée, amplitude).

2.3.1.3.1 Forme du signal d'entrée

Une séquence d'entrée souvent utilisée est l'échelon, mais ce type d'entrée n'excite pas suffisamment le système. Par ailleurs, l'échelon présente le défaut de ne pas être de valeur moyenne nulle. L'idéal serait d'utiliser une séquence SBPA (séquence binaire pseudo aléatoire). Malheureusement, ce type de séquence n'est pas toujours facile à appliquer sur un procédé et peut le perturber de façon inquiétante pour les opérateurs. Un bon compromis consiste à choisir un signal entre l'échelon et la SBPA, à savoir une succession de créneaux (**Figure 2.9**).

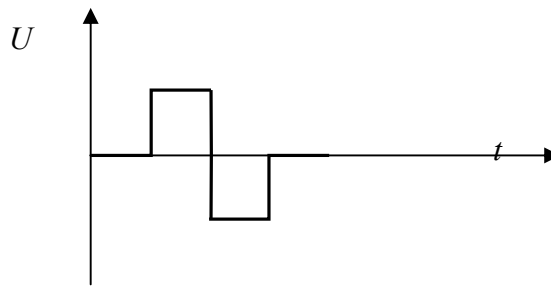


Figure 2.9 Séquence d'entrée en créneau

2.3.1.3.2 Dimensionnement de la séquence

La forme du signal étant choisie, il reste à en fixer les paramètres numériques c'est-à-dire l'amplitude, la durée totale et la durée du créneau.

La *durée* de la séquence sera choisie de telle façon à ce que le créneau le plus long soit au moins égal au temps de réponse, qui, dans le cas d'un premier ordre, est à peu près égale à $3 \times \tau$.

Pour fixer l'*amplitude*, les paramètres à prendre en compte sont :

- le niveau de bruit de mesure, il faut que la sortie observée se distingue clairement,
- les perturbations tolérées sur les conditions opératoires,
- les contraintes sur les actionneurs, en valeur absolue, mais surtout en variation,
- L'aspect non-linéaire du procédé : il risque d'apparaître si l'amplitude de la variable manipulée est trop importante.
- les corrections de consigne, affichées en boucle ouverte (c'est-à-dire en mode manuel) par l'opérateur.

2.3.1.4 Echantillonnage

La conversion du signal analogique en un signal numérique fait intervenir deux opérations : l'échantillonnage du signal, et sa discrétisation. En effet, que le régulateur envisagé soit de type discret ou continu, le modèle de comportement va être calculé sous

forme discrète. Le choix de la période d'échantillonnage d'acquisition doit être donc fait de façon à ce que le signal échantillonné donne une représentation correcte du signal continu. Pour ce faire, la période d'échantillonnage T_e doit respecter la condition de Shannon :

$$T_e < \frac{1}{2.f_{max}} \quad (2.56)$$

où f_{max} est la fréquence maximale contenue dans le signal.

Pour satisfaire ce critère, pour un signal donné. Il est possible :

- soit de filtrer analogiquement de façon à réduire f_{max}
- soit d'acquérir les signaux avec une fréquence multiple de la fréquence d'échantillonnage $f_a = k.f_e$ ($T_a = T_e/k$), puis de réaliser un filtrage numérique avant de sous-échantillonner les mesures.

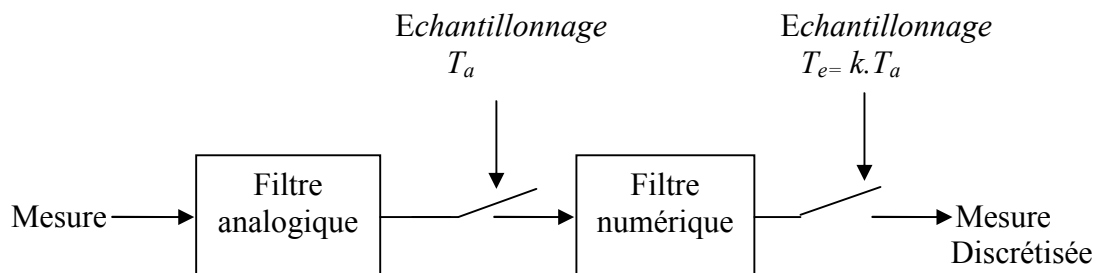


Figure 2.10 Filtrage analogique et numérique des signaux

En général, avec les systèmes de commande actuels, la seconde solution est plus facile à mettre en œuvre, car leur temps de cycle est nettement inférieur aux constantes de temps des divers signaux rencontrés sur le procédé. Par contre, une telle approche génère un nombre important de points.

2.3.2 Réalisation de l'expérience

Cette étape consiste à réaliser l'expérience proprement dite, c'est-à-dire à mettre en œuvre le protocole défini précédemment. Celle-ci dépend de la nature du procédé. Elle est en général réalisée en collaboration avec l'exploitant avec lequel un bon dialogue est nécessaire de façon à ce qu'il soit davantage sensible à l'aspect positif de la campagne de mesures, qui va permettre la mise en place d'un système de commande performant, qu'à son aspect négatif, car elle perturbe l'exploitation normale. A la suite de cette étape, on dispose des données brutes à partir desquelles nous allons essayer de caractériser la dynamique du système.

2.3.3 Prétraitement des données

2.3.3.1 Elimination des points aberrants

La première étape du prétraitement consiste à visualiser graphiquement les fichiers de mesures obtenus lors de l'expérience. A partir de cet examen visuel, on va sélectionner les parties de l'enregistrement qui contiennent les données intéressantes (on élimine les phases de démarrage et d'arrêt de l'expérience). Si cette zone contient des points manifestement aberrants, on les éliminera manuellement.

2.3.3.2 Sous-Echantillonnage

L'objectif du sous-échantillonnage est d'augmenter le cas échéant la période d'échantillonnage des points obtenus après acquisition de façon à ce que sa nouvelle valeur ne soit pas trop petite par rapport à la constante de temps principale du procédé (au minimum 1/100 de celle-ci). Si cette étape n'a pas lieu d'être, on peut passer directement à la suivante.

2.3.3.3 Cas d'un système intégrateur

Les systèmes intégrateurs sont difficiles à identifier directement. Il est préférable de se ramener à un système sans intégrateur. Ceci peut se faire, soit en cherchant le modèle qui relie

l'entrée à la dérivée de la sortie, soit en cherchant le modèle qui relie l'intégrale de l'entrée à la sortie.

2.3.3.4 Elimination de la composante continue et des dérives

Avant d'identifier le modèle dynamique, il est nécessaire de calculer les variations des grandeurs considérées autour du point de fonctionnement à partir des données recueillies expérimentalement. En effet, le point de fonctionnement en régime permanent de la variable manipulée et de la variable mesurée sont parfaitement connus, auquel cas il suffit de le retirer aux grandeurs absolues de l'entrée et de la sortie. Ou bien le point de fonctionnement n'est pas connu exactement, et il est constant, et que les signaux utilisés sont centrés il est possible de l'estimer en prenant la moyenne de l'enregistrement de la grandeur correspondante.

Il arrive aussi très souvent que le point de fonctionnement au niveau de la sortie, soit sujet à de lentes dérives, et leur élimination consiste à réaliser un filtre passe-haut des signaux, qui ne modifie pas le modèle reliant les signaux filtrés. Par contre, il va modifier l'identifiabilité des paramètres.

2.3.3.5 Filtrage du bruit de mesure

Les variations des entrées et sorties autour du point de fonctionnement peuvent présenter une allure bruitée qui sera filtrée à l'aide d'un filtre passe-bas parallèle.

2.3.3.6 Mise à l'échelle

Avant de procéder au calcul des coefficients du modèle proprement dit, il faut s'assurer que l'échelle de l'entrée et de la sortie sont du même ordre de grandeur, ceci afin d'éviter des problèmes numériques.

2.3.4 Calcul des paramètres du modèle

2.3.4.1 Choix de la méthode

Le calcul des paramètres du modèle se fait en utilisant une des méthodes citées précédemment. Dans la plupart des cas, le spectre des bruits de mesure est situé hors du spectre des signaux utiles. Par conséquent, un filtrage passe-bas permet d'éliminer le bruit tout en conservant l'information utile sur la dynamique du procédé. On pourra alors se contenter d'une méthode d'identification s'appuyant sur un modèle simplifié du bruit. A savoir la méthode des moindres carrés et la méthode du modèle.

2.3.4.2 Problème du retard

Avec les méthodes d'identification numériques, le retard doit être déterminé à priori. Pour l'obtenir, la solution la plus simple consiste à observer les données, et à le déterminer visuellement.

2.3.4.3 Choix de la structure

La structure du modèle est déterminée par le choix des degrés des polynômes $A(q^{-1})$ et $B(q^{-1})$ composant le modèle discret du système :

$$y(k) = -a_1 y(k-1) - a_2 y(k-2) - \dots - a_{na} y(k-na) + b_1 u(k-1-r) + \dots + b_{nb} u(k-nb-r) \quad (2.58)$$

Dans la plupart des cas, un modèle d'ordre un ou deux, voire trois, est suffisant pour représenter correctement un procédé. Le choix de l'ordre du système pourra se faire par essai/erreur en partant d'un ordre un, qu'on pourra augmenter progressivement en évaluant l'impact sur la qualité de la modélisation. Toutefois, certains indices visuels peuvent aider au choix de l'ordre du système comme le caractère oscillant de la réponse indicielle ou l'examen de la tangente au début de la réponse.

En règle générale, l'augmentation de l'ordre n'améliore pas la qualité de l'identification. Si un modèle d'ordre faible ne conduit pas à des résultats satisfaisants, il faut plutôt se tourner vers un examen des non linéarités du système, ou essayer de rechercher l'influence que peuvent avoir certaines variables autres que l'entrée (perturbations mesurables).

2.3.4.4 Conditions initiales

Dans la méthode des moindres carrés, que ce soit pour l'identification d'une fonction de transfert discrète ou pour celle des coefficients de la réponse impulsionnelle, les valeurs prédites comparées aux valeurs observées sont calculées à partir des valeurs passées sur un horizon $h_0 = \max \{n_a, n_b\}$ dans le cas du calcul de la fonction de transfert ou sur un horizon $h_0 = N_p$ pour le calcul de la réponse impulsionnelle. Par conséquent, le critère utilisé pour l'identification ne prendra en considération que les valeurs observées à partir de h_0 .

$$C(\theta) = \frac{1}{2} \sum_{k=h_0}^N e^2(k) \quad (2.60)$$

La méthode du modèle, elle est basée sur une simulation réalisée à partir du modèle en cours d'identification. Ce modèle doit bien sûr être initialisé. Plusieurs approches sont possibles :

- Initialiser le modèle sur les valeurs mesurées.
- Laisser, comme précédemment, un horizon initial avant de considérer l'erreur dans le critère. La durée de cet horizon est fixée en fonction du temps de réponse du procédé T_{95} (temps de réponse à 95%), sachant qu'au bout de ce temps, les conditions initiales d'un modèle linéaire stable n'ont plus beaucoup d'effet sur la sortie ; elles sont en quelque sorte oubliées.
- Ajouter les valeurs initiales dans le critère et les considérer comme des paramètres supplémentaires.

Cette dernière solution est lourde à mettre en œuvre, et une des deux précédentes sera préférée, la première étant satisfaisante si le bruit de mesure n'est pas trop important.

2.3.5 Validation des résultats

L'étape de validation des résultats consiste à s'assurer que le modèle calculé est adéquat pour représenter le système considéré.

On commencera par vérifier que la minimisation du critère d'identification a bien abouti, et que la sortie prédite par le modèle et la sortie réelle sont proches. Si cette condition n'est pas remplie, cela signifie que la structure du modèle n'est pas adaptée. Sinon, il convient de vérifier que les paramètres ont été identifiés avec précision. Ce point se juge en examinant la covariance des paramètres du modèle.

Par ailleurs, comme on s'intéresse en général à la capacité du modèle à simuler le comportement du système réel, on comparera la sortie simulée au moyen du modèle avec la sortie réelle. L'une des sources du problème est le mauvais choix de la période d'identification qui intervient sur l'identifiabilité des paramètres, pouvant expliquer les différences de la sortie prédite vis-à-vis de la sortie simulée est le niveau du bruit.

Jusqu'ici, on a cherché à vérifier que le modèle permettrait de représenter le jeu de données initiales. Ce n'est pas suffisant. Il faut maintenant comparer les simulations aux sorties réelles sur d'autres jeux de données. La séquence d'entrée pourra être choisie de façon à tester le modèle aux limites en prenant par exemple une amplitude de l'action plus forte ou une fréquence de commutation plus élevée. On essaiera de limiter le prétraitement des données pour avoir une comparaison la plus réaliste possible. Celle-ci n'est pas toujours facile, par exemple, dans le cas où le point de fonctionnement dérive lentement. Si ce test est réussi, on peut considérer que le modèle est une bonne approximation du procédé.

Conclusion

Dans ce chapitre, on a vu l'aspect général de l'identification des procédés, ainsi que les principales méthodes utilisées en les classant en deux catégories : les méthodes graphiques et les méthodes numériques. A la fin du chapitre, on a évoqué les étapes et les précautions à prendre pour la mise en œuvre d'une bonne identification après l'acquisition de données. Le chapitre suivant sera consacré à l'étape d'acquisition des données qui joue un rôle du premier plan dans le processus d'identification.

CHAPITRE 3:

Acquisition des données avec Lab View

Introduction

Ce chapitre a pour but de donner un aperçu général sur l'acquisition de données du point de vue purement théorique. En effet l'acquisition est une étape très importante du processus d'identification, c'est pour cela qu'il faut prendre quelques précautions avant de l'entamer et s'assurer que le matériel utilisé et sa configuration sont adéquats.

3.1 C'est quoi l'acquisition ? [7]

Pour agir efficacement sur un processus physique, chimique ou biologique, naturel ou industriel, il est important, au préalable, de bien le connaître : la chaîne d'acquisition fournit au décideur (homme ou machine) les informations permettant d'orienter son action et de valider ses décisions.

D'un monde réel complexe où s'enchevêtrent de multiples phénomènes, la chaîne d'acquisition permet d'extraire les informations simples et nécessaires à une représentation valable et utile. Ces informations caractérisent les grandeurs physiques telles que : température, pression, débit, force, etc. Connaître une grandeur, c'est, pour le physicien ou l'ingénieur, lui affecter une valeur numérique ou une succession de valeurs numériques significatives qui résultent d'opérations de mesure : la chaîne d'acquisition est donc une chaîne de mesure. La grandeur objet de mesure ou mesurande est caractérisée par son intensité et par son évolution temporelle ; ce sont ces informations que pour chaque mesurande, la chaîne d'acquisition doit pouvoir délivrer sous la forme de signaux électriques représentatifs et exploitables.

Les difficultés que présente l'acquisition de données ont deux causes principales :

- L'imbrication dans la réalité de multiples grandeurs qui rend souvent difficile l'isolement et la saisie du seul mesurande recherché ;
- Les imperfections des dispositifs constitutifs de la chaîne qui entraînent une détérioration des signaux et donc de l'information ;

C'est à ces deux niveaux que doit se porter l'attention du concepteur pour limiter l'imprécision des mesures et éviter les risques d'erreurs. La **Figure 3.1** illustre parfaitement la place et l'importance de la chaîne d'acquisition dans un ensemble de mesure-contrôle de procédé.

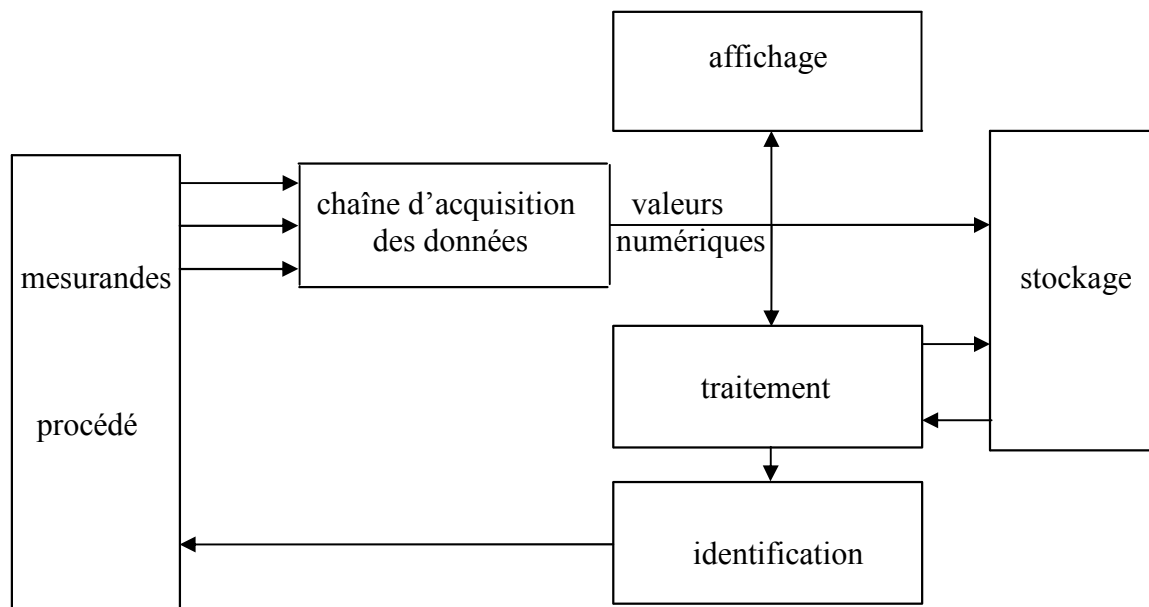


Figure 3.1 Place de la chaîne d'acquisition de données dans un ensemble de mesure-contrôle de procédé.

3.2 Traitement des mesures

Les dispositifs dont l'association au capteur constitue la chaîne d'acquisition (amplificateur, filtre, échantillonneur-bloqueur, convertisseur analogique-numérique) ont pour fonction de traiter le signal de façon à le rendre approprié à l'exploitation prévue. Cependant, les caractéristiques de ces dispositifs ont des limitations donc il faut tenir compte pour éviter une détérioration du signal et donc une perte d'information : dérives thermiques et temporelles, bruit de fond, sensibilité aux perturbations électromagnétiques, non linéarité, bande passante limitée ou temps de réponse fini. Concevoir une chaîne d'acquisition, c'est choisir les dispositifs et leur montage de façon que leurs limitations soient compatibles avec la sauvegarde de l'information liée au signal.

La généralisation du rôle des ordinateurs dans le contrôle-commande des processus industriels et en particulier, leur association aux chaînes d'acquisition n'a pas supprimé les problèmes précédents : elle donne, cependant, des moyens nouveaux, automatisés et rapides pour en tenir compte et les corriger, déchargeant ainsi l'opérateur des tâches fastidieuses de réglage, de compensation, d'annulation, etc. En outre, l'ordinateur est capable de remplir, en aval de la chaîne, les fonctions complémentaires à l'acquisition que sont la mémorisation, l'affichage des données, leur analyse et leur traitement en temps réel.

3.3 Acquisition de données avec LabView

3.3.1 Présentation du logiciel LabView [8]

LabView (Laboratory Virtual Instrument Engineering Workbench) est un langage de programmation dédié au contrôle d'instruments et l'analyse de données. Contrairement à la nature séquentielle des langages textuels, LabView est basée sur un environnement de programmation graphique utilisant la notion flot de données pour ordonnancer les opérations.

Le nom du logiciel LabView signifie «Laboratory Virtual Instrument Engineering Workbench ». Ce langage de programmation graphique est un environnement de programmation à caractère universel particulièrement bien adapté à la mesure, au test, à l'instrumentation et à l'automatisation. Il a été développé par la société National Instrument à partir de 1983 et c'est l'un des premiers langages de programmation graphique destiné au développement d'applications d'instrumentation. Couplé à des cartes d'entrées/sorties, il permet de gérer des flux d'informations numériques ou analogiques et de créer ou de simuler des instruments de mesures (Oscilloscope, Multimètre,etc...). Ici on utilise la version 6.0 de LabView.

LabView est un outil d'**acquisition**, d'**analyse** et de présentation de **données** comme il est illustré dans la **Figure 3.2** suivante :

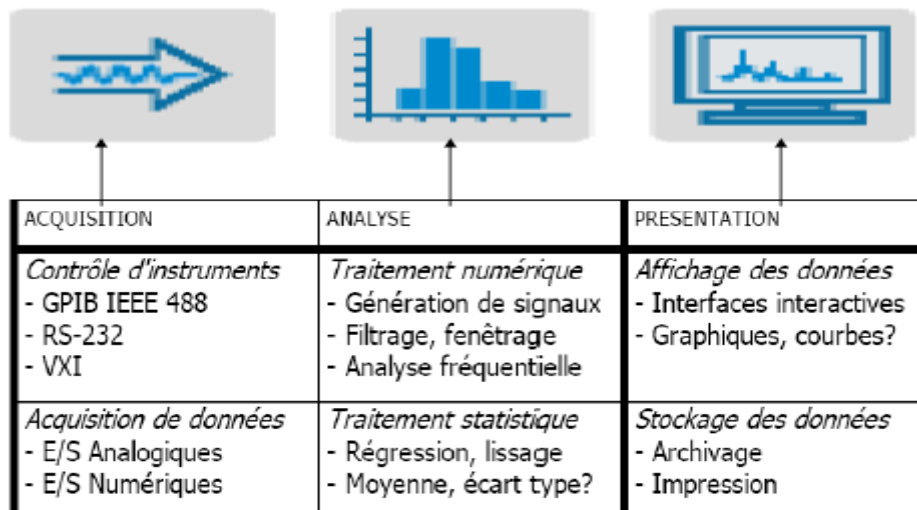


Figure 3.2 Représentation des différents outils de LabView

Les programmes LabView comportent des Instruments Virtuels ou Virtual Instruments (VI). On parle d'instruments virtuels car leur apparence et leur fonctionnement sont semblables à ceux d'instruments réels, tels que les oscilloscopes et les multimètres.

3.3.1.1 Les éléments principaux d'un VI

Une **face-avant** : interface utilisateur de la VI, permettant de réceptionner les données acquises et d'afficher celles fournies en sortie par le programme.

Voici un exemple de face-avant :

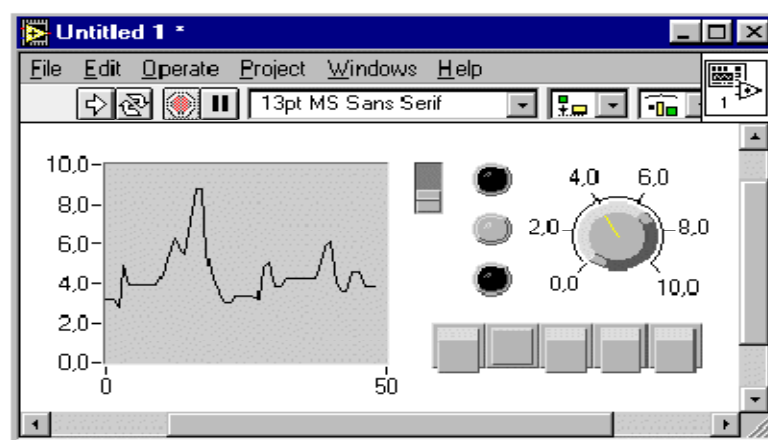


Figure 3.3 Face avant d'un VI.

Un **diagramme** : Fenêtre de programmation et d'affichage du code source. Interaction entre face avant et diagramme.

Voici un exemple de diagramme :

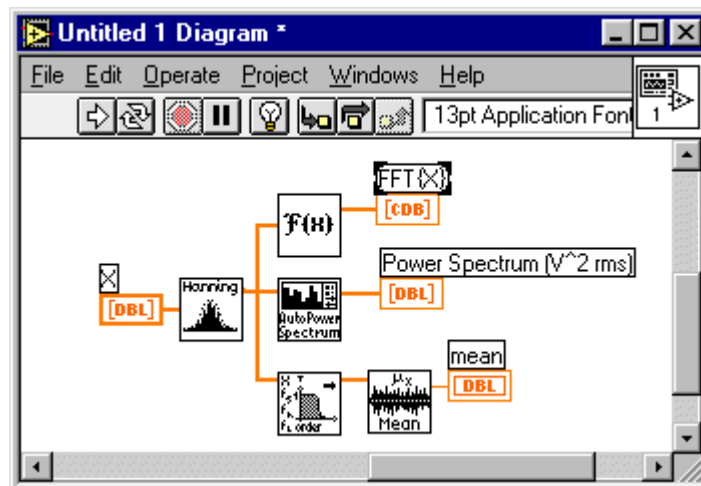


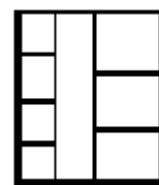
Figure 3.4 Face diagramme d'un VI.

Une **icône et un connecteur** : une icône est la symbolisation de l'instrument virtuel qui permettra de faire appel à un instrument virtuel déjà créé à l'intérieur d'un autre instrument virtuel. Un connecteur est un ensemble de terminaux correspondant aux commandes et aux indicateurs du VI qui sont accessibles.

Voici un exemple d'icône :



Voici un exemple de connecteur :



On dispose pour écrire un programme sur LabView, des Palettes qui offrent la possibilité de modifier la face avant et le digramme, on trouve trois palettes :

• Palette d'outils

Elle est disponible sur la face-avant et sur le diagramme, elle contient les outils nécessaires pour faire fonctionner et modifier la face avant et les objets du diagramme.



Figure 3.5 Palette d'outils.

. Palette de commandes

Elle est disponible uniquement sur la face-avant, elle contient les commandes et les indicateurs de la face-avant nécessaire pour créer l'interface utilisateur.



Figure 3.6 Palette de commandes.

. Palette de fonctions

Elle est disponible uniquement sur le diagramme. Elle contient les objets nécessaires pour la programmation graphique comme les opérations d'arithmétique, d'E/S d'instrument, d'E/S de fichier et d'acquisition de données.

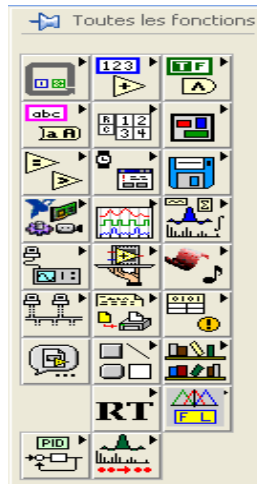


Figure 3.7 Palette de fonctions.

3.3.1.2 Structure de données dans LabView

LabView utilise un langage fortement typé et toutes données ou structure de données ne peuvent être manipulées qu'avec des fonctions admettant ce type, en fait dans LabView on trouve les types de base scalaire, les types entiers (signés ou non, codes sur 8, 16 ou 32 bits), le type réel (code sur 16, 32 ou 64 bits), le type booléen et le type chaîne de caractères (**Figure 3.6**). Il est important de noter que les éléments représentant ces données, ainsi que les liaisons issues de ces éléments, sont de forme et de couleur différente.

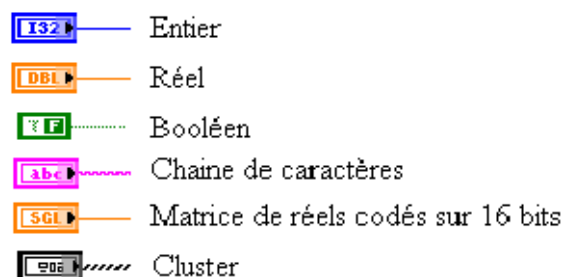


Figure 3.8 Différents types de structures de données dans LabView.

Le langage permet aussi de créer des structures de données plus élaborées :

- **le type tableau (structure de données homogènes)**

Comme tous les langages de programmations la présence de notion de tableau est obligatoire, puisqu'elle définit un outil de base pour le stockage et les traitements des informations. La bibliothèque tableau dans LabView est riche en fonctions prédéfinies, tel que, l'initialisation, l'indexation, l'inversement d'un tableau, la concaténation de deux tableaux, et plusieurs autres fonctions, bien sûr, la notion des tableaux peut être élargie pour contenir les matrices. Et comme toutes les autres bibliothèques, on peut l'enrichir avec des fonctions ou des procédures créées à partir des langages de bas niveau comme le langage C.

- **Le type « cluster » (structure de données hétérogènes)**

Ce second constructeur de type est l'équivalent du « struct » en C. Les différents composants ou champs de ce groupe de données (cluster) peuvent être assemblés ou récupérés par des fonctions spécifiques.

3.3.1.3 Traitement numérique

- Les fonctions prédéfinies : LabView possède les instructions de base d'un langage de programmation permettant de traiter les différents types de données. Ainsi, on a des fonctions liées aux variables numériques (entiers, réels et complexes), aux variables booléennes, aux chaînes de caractères, et aux tableaux. On trouve aussi les opérateurs de tests et de comparaisons liés à ces différents types de données. À partir des structures de contrôle, des fonctions et des opérateurs, de base, il est alors possible de traduire un algorithme quelconque et d'enrichir la bibliothèque des fonctions en utilisant le mécanisme d'encapsulation. Un diagramme complet est alors réduit à un nœud qui peut être ensuite réutilisé.

- Boîtes à outils mathématique : LabView contient aussi des boîtes de calcul mathématiques qui servent à introduire des commandes complexes tel que (linspace, ode45, sin, cos, etc..).

3.3.1.4 Quel type de programmation utilise-t-on pour LabView ?

Le langage propre de LabView est une programmation graphique en langage « G ». Le langage G se base sur le principe du flot de données, auquel ont été rajoutées des structures de programmation afin d'obtenir un langage de programmation complet.

Remarque 3.1

Un langage textuel suit des règles d'exécution séquentielle déterminée par la position des instructions dans le fichier source, tandis qu'un programme graphique s'exécute selon le principe de flux de donnée.

3.3.1.5 Structure de programme

LabView dispose d'un programme qui est enrichi de quatre types de structures : la séquence, deux structures d'itération (la boucle «Pour » avec un nombre d'itérations fixe et la boucle « Tant Que » avec un nombre d'itérations soumis à une condition) et la structure de choix.

3.3.2 Démarche d'acquisition

Pour une bonne acquisition, il faut au préalable réunir le matériel nécessaire et la configuration requise pour ce dernier. Tout d'abord le contrôle d'instrument s'effectue par une communication (liaison) GPIB entre le PC (micro-ordinateur) et le procédé à l'aide d'une carte d'acquisition qui sert à transmettre les signaux du PC vers le procédé (capteur) pour la commande, et vice-versa pour l'acquisition (mesure). La carte est reliée à un boîtier amplificateur pour amplifier les signaux de commande et un conditionneur pour une conversion soit analogique-numérique (CAN) ou bien numérique-analogique (CNA) comme le montre la **Figure 3.8**.

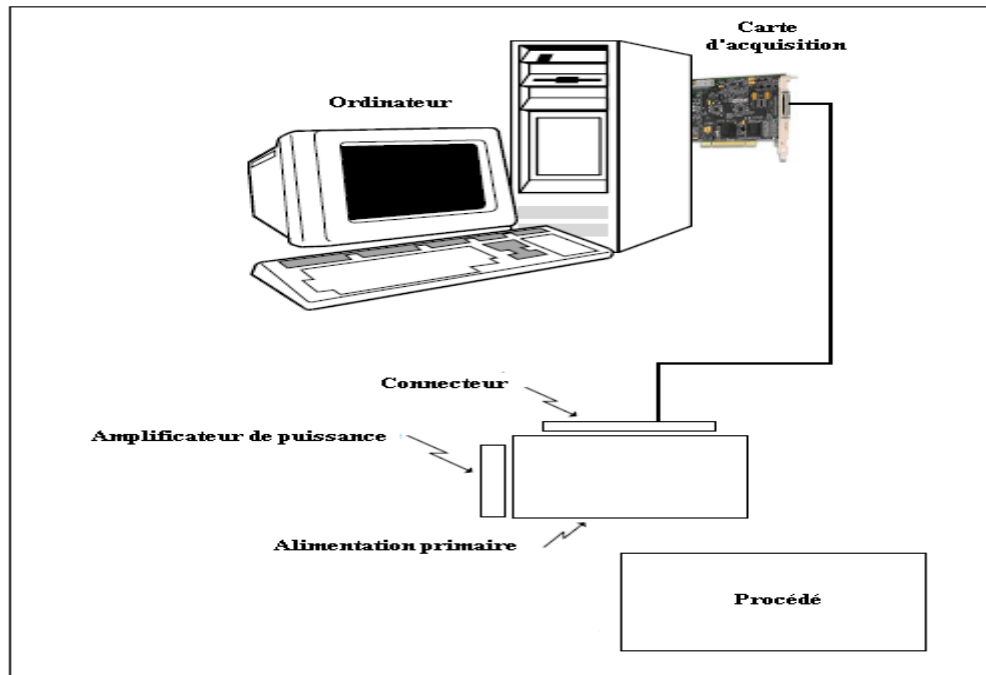


Figure 3.8 Schéma de liaison du matériel.

Conclusion :

Dans ce chapitre, on a présenté des notions générales relatives à l'acquisition des données, et on a introduit le langage graphique LabView utilisé dans des applications de commande et d'acquisition.

Dans le chapitre suivant, on s'intéressera à l'identification d'un système basée sur une acquisition des données par le LabView.

CHAPITRE 4:

Identification d'une station de pression

Introduction

Dans le présent chapitre, on va procéder à la mise en œuvre de l'identification d'un modèle d'un compresseur existant dans le laboratoire d'automatique industrielle et de régulation au niveau du département Automatique de l'Université Mouloud MAMMERI de Tizi-Ouzou. L'identification sera réalisée par une méthode graphique en utilisant les mesures acquises par le logiciel LabView comme a été décrit dans le chapitre 3.

4.1 Présentation du procédé

L'environnement de travail dans le laboratoire est illustré par la **Figure 4.1**. Il se compose du matériel suivant : un bloc d'alimentation, une station de pression et un PC équipé d'une carte d'acquisition.

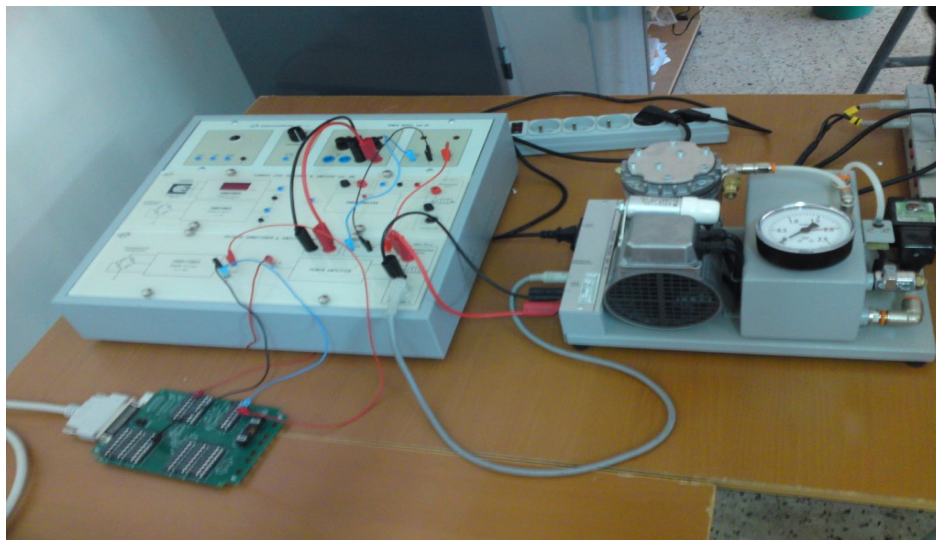


Figure 4.1 Environnement du procédé.

4.1.1 Bloc d'alimentation

Pour l'alimentation du compresseur, on utilisera les deux parties du bloc illustré dans la **Figure 4.2**, à savoir :

-un conditionneur qui sert à conditionner et linéariser le signal de commande et les mesures avec une échelle 0~10 volts qui correspond à 0~2 bars.

-un amplificateur de puissance pour amplifier le signal d'entrée à travers la vanne électrique.

Ces derniers sont alimentés par deux sources de tensions (± 15 volts) pour l'amplification et (± 30 volts) pour les composants des circuits internes tels que les transistors.

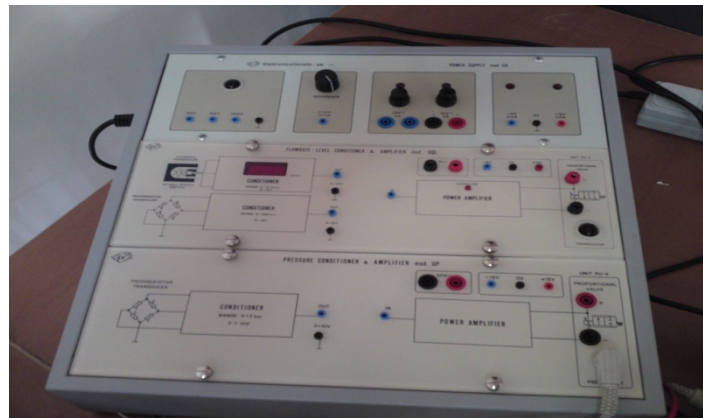


Figure 4.2 Bloc d'alimentation.

4.1.2 Station de pression

4.1.2 .1 Eléments de la station

La station sur laquelle on a travaillé est une station de régulation de pression illustrée par la **Figure 4.3**. Elle est constituée de :

- Un compresseur d'air,
- Une chambre de pression,
- Un moteur à courant continu,
- Une soupape de sécurité,
- Une vanne manuelle,

- Une vanne proportionnelle,
- Un capteur piézo-électrique,
- Un manomètre.



Figure 4.3 Station de pression.

4.1.2.2 Fonctionnement de la station

Le compresseur d'air aspire l'air ambiant à travers une entrée, cet air sera comprimé dans la chambre de pression à travers une sortie. L'ouverture et la fermeture de l'entrée et la sortie se fait à l'aide d'un piston qui prend la position soit en haut : il comprime l'air, soit en bas : il aspire l'air. Ceci est réalisé par un moteur à courant continu.

La capacité cylindrique sert pour le démarrage et la soupape de sécurité (ou de protection) fait dégager la pression vers l'extérieur pour protéger le procédé.

La vanne manuelle est utilisée généralement pour simuler une fuite, par contre la vanne proportionnelle fait juste passer l'air d'une façon proportionnelle.

Pour mesurer la variation de pression, un capteur piézo-électrique sensible à la pression, est placé à la sortie du procédé.

En fin, l'affichage est assuré par un manomètre qui traduit les variations de pressions en bar et la pression à ne pas dépasser est 2.5 bars.

4.1.3 PC et carte d'acquisition [9]

L'acquisition des données de la station doit être réalisée par le biais de la carte d'acquisition KPCI-3102 de Keithley Instruments. Ce matériel est compatible avec LabView mais nécessite un driver supplémentaire pour son fonctionnement.

KPCI-3102 est utilisée avec une liaison de type GPIB dont voici les caractéristiques techniques :

- La liaison GPIB est une liaison parallèle sur 8 bits.
- Interconnexion de 15 appareils maximum : chaque appareil possède une adresse comprise entre 0 et 30.
- Vitesse de transfert maximum : 1Mo/s.
- Longueur de câble de 4 m au maximum entre 2 appareils.
- Longueur totale de câble est de 20 m au maximum.

Cette carte supporte 16 canaux comme entrées analogiques simples ou bien 8 différentiables supportant des signaux compris entre 0,05~10V, et 2 canaux comme sorties analogiques différentiables (DAC0 et DAC1) variant de 0~10V.

Le Kit d'acquisition utilisé (**Figure 4.4**) comprend également un connecteur d'entrées/sorties permettant une connexion plus rapide et facile, via la carte d'acquisition, entre le procédé et l'ordinateur, fonctionnant sous le système d'exploitation Windows, afin d'effectuer les mesures.

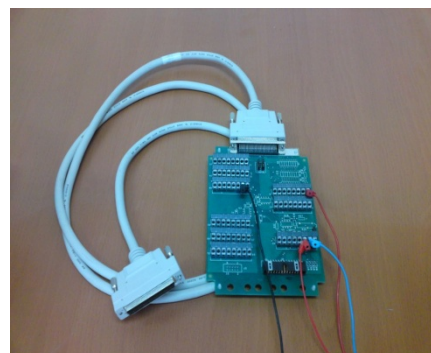


Figure 4.4 Kit d'acquisition des mesures.

En effet, l'entrée de l'amplificateur de puissance du bloc d'alimentation est branchée au premier convertisseur numérique analogique (N/A) de la carte d'acquisition, tandis que le conditionneur est branché au décodeur de cette même carte. Ces signaux d'entrées/sorties sont accessibles par l'entremise du logiciel LabView. Dans les manipulations qui suivront, ce logiciel sera utilisé pour une acquisition en temps réel, dans un langage graphique propre à LabView.

4.2 Acquisition de données par LabView

Le logiciel LabView (**Version 6.0**) est utilisé pour acquérir les mesures de la pression en temps réel via la carte d'acquisition. La programmation de LabView est réalisée grâce à un langage graphique de haut niveau. Pour faciliter la compréhension et éviter les piétinements inutiles, un programme de base permettant de lier les entrées/sorties du procédé à un environnement en temps réel dans LabView est fourni. Ce programme, illustré par la **Figure 4.5**, permet d'envoyer une tension à l'entrée de l'amplificateur, d'échantillonner la pression et d'enregistrer les valeurs pour un nombre d'échantillons et une période d'échantillonnage convenablement choisie. En plus de la fenêtre de programmation, LabView fait toujours intervenir une fenêtre de contrôle. Cette fenêtre, illustrée par la **Figure 4.6**, permet non seulement de moduler certaines variables du programme au cours de son exécution, mais également d'en visualiser d'autres à l'aide d'outils de visualisation tels que des graphiques. Dans la fenêtre de contrôle de la **Figure 4.6**, un contrôle de type valeur numérique permet de moduler la tension envoyée au procédé tandis que les deux graphiques compteurs (vumètre) et graphe illustrent l'évolution de la pression.

Dans la fenêtre de programmation (**Figure 4.5**), on distingue un rectangle qui englobe toute la partie du programme qui doit être exécutée à chaque période d'échantillonnage. Ce rectangle représente une boucle de programmation de type « **for** » qui est synchronisée sur une horloge de 500 millisecondes qui est représentée par l'icône métronome. Comme on peut le constater, il est très facile de modifier la période d'échantillonnage et le nombre d'échantillons.

En effet, à l'intérieur de la boucle « **for** » on trouve un bloc « **ai** » (analogue input) préconfiguré par LabVIEW et connecté à une chaîne de blocs. Le premier « array to cluster » qui convertit une entrée d'une dimension à une succession de valeurs qui seront désassemblées à l'aide d'un « désassembleur ». Pour avoir une bonne échelle pour les courbes, les valeurs obtenues doivent être multipliées par un coefficient de **0.20** déjà calculé. Ces dernières seront en même temps affichées à la fois sur un compteur numérique et un vumètre pour enfin les sauvegarder dans un tableau après chaque itération.

A l'extérieur de la boucle, les valeurs seront simultanément affichées sur un graphe déroulant dans la face avant et écrites dans un fichier. On dispose aussi d'un bloc « **ao** » (analogue output) qui permet de contrôler la valeur de la tension d'entrée, dans notre cas on utilisera un signal d'excitation de type échelon avec une densité spectrale homogène qui balaie l'ensemble de la bande passante (toute la dynamique) du procédé.

En sachant que la station de pression admet une tension d'entrée maximum de 10 volts, la valeur de l'échelon envoyé est de 9 volts et suffit largement pour avoir un signal de sortie exploitable pour l'identification.

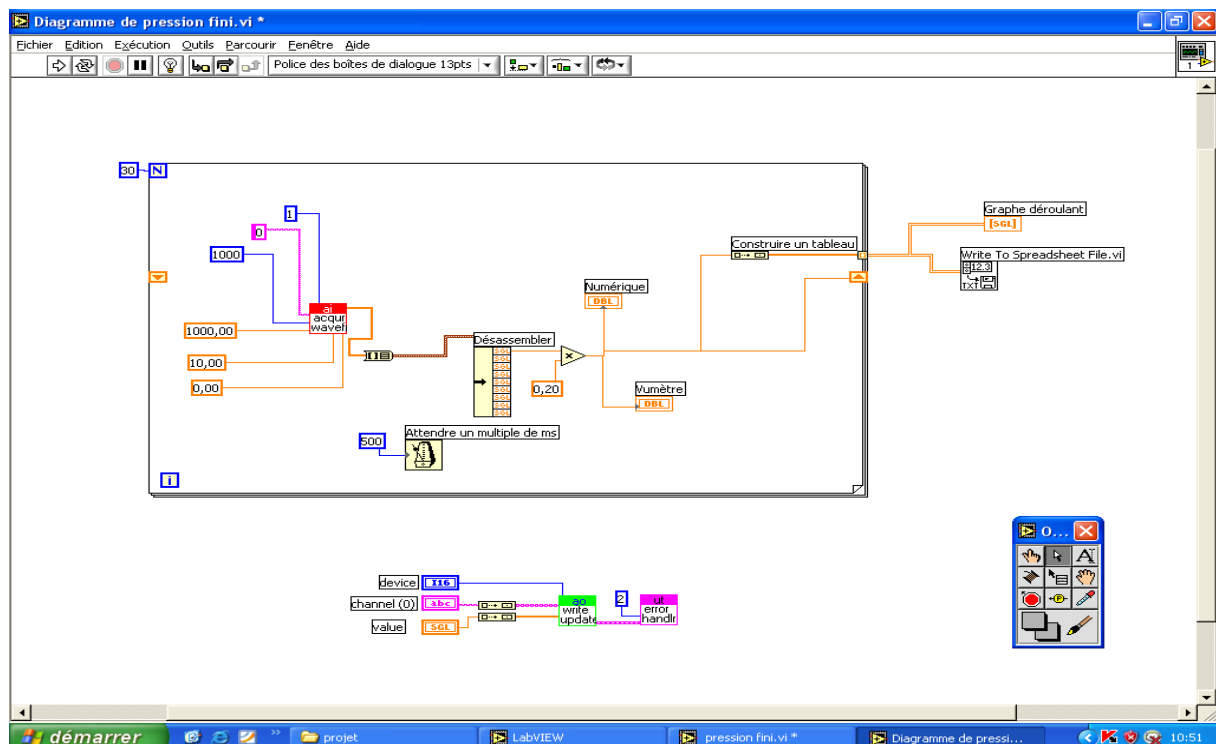


Figure 4.5 Face diagramme.

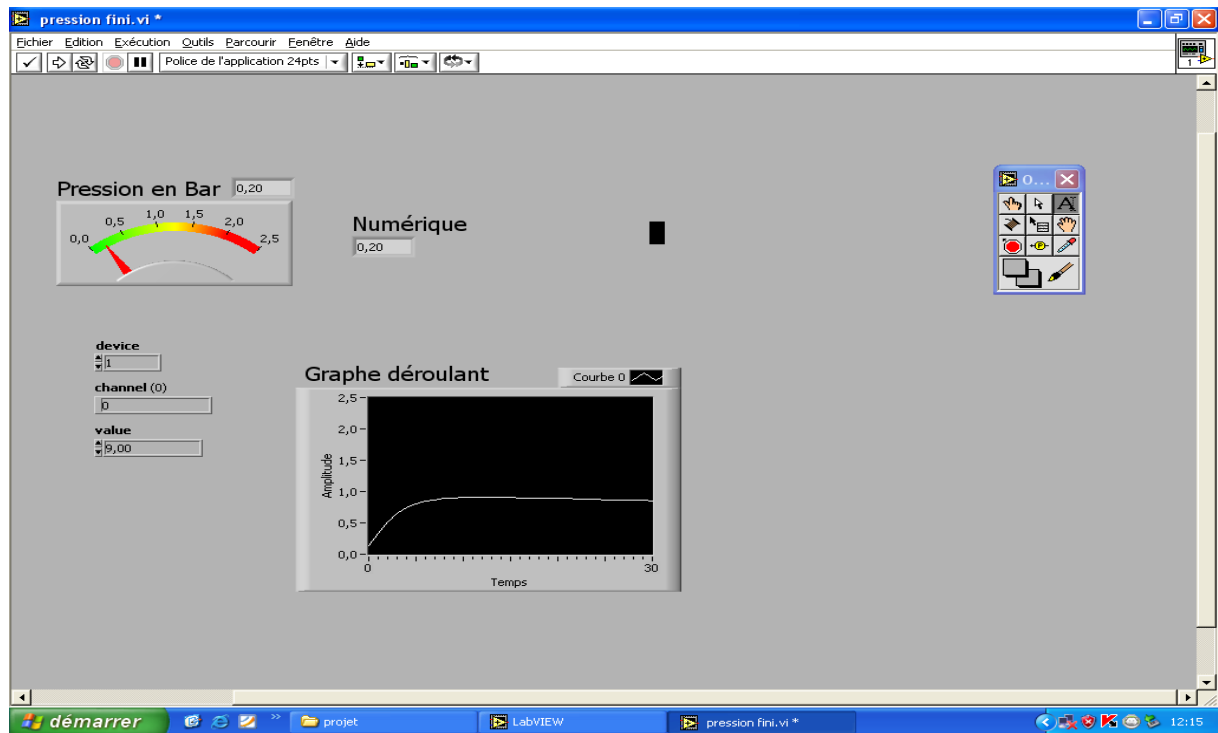


Figure 4.6 Face avant.

4.3 Identification du procédé

Les données étant acquises, il faut maintenant passer à l'étape d'identification du procédé et obtenir son modèle pour ensuite le valider. En effet, le fichier sauvegardé précédemment contient toutes les mesures, est exportés soit dans Matlab ou dans Excel pour les traités.

Dans notre cas, on a utilisé Matlab pour tracer l'allure de la courbe représentative de ces mesures (**Figure 4.7**) et pour identifier par la suite les paramètres du modèle mathématique du procédé.

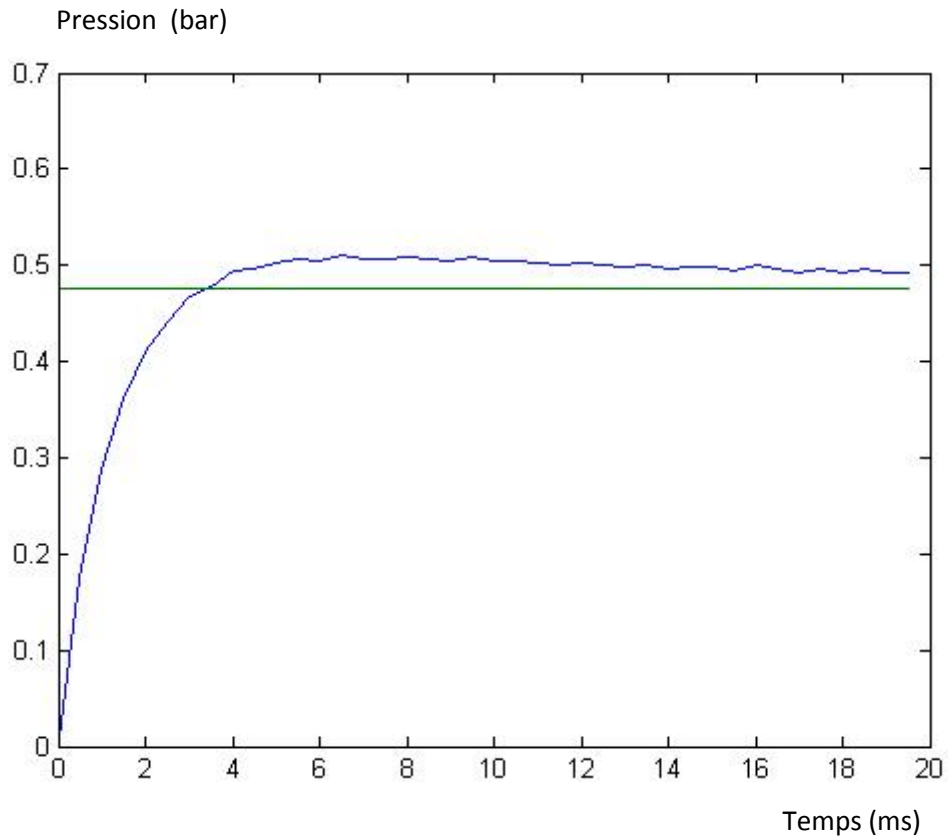


Figure 4.7 Réponse du système (sortie mesurée).

4.3.1 Méthode d'identification utilisée

On remarque que la sortie obtenue suit une allure d'un système de premier ordre avec les caractéristiques suivantes :

- Sans retard ;
- Lisse non bruitée ;
- Apériodique (amortie) ;

Ainsi, pour la première étape d'identification, on s'intéresse à la structure du modèle. Par conséquent d'après les remarques faites précédemment, on peut modéliser le comportement dynamique de la station par un modèle de premier ordre. Donc, la fonction de transfert est :

$$G(s) = \frac{k}{Ts+1} \quad (4.1)$$

Puisque la structure du modèle est simple (typiquement du premier ordre sans retard), on propose de procéder à l'estimation des paramètres T (constante de temps) et k (gain statique) par une méthode graphique à l'aide de Matlab.

k représente le gain du système, il est obtenu en se basant sur la notion du théorème de la valeur finale, comme suit :

La fonction de transfert du système est

$$G(s) = \frac{k}{Ts+1} = \frac{y(s)}{u(s)} \quad (4.2)$$

Avec

$$u(t) = A \quad \text{qui est un échelon d'amplitude } A \text{ égal à 9 volts}$$

Donc

$$y(s) = G(s).u(s) = \frac{k}{Ts+1} \cdot \frac{A}{s} \quad (4.3)$$

En utilisant le théorème de la valeur finale, il vient :

$$y(t)_{t \rightarrow +\infty} = \lim_{s \rightarrow 0} sy(s) = \lim_{s \rightarrow 0} s \frac{k}{Ts+1} \frac{A}{s} = kA = 0.5 \quad (4.4)$$

$$\Rightarrow k = \frac{0.5}{A} = \frac{0.5}{9} = 0.055$$

T représente la constante de temps du système. Elle est calculée en utilisant la relation entre la constante de temps et le temps de réponse d'un élément de premier ordre suivante :

$$T_r = 3T \Rightarrow T = \frac{T_r}{3} \quad (4.5)$$

avec T_r est le temps de réponse du système à 95% est égal à 3.2964 s obtenu avec la fonction « ginput » de Matlab, alors on déduit facilement que $T = 1.0988$ s.

On obtient finalement le modèle donné sous forme de la fonction de transfert de ce système suivante :

$$G(s) = \frac{0.055}{1.0988s+1} \quad (4.6)$$

4.4 Validation du modèle

En représentant la sortie prédite par le modèle et la sortie mesurée, c'est-à-dire la réponse de la fonction de transfert $G(s)$ sollicité par un échelon d'amplitude $A=9$ volts, et on la comparant avec celle du système réel (**Figure 4.8**), on peut confirmer que ce modèle peut bien être validé, car il reproduit fidèlement le comportement du procédé physique étudié.

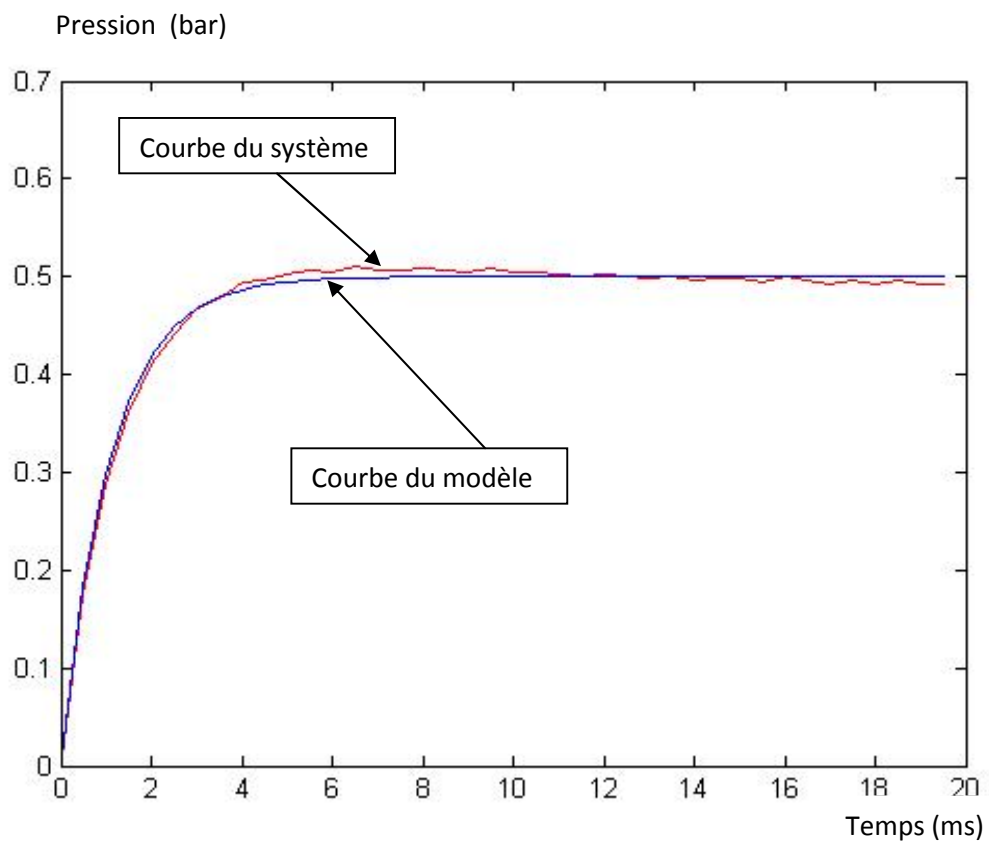


Figure 4.8 Sortie prédite par le modèle et celle mesurée.

Conclusion

Dans ce chapitre, on a présenté brièvement les différents éléments constituant la station de pression qui a fait l'objet de l'étude présentée dans ce mémoire. Ensuite, on a détaillé les différentes étapes réalisées pour l'obtention du modèle mathématique du compresseur à savoir la façon dont on a procédé pour l'acquisition des données avec le logiciel LabView, et la méthode d'identification utilisée pour l'estimation de ses paramètres. Enfin, on a conclu par une validation du modèle.

Conclusion générale

Conclusion générale

Le travail présenté dans ce mémoire s'inscrit dans le cadre de l'identification des systèmes dynamiques. Il consiste à identifier une station de pression se trouvant au laboratoire pédagogique du département Automatique. L'objectif est d'acquérir les mesures à l'aide du logiciel LabView.

Ainsi, après avoir présenté des généralités sur les systèmes dynamiques et les modèles mathématiques. On a abordé les différentes méthodes d'identification, et détaillé l'étape d'acquisition des données d'une manière générale, puis en particulier en utilisant le logiciel LabView. La dernière partie est réservée à une application complète concernant l'identification graphique de la station de pression.

Pour l'obtention du modèle mathématique, les différentes étapes ont été bien expliquées avec les précautions à prendre pour réussir l'identification.

Avant d'entamer l'identification, évidemment on a besoin de données ou des mesures. Pour cela, on a fait appel à un logiciel d'acquisition : LabView. Ce dernier est un logiciel développé par National Instrument utilisé dans divers domaines en particulier pour l'acquisition en temps réel et il utilise un langage graphique simple pour la programmation.

Une fois les données en main, l'identification a été réalisée à l'aide du logiciel Matlab en adoptant une méthode d'identification graphique vu que la caractéristique de la courbe obtenue ne représentait aucune difficulté. Le modèle ainsi obtenu a été bien validé par la suite.

En somme, le concept de base de la programmation graphique du logiciel LabView facilite la tâche et permet de gagner du temps non seulement pour l'identification, mais pour d'autres applications par exemple la commande.

Nous estimons qu'il est très intéressant de tester ou vérifier l'efficacité de ce logiciel en étudiant des systèmes multivariables ou des systèmes présentant des fortes linéarités. Une autre piste intéressante est d'implémenter un système de commande sous l'environnement LabView.

Bibliographie

- [1] **ARZELIER, D.**, « *Présentation et analyse des systèmes linéaires* ».
<http://www.laas.fr/~arzelier>
- [2] **BORNE, P.**, “*Modélisation et identification des processus* “. Edition Technip, 1992.
- [3] **FLAUS, J.M.**, “ *La régulation industrielle régulateurs PID + Prédictifs et flous* “. Edition Hermes, 2000.
- [3] **FLAUS, J.M.**, “ *La régulation industrielle régulateurs PID + Prédictifs et flous* “. Edition Hermes, 2000.
- [4] **Landau, L. D.**, “ *Identification des systèmes* “. Edition Hermes, 1998.
- [5] **COUR D’IDENTIFICATION** « *1^{ère} année Master automatique* ». 2008/2009
- [6] **Landau .**, “*Commande des systèmes , conception, identification et mise en œuvre* “. Edition Lavoisier, 2002.
- [7] **Asch, G.**, “ *Acquisition des données du capteur à l’ordinateur* “. Edition DUNOD, 2003
- [8] **MOUADH, S.**, « *Projet de fin d’étude* ».
http://www.bh-automation.fr/Download/Automaticiens/Projets_automatisme_2007_S_MOUADH.pdf
- [9] **KEITHLEY.**, User ‘s Manual (document inclus dans le driver de la carte d’acquisition)

Résumé du mémoire :

L'objectif de ce présent mémoire est d'acquérir les mesures d'une station de pression existant dans le laboratoire D'automatique industrielle et de régulation au niveau du département Automatique de l'Université Mouloud MAMMERI de Tizi-Ouzou à l'aide du logiciel LabView développé par la société National Instrument afin de l'identifier.

Après une première partie qui sera sous forme des notions générales sur les systèmes et le modèle mathématique tout en détaillant les différents types de ce dernier, le mémoire abordera l'identification des procédés et sa mise en œuvre. Ensuite, il définira l'acquisition des données avec LabView. L'application en dernière partie concrétisera l'objectif cité précédemment.

Mots clés :

Modèles, identification, acquisition de données, LabView, pression.